



IBM Systems - iSeries

アプリケーション表示プログラミング

バージョン 5

SC88-4031-00

(英文原典：SC41-5715-01)





IBM Systems - iSeries

アプリケーション表示プログラミング

バージョン 5

SC88-4031-00

(英文原典：SC41-5715-01)

ご注意！

本書および本書で紹介する製品をご使用になる前に、777 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM i5/OS (製品番号 5722-SS1) のバージョン 5、リリース 4、モディフィケーション 0 に適用されます。また、改訂版で断りがない限り、それ以降のすべてのリリースおよびモディフィケーションに適用されます。このバージョンは、すべての RISC モデルで稼働するとは限りません。また CISC モデルでは稼働しません。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： IBM Systems - iSeries
Application Display Programming
Version 5 Release 4

発 行： 日本アイ・ピー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

第1刷 2006.2

この文書では、平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 1997, 2006. All rights reserved.

© Copyright IBM Japan 2006

目次

図	xiii
表	xvii
「アプリケーション表示プログラミング」(SC88-4031) について	xix
「アプリケーション表示プログラミング」に対する変更の要約	xxi
第 1 部 オンライン・ヘルプ情報のあるサンプル画面の作成	1
第 1 章 オンライン・ヘルプ情報のあるサンプル画面の作成	3
アプリケーション表示	3
オンライン・ヘルプ情報	3
第 2 部 ディスプレイ・ファイルを使用したアプリケーション表示のプログラミング	11
第 2 章 ディスプレイ・ファイル内での表示の定義	17
ディスプレイ・ファイルの設定	17
ファイル記述の決定	18
データの記述をプログラムの内部または外部のいずれで行うかの決定	19
ディスプレイ・ファイルと記述の作成	20
ファイル記述の変更	21
ファイル記述の変更の検出	22
レコード様式内での表示フィールドおよび機能の定義	23
ディスプレイ・ファイルの DDS	23
プログラムによって使用されるレコード様式	24
画面上のレコード様式	24
フィールド属性文字について	24
レコード様式フィールドが使用される方法について	25
機能キーの定義	27
コマンド・アテンション (CAnn) およびコマンド機能 (CFnn) キーの定義	28
代替キーの指定	29
標識による情報の引き渡し	30
メッセージ記述からの固定フィールド・テキストを挿入する	31
右から左へのカーソル移動の許可	31
入力可能位置にのみカーソル移動を定義する	31
入力フィールドのカーソル移動先の定義	33

入力フィールドの属性の定義	34
編集マスクを使用した入力フィールドの保護	34
右から左への表示処理の指定	34
フィールドの語の折り返しの指定	35
フィールドの強調	36
カラーの追加	37
出力フィールドの編集	38
ユーザー独自の編集コードの定義	38
有効な画面サイズの指定	39
画面を印刷できるようにする	40
ウィンドウの定義	41
プログラム記述データの使用	42
入力専用ファイルの定義	43
出力専用ファイルの定義	43
入出力ファイルの定義	43

第 3 章 アプリケーションでのディスプレイ・ファイルの処理	45
システムがリソースを割り振る方法についての理解	45
ディスプレイ・ファイルのオープン	46
入出力操作用のディスプレイ装置の獲得	46
ディスプレイ・ファイルおよびディスプレイ装置についての情報の入手	47
オープンおよび入出力操作についての情報の入手	47
ディスプレイ装置についての属性情報の入手	48
データの送信および受信	48
ディスプレイ上のアクティブ・レコード様式の判別	49
ディスプレイへの出力の書き込み	49
ディスプレイに対する入力の送信勧誘	73
ディスプレイからの送信勧誘された入力の読み取り	74
ディスプレイからの入力の読み取り	76
出力書き込みおよび入力読み取りの同時実行	83
待機されなかった入力の取り消し	83
入出力操作中のキーボードのロックおよびカーソルの位置決め	83
前に表示した情報の保管	85
コマンド・キーに対する入出力操作の影響についての理解	87
5250 ディスプレイ装置上のレコード様式の問題の回避	87
獲得されたディスプレイ装置の入出力操作からの解放	90
ディスプレイ・ファイルのクローズ	91
表示操作の高水準言語へのマッピング	91
同一ジョブ内でのディスプレイ・ファイルの共用	92
ジョブ内で共用されるファイルのオープン操作についての理解	92
ジョブ内で共用されるファイルの入出力操作についての理解	93

ジョブ内で共用されるファイルのクローズ操作に
ついての理解 93

第 4 章 サブファイルを使用したレコード のグループの表示 95

サブファイルの使用法についての理解	95
DDS ソースでのサブファイルの記述	97
プログラムでのサブファイルの使用	104
サブファイルに関する入出力操作の要求	106
サブファイル・レコード様式に関する入出力操作 の要求	106
サブファイル制御レコード様式に関する入出力操 作の要求	108
高水準言語でのサブファイル入出力要求について の理解	109
サブファイルの外観の制御	109
表示モードでサブファイルを横方向に表示する	111
ページ・サイズと等しいサブファイル・サイズの指 定	112
ページ・サイズと等しくないサブファイル・サイズ の指定	114
サブファイル・データに関する妥当性検査	115
サブファイルからのエラー・メッセージの表示	116
表示されているサブファイル上のカーソルの位置決 め	117
カーソルの最初の位置決め	117
ロール・キー使用時のカーソルの位置決め	117
継続または切り捨てキー使用時のカーソルの位置 決め	120
複数のレコードが表示される場合のカーソルの位 置決めとロール	120
サブファイル DDS およびプログラム論理の例につ いての理解	124

第 5 章 ディスプレイ・ファイルを使用 したウィンドウの定義 127

ウィンドウの用語	128
DDS ウィンドウ・キーワード	128
ウィンドウの表示内容とハードウェア構成	128
ウィンドウの作成	129
ウィンドウ定義レコード	129
ウィンドウ参照レコード	129
ウィンドウ・サイズおよび場所	130
カーソル位置	131
エラー・メッセージ	132
サブファイル	132
DDS ヘルプ・レコード	132
ウィンドウ枠の定義	133
枠のデフォルト	133
複数の枠定義	133
UIM ヘルプ・ウィンドウ枠	134
ウィンドウ・タイトルの定義	135
ウィンドウ・タイトル例の DDS	136
ウィンドウからのデータの読み取り	136
ウィンドウの枠と内容の変更	136
ウィンドウの移動および複写	137

一度に 2 つのウィンドウを活動状態のように表示 する	137
一連のウィンドウの中の 1 つを目立たせる	138
ウィンドウの除去	138
すべてのウィンドウの除去	138
新しい方のウィンドウの除去	139
アプリケーションのパフォーマンスの改善	139
システムの保管および復元の操作	139
システムの保管および復元操作の迂回	140
プログラミングの例	141
基本的なウィンドウ機能の使用法	142
別のディスプレイ・ファイルでのウィンドウの定 義	147
RPG プログラム・ソース	147
WINPGM の RPG プログラム・ソース	148

第 6 章 画面用のグラフィカルな表示の 作成 151

グラフィカルな表示に影響を与える要素	151
ハードウェア構成	151
拡張表示パラメーター	153
DDS キーワード	154
メニュー・バーの作成	157
メニュー・バー選択項目の定義	157
メニュー・バー区切り記号の抑制	158
メニュー・バー区切り記号の定義	158
選択フィールドの概要	160
選択フィールド用 DDS の例	161
垂直単一項目選択フィールドの作成	162
垂直複数項目選択フィールドの作成	163
水平選択フィールドの作成	163
垂直項目選択フィールドにおけるカーソル移動	164
水平項目選択フィールドにおけるカーソル移動	165
選択フィールド内の選択標識の制御	166
単一項目選択フィールドを使用したプルダウン・メ ニューの作成	167
プルダウン・メニュー内の選択標識の制御	169
アクセラレーター・キーの定義	170
メニュー・バー切り替えキーの定義	171
取り消しキーの定義	172
カーソルがプルダウン・メニューの外にある場合 の機能の制限	172
選択リストの概要	173
選択リスト用 DDS の例	174
選択リストの作成	175
選択リスト内の選択標識の制御	176
スクロール・バーの概要	178
スクロール・バーの作成	179
スクロール・バー用 DDS の例	181
スクロール・バーの操作	182
押しボタンの概要	182
押しボタン用 DDS の例	183
押しボタンの作成	183
選択項目の選択可能性の制御	184
単一項目選択フィールドにおける自動選択	185
単一項目選択フィールドにおける自動入力	185

略号の定義	186
選択項目のカラーおよび属性の定義	187
継続入力フィールドの概要	190
継続入力フィールドでの語の折り返しの指定	191
継続入力フィールドでの DBCS に関する考慮事項	191
継続入力フィールドの DBCS データを戻す方法	192
継続入力フィールドでのキーボード機能	192
前方フィールド出口処理	196
後方フィールド出口処理	197
メニュー・バーがアプリケーションと相互作用する方法	197
アプリケーション・レコードでの MNUBARDSP キーワードの定義	197
メニュー・バー・レコードでの MNUBARDSP キーワードの定義	199
プルダウン・メニューからの入力データの受け取り	200
入力データを受け取った後のプルダウン・メニューの除去	202
表示前のプルダウン・メニューの更新	202
アプリケーション・ヘルプの定義	203
選択項目レベルのヘルプの定義	203
フィールド用ヘルプの定義	206
メニュー・バーとプルダウン・メニューのキーの相互作用	207
カーソル移動	208
タブ・キーを押す	208
カーソル・キーを押す	209
プログラミングの例	209
アプリケーション・レコードでの MNUBARDSP キーワードの使用	209
メニュー・バー・レコードでの MNUBARDSP キーワードの使用	212
画面の外観	213
単純ホット・スポット	214
コマンド・キー・エミュレーション	215
前ページおよび次ページ・キー・エミュレーション	215
プログラマブル・マウス・ボタン - 概要	216
ポインター装置イベント	216
戻される AID コード	217
プログラマブル・マウス・ボタン - 利点	217
プログラマブル・マウス・ボタンの操作	218
プログラマブル・マウス・ボタン - NWS の考慮事項	219
プログラマブル・マウス・ボタン - イベント処理の状態	219
プログラマブル・マウス・ボタン - イベント処理の優先順位	220
グリッド・ライン構造の概要	226
グリッド・ライン構造用 DDS の例	227
グリッド・ライン構造およびウィンドウ	228
グリッド・ライン構造におけるハードウェア要件	228
HTML タグの挿入	229
HTML フィールドのオーバーラップの解決	229

プログラミングの例	231
-----------	-----

第 7 章 ディスプレイ・ファイルとディスプレイ・ファイル属性のオーバーライド

ド	235
オーバーライドを使用するかしないかの判別	235
HLL プログラムでのファイル属性のオーバーライド	235
例	236
HLL プログラムでのファイル名のオーバーライド	237
例	237
HLL プログラムでのファイル名と属性の両方のオーバーライド	238
例	238
プログラムのコンパイル時のオーバーライドの適用	238
例	239
オーバーライドの削除	240
オーバーライドの表示	240
ファイルのリダイレクトを使用したファイル名およびライブラリーまたはファイル・タイプのオーバーライド	240
同じファイル・タイプでのファイルのオーバーライド	240
異なるファイル・タイプでのファイルのオーバーライド	240
オーバーライドを無視または制限するコマンドの認識	243

第 8 章 ディスプレイ・ファイルに関するメッセージとエラーの処理

独自のメッセージの作成および表示	247
メッセージ行にメッセージを表示する	248
サブファイル制御レコードが書き込まれる際にメッセージ行にメッセージを表示する	248
メッセージ・フィールドを使用してメッセージ行にメッセージを表示する	249
メッセージ行にメッセージを表示する場合の優先順位	249
ディスプレイ上のフィールドにメッセージを表示する	249
プログラム・メッセージ待ち行列中のメッセージを表示する	250
サブファイルを使用したエラー・メッセージの表示	250
メッセージに関するアラームの鳴動	253
ディスプレイ装置での永続的な入出力エラーの自動処理	253
システムから送信されるエラー・メッセージの分析	254
メッセージおよびメッセージ・モニターの説明	255
メジャー/マイナー戻りコードの説明	256
エラーからの回復	257

第 9 章 ディスプレイ・ファイルを使用したメニューの作成およびアクセス

システム・メニューとユーザー定義メニューの実行	261
-------------------------	-----

GO コマンドを実行後、メニューに戻る	261
直前のメニューの判別	261
メニューでの取り消しキーおよび終了キーの使用	262
サインオン時に表示されるメニューの選択	263
ユーザー独自のディスプレイ・ファイル・メニュー の定義	263
メニューに関する DDS およびディスプレイ・フ ァイル考慮事項	264
メッセージ・ファイル内のメニュー・アクション の記述	265
メニュー用のヘルプ・フォーマットの命名	265
ディスプレイ・ファイル・メニューの作成	267
ユーザー独自のプログラム・メニューの定義	270
プログラム・メニューのパラメーターの引き渡し	270
プログラム・メニューの作成	271
前のメニューに戻らずに、プログラム・メニュー から出る	273
メニュー名の競合の回避	275
メニューの命名	275
ライブラリー・リスト内の高位ライブラリーへの メニューの配置	275
メニューを含むライブラリーの指定	275
一般メニュー仕様の使用	275
コマンド複製後のコマンド・デフォルトの変更	275
メニュー属性の表示	276
メニュー属性の変更	276
メニューの削除	276

第 10 章 ユーザー定義のデータ・スト リームの使用	277
ディスプレイ装置の違いについて	277
ユーザー定義のデータ・ストリームの制限について	278

第 11 章 プログラム間でのデータの受 け渡し	283
ジョブ内の同じ経路指定ステップ内でのデータの受 け渡し	283
ジョブ内の経路指定ステップ内でのデータの受け渡 し	284

第 12 章 ディスプレイ・ファイル、ICF ファイル、およびデータ待ち行列からの 入力待ち	287
---	------------

第 13 章 代替文字セットおよびコー ド・ページの使用	289
システムに装置で通常表示されない文字がある	289
装置がシステム上で表示されない文字を渡す	290
フィールド用の文字変換を指定する	290
ご使用のディスプレイ用の文字 ID (CHRID) 値の判 別	292

第 14 章 ディスプレイでのシステム・ パフォーマンスの向上	295
ディスプレイ・ファイルの書き込み操作の据え置き	295

ディスプレイ・ファイルの 1 次画面サイズの指定	295
サブファイル・レコードを一度に 1 ページだけ書 き込む	295
同じジョブのオープン・データ・パス (ODP) を共 用する	296
入力フィールドを持つレコードを、順番に画面に送 信する	296
繰り返し送信されるレコードを重複させ、削除しな い	296
画面の復元	297
コマンド機能キーではなくコマンド・アテンショ ン・キーの定義	297
送信勧誘操作の使用	297
ウィンドウの使用	298

第 3 部 パネル・グループを使用したアプリケーション表示のプログラ ミング 299

第 15 章 ユーザー・インターフェース 管理機能による生産性の向上	301
ユーザー生産性の増大	301
アプリケーション・プログラマーの生産性の増大	301
データ記述仕様 (DDS) の代わりに UIM を使用す る前に考慮すべき事柄	301

第 16 章 ユーザー・インターフェース 管理機能の概要	303
UIM の概要	303
UIM のサポート対象	304
パネル・グループとは	304
メニューとは	305
オブジェクトの作成	305
パネル・グループ内のエレメント	305
UIM 言語タグの使用法	306
ダイアログ・コマンドの使用法	306
制御言語 (CL) コマンドの使用法	307
アプリケーション・プログラミング・インターフェ ース (API) の使用法	307
UIM を使用したメニュー・オブジェクトの定義	308
メニュー・パネルの作成	309
メニュー・パネルに必要なタグ	310
メニューの例のソース	311
UIM を使用したパネル・グループ・オブジェク トの定義	315
リスト・パネルの作成	316
リスト・パネルの必須タグ	318
リスト・パネルの例のソース	319
リスト・パネルのアプリケーション・プログラミ ング	327
確認リスト・パネルの作成	329
確認リスト・パネルに必要なタグ	330
確認パネルの例のソース	330
自動確認処理	332
確認処理のアプリケーション・プログラミング	333

データ表示パネルの作成	333
データ表示パネルに必要なタグ	336
データ表示パネルの例のソース	337
データ表示パネルのアプリケーション・プログラミング	347
データ入力パネル	350
メニュー・バーのあるパネルの作成	351
メニュー・バーのあるパネルに必要なタグ	353
メニュー・バーのあるパネルの例のソース	354
メニュー・バー・パネルのアプリケーション・プログラミング	363

第 17 章 ユーザー・インターフェース

管理機能の詳しい使用方法 365

UIM アプリケーションを開く	365
ダイアログ変数の定義	365
ダイアログ変数の使用上の制約事項	367
ダイアログ変数のエラー・メッセージ	367
ダイアログ変数を使用する表示パネルへのフィールド値の提供	368
変数プール・サービスの使用方法	368
ダイアログ変数および特殊値	369
文字セットおよびコード・ページについての考慮事項	369
リストの管理	371
リストの定義	372
リストの初期化	372
リストの表示	372
リストの更新	373
不完全リストの処理	373
リストからの項目の除去および挿入	373
リスト画面上のリスト項目の制御	374
リスト画面の対話応答時間の改善	374
処置リストおよび選択リストの使用	375
選択文字の使用	377
パネル機能の管理	377
GUI への変換の使用可能化	378
スクロールのサポート	378
スクロール可能域の定義	378
機能キー・スクロールの定義	378
スクロールおよびエラー条件	379
リスト域のスクロール	379
メニュー域のスクロール	379
情報域のスクロール	379
データ項目グループのスクロール	380
テキスト域のスクロール	380
文脈ヘルプの定義	380
コマンド行の制約事項	382
コマンド行の解釈	382
長すぎるコマンドの入力	383
機能キーの定義	383
機能キーの書式設定	383
機能キーおよび VARUPD 値の処理	384
パネルの書式設定の概念	385
パネルの書式設定をいつ行うか	385
パネルの書式設定のアプリケーション制御	386

パネル・フォーマッターの制限	387
EXIT の要求時に複数のパネルをまとめて閉じる	387
リスト・パネルをまとめて閉じる	388
別のパネルへのポップアップ・ウィンドウの追加	389
メニュー・バーの使用方法	390
プルダウン・メニューとポップアップ・ウィンドウとの違い	391
ポップアップ・ウィンドウの使用方法	393
アプリケーション・ウィンドウの定義	393
ウィンドウの追加および除去	393
ウィンドウ内でのコマンド行の使用方法	394
パネルを表示する際の要求処理プログラムとしての UIM	395
印刷の概念	396
印刷ヘッド・パネルの印刷	398
印刷パネルの印刷	398
分離のためブランク行の使用	398
フォントおよび強調表示	398
トレーラーの印刷	398
プロログ域の定義	398
ヘッダー域の定義	399
印刷中のページ排出機能の使用方法	399
プリンター・ファイルの共用およびオーバーライド	399
2 バイト文字セット (DBCS) 印刷の考慮事項	400
UIM に関するよく尋ねられる質問	400

第 4 部 ヘルプ表示のプログラミング 401

第 18 章 ディスプレイ・ファイルのオンライン・ヘルプ情報をアクセス可能にする 403

ヘルプ・キーの使用可能化	404
ヘルプにパネル・グループとレコードのいずれかを 選択する	404
オンライン・ヘルプ情報を必要とする画面領域の定義	405
ディスプレイ・ファイルでヘルプ用にパネル・グループを指定する	407
オプション標識を使用したパネル・グループの定義	410
パネル・グループを使用したヘルプを指定する QUSRTOOL 例のコピー	411
ディスプレイ・ファイルでレコードを指定する	411
オプション標識を使用したレコードの定義	412
ヘルプ情報が含まれるレコードの入力	413
同じディスプレイ・ファイルにおけるヘルプ用のレコードおよび文書の使用	413
レコードに関する制約事項の理解	414
レコードを使用するヘルプ画面間でのページング	415
ヘルプ・キーを押した後に制御をプログラムに戻す	418
ヘルプ画面を表示した後に制御をプログラムに戻す	419
ヘルプ画面を表示せずに制御をプログラムに戻す	420

第 19 章 パネル・グループのオンライン・ヘルプを利用できるようにする . . . 421

定義と説明	421
索引検索へのヘルプ・パネル・グループ・アクセス権を与える	423
ユーザー定義パネル・グループへのヘルプ・パネル・グループ・アクセス権を与える	423
「F18=続きの見出し」へのアクセス権を除去する	424
リスト域のヘルプ	424
ヘルプのコーディング	425
メニュー域のヘルプ	426
ヘルプのコーディング	426
データ域のヘルプ	428
ヘルプのコーディング	429
メニュー・バー域のヘルプ	431
ヘルプのコーディング	432
機能キー域のヘルプ	433
ヘルプのコーディング	434

第 20 章 オンライン・ヘルプ情報の定義 437

パネル・グループ中のオンライン・ヘルプ情報の定義	437
ヘルプ用にパネル・グループの UIM ソースを入力する	437
パネル・グループの作成および削除	450
パネル・グループをコマンドのヘルプとして割り当てる	450
検索索引でのパネル・グループの使用法	451
パネル・グループ中のヘルプを定義する QUSRTOOL 例のコピー	452
DDS レコードでのオンライン・ヘルプ情報の定義	452

第 5 部 IBM i5/OS-Style 画面の指針 453

第 21 章 IBM i5/OS スタイルの画面の設計 455

QUSRTOOL ライブラリーにおける画面例の使用	455
サンプル・オブジェクトの認識	455
サンプル・オブジェクトのインストール	457
サンプル画面、コマンド、およびオンライン・ヘルプ情報の表示	457
独自に使用するためのサンプル・オブジェクトのソースのコピー	461
すべての画面に対する特殊機能および属性の定義	462
単一選択メニュー表示の設計	462
タイトル	463
指示行	463
メニュー・オプション	463
メニュー選択項目フィールド	464
機能キー	464
オンライン・ヘルプ情報	464
一般メニュー画面の操作	464

入力画面の設計	465
タイトル	465
指示行	465
プロンプト域	466
機能キー	470
オンライン・ヘルプ情報	470
一般入力画面の操作	470
情報画面の設計	471
タイトル	472
位置情報	472
プロンプト域 1.	473
プロンプト域 2.	474
指示行	474
機能キー	474
オンライン・ヘルプ情報	474
一般情報画面の操作	474
リスト画面の設計	474
タイトル	475
プロンプト域	476
指示行	476
オプション行	476
列見出し	477
拡張処置項目域	478
リスト・フィールド	478
ページング位置情報	478
機能キー	479
オンライン・ヘルプ情報	479
一般リスト画面の操作	479
すべての画面用の機能キー域の定義	486
任意指定のコマンド行および ID フィールド	487
共通キーの割り当て	490
すべての画面のヘルプ情報の定義	495
メニュー画面のヘルプ	495
入力画面のヘルプ	496
情報画面のヘルプ	497
リスト画面のヘルプ	498
メッセージの定義と表示	499
共通ユーザー・アクセス (CUA) 基本レベル・モデルの設計	501
基本ダイアログ処置	501
単一項目選択 (メニュー)	502
入力画面	504
情報画面	505
リスト画面	506
ヘルプ情報	507

第 6 部 付録 509

付録 A. UIM パネル・グループ定義言語 511

段落としてフォーマットされるタグの内容	512
パネル域	513
パネル	513
パネル・グループ・オブジェクト	513
パネルに関するヘルプ	514
パネル・グループの編成	514
名前構文	515

シンボル	515	オプションのテキスト	558
注釈	517	DATAI (データ項目)	559
埋め込み	517	必須属性	560
DBCS 図形リテラル	517	オプションの属性	560
16 進リテラル	518	オプションのテキスト	563
APPFMT (アプリケーション・フォーマット済み領域)	518	DATAIX (データ項目エクステンダー)	564
必須属性	519	必須属性	565
オプションの属性	519	オプションの属性	565
アプリケーション・フォーマット済みデータ	519	DATASLT (データ選択フィールド)	568
例: アプリケーション・フォーマット済み領域	521	必須属性	568
BOTINST (最下部指示)	521	オプションの属性	569
オプションの属性	521	オプションのテキスト	570
オプションのテキスト	522	例 1: データ入力パネル	570
CHECK (妥当性検査)	522	例 2: 複数選択フィールド	571
必須属性	522	DATASLTC (データ選択フィールド選択項目)	573
オプションの属性	523	オプションの属性	573
例: 妥当性検査	524	オプションのテキスト	575
CIT (タイトルの引用)	524	DL (定義リスト)	575
オプションのテキスト	524	オプションの属性	576
例: タイトルの引用	524	必須タグ	576
CLASS (クラス定義)	525	オプションのタグ	576
必須属性	526	例 1: 定義リスト	577
オプションの属性	531	例 2: 短縮定義リスト	577
例: クラス定義	536	FIG (図表)	577
数値の表示画面形式	537	オプションの属性	578
文字、日付、および時刻値の表示画面形式	538	オプションのタグ	578
CMDLINE (コマンド行)	539	例: サンプル図表	578
必須属性	539	HELP (ヘルプ・モジュール)	579
オプションの属性	540	必須属性	579
オプションのテキスト	540	オプションの属性	579
COND (条件定義)	540	オプションのテキスト	580
必須属性	540	例: ヘルプ・パネル定義	581
オプションの属性	543	HP0 から HP9 (強調表示句)	582
例: オプションの条件付け	543	オプションのテキスト	583
COPYR (著作権)	544	H1 から H4 (見出し)	583
必須テキスト	544	必須テキスト	584
例: 見出しタグ	584	IMHELP (組み込みヘルプ)	584
DATA (データ表示域)	544	必須属性	585
必須属性	545	例: 組み込みヘルプ	585
オプションの属性	546	IMPORT (インポート)	586
オプションのテキスト	548	必須属性	586
印刷の書式設定の考慮事項	548	オプションの属性	587
例 1: データ入力パネル	548	INFO (情報域)	587
例 2: データ入力パネルでの 2 列形式	549	必須属性	588
例 3: データ項目用の 2 つの表示域	551	オプションの属性	588
例 4: メニュー域を持つデータ表示域	552	オプションのテキスト	588
例 5: ネストされたデータ・グループを持つデータ入力パネル	553	印刷のフォーマットの考慮事項	588
DATAC (データ項目選択)	554	ISCH (索引検索)	588
オプションの属性	554	必須属性	589
オプションのテキスト	555	必須テキスト	589
DATACOL (データ列)	555	例: 索引検索	589
必須属性	555	ISCHSUBT (索引検索サブトピック)	590
オプションのテキスト	556	必須属性	590
DATAGRP (データ・グループ)	556	例: 索引検索階層	590
オプションの属性	557	ISCHSYN (索引検索同義語)	591

必須属性	591	必須属性	634
必須テキスト	591	オプションの属性	634
例: 索引検索同義語	592	例: メニュー・バー	634
KEY1 (キー・リスト項目)	593	MBARC (メニュー・バー選択項目)	636
必須属性	593	必須属性	636
オプションの属性	594	必須テキスト	637
オプションのテキスト	595	MENU (メニュー域)	637
例: キー定義	595	必須属性	638
KEYL (キー・リスト)	596	オプションの属性	638
必須属性	596	オプションのテキスト	638
オプションの属性	596	例 1: 単純なメニュー域	638
例: キー・リスト	596	例 2: グループを持つメニュー域	639
LINES (不定形式行)	597	MENUGRP (メニュー・グループ)	641
オプションのテキスト	597	オプションの属性	641
例: 不定形式行	597	オプションのテキスト	641
LINK (ハイパーテキスト・リンク定義)	598	MENUI または MI (メニュー項目)	642
必須属性	599	必須属性	642
オプションの属性	599	オプションの属性	643
条件式	600	オプションのテキスト	644
双方向の考慮事項	601	NT または NOTE (注記)	644
例: ハイパーテキスト・リンク	601	オプションのテキスト	645
LIST (リスト域)	603	例: 注記の使用	645
必須属性	605	OL (番号付きリスト)	645
オプションの属性	605	オプションの属性	646
オプションのテキスト	608	必須タグ	646
印刷のフォーマットの考慮事項	608	例: 番号付きリスト	646
例 1: リスト域	609	OPTLINE (オプション行)	647
例 2: 3 つのレイアウト列を持つリスト域	610	オプションの属性	647
例 3: リスト列グループを持つリスト域	611	オプションのテキスト	647
例 4: 動的リスト列見出しのフォーマット	612	P (段落)	648
LISTACT (リスト・アクション)	614	オプションのテキスト	648
必須属性	615	例: 段落タグ	648
オプションの属性	615	PANEL (表示パネル)	649
オプションのテキスト	619	必須属性	650
確認パネルの要件	619	オプションの属性	650
確認パネルの規則	619	オプションのテキスト	655
例: リスト・アクション	620	例: パネル定義	656
LISTCOL (リスト列)	621	PARML (パラメーター・リスト)	656
必須属性	621	必須タグ	657
オプションの属性	622	例: パラメーター・リスト	658
オプションのテキスト	624	PC (段落の連結)	658
フォーマットの考慮事項	625	例: 段落の連結	658
LISTDEF (リスト定義)	625	PDACCEL (プルダウン・アクセラレーター)	659
必須属性	626	必須テキスト	659
オプションの属性	626	PDFLD (プルダウン・フィールド)	659
LISTGRP (リスト列グループ)	628	オプションの属性	660
必須属性	629	PDFLDC (プルダウン・フィールド選択項目)	660
オプションの属性	629	必須属性	661
オプションのテキスト	629	オプションの属性	661
例: リスト列グループ	630	オプションのテキスト	663
LISTVIEW (リスト・ビュー)	631	確認パネルの要件	663
必須属性	631	確認パネルの規則	664
オプションの属性	631	PK (プログラミング・キーワード)	664
LP (リストの一部)	633	オプションの属性	665
例: リストの一部	633	必須テキスト	665
MBAR (メニュー・バー)	633	PNLGRP (パネル・グループ)	665

オプションの属性	666	XH1 から XH4 (拡張ヘルプの見出し)	696
PRTHEAD (印刷ヘッド・パネル)	668	必須テキスト	696
必須属性	669	フォーマットの規則	696
オプションの属性	669	例: サンプル見出し	697
オプションのテキスト	670	XMP (例)	697
タイトル行のレイアウト	671	例: 例のフォーマット設定	698
例: 印刷タイトル行	673	付録 B. UIM ダイアログ・コマンド	699
PRTPNL (印刷パネル)	673	VARUPD 属性	702
必須属性	674	ACTIONS (メニュー・バー・カーソル・アクション)	702
オプションの属性	674	メッセージ	702
オプションのテキスト	674	CALL (プログラム呼び出し)	703
PRTRAIL (トレーラー・メッセージ印刷)	675	必須パラメーター	703
必須テキスト	675	CANCEL	705
例: トレーラー・メッセージ	675	オプション・パラメーター	705
PV (プログラミング変数)	675	CHGVIEW (ビュー変更)	705
必須テキスト	676	パラメーター	706
RT (テキスト反転)	676	VARUPD 値	706
例 1: 右から左方向のパネル上の、左から右方向 のフォーマット設定	677	メッセージ	706
例 2: 左から右方向のパネル上の、左から右方向 のフォーマット設定	678	CMD (システム・コマンド)	706
SL (単純リスト)	678	パラメーター	706
オプションの属性	679	VARUPD 値	707
必須タグ	679	メッセージ	707
例: 単純リスト	679	ヒント	707
TEXT (テキスト域)	680	CMDLINE (コマンド行)	707
必須属性	680	パラメーター	707
オプションの属性	681	メッセージ	707
カーソル位置	681	DSPHELP (ヘルプ表示)	707
テキスト・データ	682	必須パラメーター	707
例: テキスト域	685	オプション・パラメーター	707
TI (変換リスト項目)	685	ENTER	708
オプションの属性	685	パラメーター	708
オプションのテキスト	686	VARUPD 値	708
TL (変換リスト)	686	メッセージ	708
オプションの属性	686	ENTER、HELP、および PROMPT ダイアログ・ コマンドの使用における考慮事項	708
例: 変換リスト	687	EXIT (表示終了)	709
TOPINST (最上部指示)	687	オプション・パラメーター	709
オプションの属性	688	EXTHELP (拡張ヘルプ)	710
オプションのテキスト	688	パラメーター	710
TT (真理値表)	688	HELP	710
必須属性	689	パラメーター	710
例: 真理値表	689	VARUPD 値	710
TTROW (真理値表行)	689	メッセージ	710
必須属性	689	HELPHelp	710
UL (順不同リスト)	690	パラメーター	711
オプションの属性	691	メッセージ	711
必須タグ	691	HELPIPDX	711
例: 順不同リスト	691	パラメーター	711
VAR (変数定義)	692	メッセージ	711
必須属性	692	HOME (ホーム・メニュー表示)	711
オプションの属性	692	パラメーター	711
UIM によって定義されるダイアログ変数	693	VARUPD 値	711
VARRCD (変数レコード定義)	695	メッセージ	712
必須属性	695	KEYSHELP	712
オプションの属性	695		

メッセージ	712
MENU	712
必須パラメーター	712
オプション・パラメーター	712
MOREKEYS (機能キー続き表示)	713
パラメーター	713
メッセージ	713
MOVETOP (先頭へ移動)	713
パラメーター	714
VARUPD 値	714
メッセージ	714
MSG (メッセージ表示)	714
必須パラメーター	714
オプション・パラメーター	714
PAGEDOWN	715
パラメーター	715
メッセージ	715
PAGEUP	715
パラメーター	715
VARUPD 値	716
メッセージ	716
PRINT (画面印刷)	716
パラメーター	716
VARUPD 値	716
メッセージ	716
PROMPT	716
パラメーター	716
VARUPD 値	717
メッセージ	717
入力フィールドのプロンプト	717
処置リスト・オプションまたはコマンドのプロンプト	717
PULLDOWN (プルダウン・メニュー表示)	718
パラメーター	718
RETRIEVE (コマンド・ストリング取得)	719
パラメーター	719
VARUPD 値	719
メッセージ	719
RETURN (アプリケーションへの制御戻り)	719
必須パラメーター	719

付録 C. ディスプレイ・ファイルのフィールドバック域のレイアウト 721

オープン・フィールドバック域	721
装置定義リスト	724
入出力フィールドバック域	727
共通入出力フィールドバック域	727
ディスプレイ・ファイルに関する入出力フィールドバック域	731
取得属性	733

付録 D. ディスプレイ・ファイルの戻りコード 739

メジャー・コード 00	739
-------------	-----

メジャー・コード 02	740
メジャー・コード 03	740
メジャー・コード 04	741
メジャー・コード 08-11	741
メジャー・コード 34	742
メジャー・コード 80	743
メジャー・コード 81	746
メジャー・コード 82	748
メジャー・コード 83	752

付録 E. 編集コード 757

i5/OS 編集コード	757
i5/OS 編集コードを使用した編集の例	758
ユーザー定義編集コード	759
ユーザー定義編集コードの使用法	760
ユーザー定義編集コードの例	761

付録 F. System/36 と互換性のある画面データ管理機能 763

画面上の行の消去	764
ディスプレイ・ファイル・レコード用の入力データ	764
ワークステーション制御装置からの入力データ	765
自己検査	765
リターン入力	765
入力フィールドの消去	766
表示属性	766
カーソルの位置決め	767
メッセージの表示	767
一時変更の PUT	769
符号付き数字データの処理	770
機能キー	771
ヘルプ・キーの考慮事項	771
コマンド・キーを使用したアプリケーション・ヘルプの終了	772
送信勧誘取り消し操作	772
コマンドおよび機能キーの保持	773
サポートされない System/36 の機能	774
制限された DDS キーワード/機能	774

特記事項 777

プログラミング・インターフェース情報	778
商標	778
使用条件	779

参考文献 781

システム使用	781
システム管理	781
アプリケーション開発	781
通信および接続	782
プログラム・イネーブラー	782
プログラム・インターフェース	783

索引 785



1. ディスプレイ・ファイルのサンプル DDS ソース	23
2. プログラム内および画面上のレコード様式	24
3. フィールド・レベルのカーソル移動先の DDS	33
4. サブファイル・レベルのカーソル移動先の DDS	33
5. 右から左への表示処理の DDS 例	35
6. プログラム-システム間フィールドの DDS 例	37
7. 2 つの画面サイズ用の DDS 例	40
8. 2 つの画面サイズ用のサブファイル用の DDS 例	40
9. CLRL キーワードを使用しない場合の画面上のレコードの有効な配置	50
10. CLRL キーワードを使用しない場合の画面上のレコードの間違った配置	51
11. レコード様式の置換	53
12. SLNO(*VAR) キーワードの使用法を示すサンプル DDS ソース	56
13. CLRL と OVERLAY の相違を示すサンプル DDS ソース	59
14. PUTOVR キーワードの使用法を示すサンプル DDS ソース	62
15. PUTOVR キーワードの効果的な使用法を示すサンプル DDS ソース	64
16. PUTOVR キーワードの別の使用法を示すサンプル DDS ソース	65
17. PUTRETAIN キーワードの使用法を示すサンプル DDS ソース	67
18. PUTRETAIN キーワードの使用法を示すサンプル DDS ソース	68
19. DSPMOD キーワードの使用法を示すサンプル DDS ソース	70
20. レコード様式の問題を示すサンプル DDS	89
21. サブファイル制御用の DDS キーワードを処理する順序	104
22. サブファイルを縦方向に表示する	110
23. サブファイルを横方向に表示する	110
24. 横方向に表示するサブファイルと縦方向に表示するサブファイルを同時に表示する	110
25. DSPMOD とサブファイルを併用したサンプル DDS	111
26. SFLSIZ、SFLPAG、および ROLLUP キーワードを使用したサンプル DDS	112
27. 可変長レコードの場合のサンプル DDS	114
28. SFLNXTCHG キーワードを使用したサンプル DDS	115
29. メッセージ・サブファイルの場合のサンプル DDS	116
30. DSPATR(PC) キーワードを使用したサンプル DDS	118
31. 顧客名検索サブファイルを示すサンプル DDS	124
32. ウィンドウ・タイトルのディスプレイの例	135
33. ウィンドウ・タイトルの DDS	136
34. ラジオ・ボタンとチェック・ボックス	151
35. メニュー・バーの例	157
36. メニュー・バー用の DDS	158
37. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上のメニュー・バー	159
38. 下線機能を持つ非グラフィカル・ディスプレイ装置上のメニュー・バー	159
39. 下線機能がない非グラフィカル・ディスプレイ装置上のメニュー・バー	159
40. 拡張インターフェースがないディスプレイ装置上のメニュー・バー	160
41. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上の選択フィールド	160
42. 非グラフィカル・ディスプレイ装置上の選択フィールド	161
43. 単一項目および複数項目選択フィールドのための DDS	162
44. 水平選択フィールド用 DDS の例	163
45. 選択フィールド内の選択標識を抑制するための DDS	166
46. 選択フィールド内で抑制された選択標識	166
47. プルダウン・メニューの例	167
48. プルダウン・メニュー用の DDS	168
49. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上のプルダウン・メニュー	168
50. 下線機能を持つ非グラフィカル・ディスプレイ装置上のプルダウン・メニュー	168
51. 下線機能がない非グラフィカル・ディスプレイ装置上のプルダウン・メニュー	168
52. 拡張インターフェースがないディスプレイ上のプルダウン・メニュー	169
53. プルダウン・メニュー内の選択標識を抑制するための DDS	169
54. グラフィカル・ディスプレイ装置上で抑制された選択標識	169
55. 非グラフィカル・ディスプレイ装置上で抑制された選択標識	170
56. アクセラレーター・キー用の DDS	171
57. プルダウン・メニュー内のアクセラレーター	171
58. メニュー・バー切り替えキーおよび取り消しキー用の DDS	172
59. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上の選択リスト	173
60. 下線機能を持つ非グラフィカル・ディスプレイ装置上の選択リスト	174
61. 選択リスト用 DDS の例	175
62. 選択リスト内の選択標識を使用可能にするための DDS	177

63. グラフィカル・ディスプレイ装置上の選択標識	178	97. i5/OS 5250 ゲートウェイ・ディスプレイでのグラフィック・イメージ	232
64. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上のスクロール・バー	179	98. ERRSFL キーワードに関するサンプル DDS ソース	251
65. 下線機能を持つ非グラフィカル・ディスプレイ装置上のスクロール・バー	179	99. SFLMSGID キーワードに関するサンプル DDS ソース	252
66. スクロール・バー用 DDS の例	181	100. PERSMENU というサンプル・メニュー用の DDS ソース	268
67. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上の押しボタン	182	101. プログラム・メニュー例用の DDS ソース	271
68. 下線機能を持つ非グラフィカル・ディスプレイ装置上の押しボタン	182	102. メニューの例	309
69. 下線機能がない非グラフィカル・ディスプレイ装置上の押しボタン	183	103. メニュー・パネルの必須 UIM タグ	311
70. 拡張インターフェースがないディスプレイ装置上の押しボタン	183	104. リスト・パネルの例	316
71. 押しボタン用 DDS の例	183	105. リストの代替ビューの例	317
72. CHCCTL キーワードの制御値	184	106. リスト・パネルの必須タグ	319
73. 選択項目の選択可能性を制御するための DDS	185	107. 確認リスト・パネルの例	329
74. 使用不可の選択項目がある単一項目選択フィールド	185	108. データ表示パネルの例	334
75. 略号が有効である DDS の例	186	109. スクロール後のデータ表示パネルの例	335
76. 略号が無効である DDS の例	186	110. データ表示パネルの必須 UIM タグ	336
77. メニュー・バー選択項目のために CHCAVAIL および CHCSLT を使用する DDS	188	111. メニュー・バーのあるパネルの例	351
78. 選択フィールドのために CHCAVAIL および CHCUNAVAIL を使用する DDS	189	112. メニュー・バーのあるパネルの必須 UIM タグ	353
79. 選択フィールドのために CHCAVAIL、CHCUNAVAIL、および CHCSLT を使用する DDS	189	113. ジョブ終了フラグの例	388
80. 単一項目選択リスト選択項目のために CHCAVAIL、CHCUNAVAIL、および CHCSLT を使用する DDS	190	114. 印刷出力の例	397
81. 複数項目選択リスト選択項目のために CHCAVAIL、CHCUNAVAIL、および CHCSLT を使用する DDS	190	115. DDS ソース内の H 仕様のサンプル	406
82. 長方形に配置された継続入力フィールド	191	116. HLPPNLGRP を示すサンプル DDS ソース	410
83. アプリケーション・レコード上で MNUBARDSP を使用する DDS	198	117. HLPPNLGRP およびオプション標識を示すサンプル DDS ソース	411
84. メニュー・バー・レコード上で MNUBARDSP を使用する DDS	199	118. Sample DDS Source Showing HLPRCD	412
85. プルダウン入力 (PULLINPUT) パラメーター用 DDS	201	119. HLPRCD およびオプション標識を示すサンプル DDS ソース	413
86. フィールド戻しパラメーター用 DDS	203	120. HLPRCD を使用したサンプル DDS ソース	413
87. メニュー・バー選択項目のヘルプ用 DDS	204	121. 2 次ヘルプを示すサンプル DDS ソース	417
88. 単一選択フィールド選択項目のヘルプ用 DDS	204	122. HLPCMDKEY を示すサンプル DDS ソース	419
89. メニュー・バーのヘルプ・リスト	206	123. HLPCMDKEY および応答標識を示すサンプル DDS ソース	420
90. 名前付きフィールド用ヘルプ	206	124. 索引検索画面	423
91. 固定情報フィールド用ヘルプ	206	125. QUSRTOOL のサンプル・メニュー	458
92. カーソル位置	207	126. QUSRTOOL のサンプル入力画面	459
93. グリッド・ライン構造	226	127. QUSRTOOL のサンプル情報画面 (2 ページ)	460
94. グリッド・ライン構造用 DDS の例	227	128. QUSRTOOL のサンプル・リスト画面	461
95. HTML キーワードを追加する前の DDS コーディング	231	129. アプリケーション・メニューのサンプル	463
96. HTML キーワードを追加した後の DDS コーディング	232	130. サンプル入力画面	465
		131. サンプル情報画面 (2 ページ)	472
		132. サンプル・リスト画面	475
		133. 位置情報を使用した画面のレイアウト	479
		134. リスト画面を使用した処理の優先順位の例	482
		135. ユーザー・プロファイル入力画面の作成	488
		136. 選択される追加パラメーターを使用したコマンド作成画面	489
		137. 追加パラメーターの 2 番目の画面	489
		138. 「PDM を使用したメンバーの処理」入力画面	490
		139. 入力画面のヘルプ域	497
		140. 情報画面のヘルプ域	498
		141. リスト画面のヘルプ域	499
		142. アプリケーション・メニューの例	503
		143. 入力画面	504

144. 情報画面の例	505	146. テキスト域で使用可能な強調表示クラス	683
145. リスト画面の例	506	147. 保管コマンドの迂回	773

表

1. サンプル画面の作成ステップで使用される名前	3	36. サンプル画面のヘルプ	407
2. アプリケーション表示を作成する別の方法	8	37. UIM ヘルプ用の他の DDS キーワード	407
3. オンライン・ヘルプ情報を作成する別の方法	8	38. パネル・グループを使用したサンプル画面のヘルプ	409
4. サンプル DDS の桁の位置	23	39. HLPRCD を使用したサンプル画面のヘルプ	412
5. フィールドを強調する DDS	36	40. QUSRTOOL の画面例のソース・メンバー (インストール、作成、および削除)	455
6. Print キーを使用する PRINT キーワードの結果	41	41. QUSRTOOL の画面例のソース・メンバー (サンプル画面、コマンド、およびオンライン・ヘルプ情報)	456
7. ディスプレイ・ファイルのオープン時に暗黙的に獲得されるディスプレイ装置	46	42. サンプル・オブジェクトを作成する際に作成されるオブジェクト	457
8. 属性取得操作で使用可能な情報	48	43. すべての i5/OS スタイルの画面に必要な機能および属性	462
9. SLNO(*VAR) 値の結果	55	44. すべての i5/OS スタイルの画面に任意指定の機能および属性	462
10. CLRL 例の結果	59	45. 機能キーの割り当て	491
11. 表示モードが変更される場合に無視されるキーワード	70	46. ヘルプ域ごとのヘルプのタイプ - メニュー画面	495
12. オペレーティング・システムでサポートされるディスプレイ・ファイル操作および同等の高水準言語コマンド	91	47. DDS の考慮事項 - メニュー画面のヘルプ	496
13. サブファイルに関するオプション機能	98	48. ヘルプ域ごとのヘルプのタイプ - 入力画面	496
14. システムによってサポートされるサブファイル操作および同等の HLL コマンド	109	49. DDS の考慮事項 - 入力画面のヘルプ	497
15. ハードウェア構成 A、B、および C でサポートされる機能	151	50. ヘルプ域ごとのヘルプのタイプ - 情報画面	497
16. ハードウェア構成 D、E、および F でサポートされる機能	153	51. DDS の考慮事項 - 情報画面のヘルプ	498
17. スクロール・バーがサイズ変更される方法	181	52. ヘルプ域ごとのヘルプのタイプ - リスト画面	498
18. スクロール・バーの操作	182	53. DDS の考慮事項 - リスト画面のヘルプ	499
19. カラーおよび表示属性を定義するために使用されるキーワード	187	54. CUA 基本ダイアログ処置	501
20. MNUCHOICE および PULLINPUT に戻される値	201	55. 継続可能なタグ属性	512
21. ささまざまなカーソル位置で実行されるアクション	207	56. 混合パネルに関連した制約事項	513
22. ファイルのリダイレクト	241	57. CLASS タグと ECLASS タグの間で許可されるタグ	526
23. ファイルのリダイレクトの組み合わせ	242	58. BASETYPE ごとの属性要約	530
24. システム・メッセージ番号の範囲	255	59. DATA タグと EDATA タグの間で許可されるタグ	545
25. メジャー戻りコードの定義	256	60. DATASLT タグと EDATASLT タグの間で許可されるタグ	568
26. ディスプレイ・ファイル・メニューの制約事項	264	61. DL タグと EDL タグの間で許可されるタグ	576
27. ディスプレイ・ファイル・メニューのための提案	265	62. KEYL タグと EKEYL タグの間で許可されるタグ	596
28. ディスプレイ・ファイルおよび ICF ファイル入力フィールド属性	288	63. LIST タグと ELIST タグの間で許可されるタグ	604
29. CHRID 値	292	64. NOCMD と NOEXT 属性の相互作用	617
30. ダイアログ変数の初期値	366	65. 強調値	627
31. 表示用の UIM CCSID/CHRID 変換	369	66. Width=80 のレイアウト値	632
32. 印刷用の UIM CCSID/CHRID 変換	370	67. WIDTH=132 のレイアウト値	632
33. ダイアログ・コマンドに割り当てられるカーソル移動に影響される機能キー	392	68. WIDTH=132 のレイアウト値	632
34. オンライン・ヘルプ情報を定義するさまざまな方法	404	69. MBAR タグと EMBAR タグの間で許可されるタグ	634
35. オンライン・ヘルプ情報に関する異なる方式の特性	404	70. MBARC タグと EMBARC タグの間で許可されるタグ	636

71. MENU タグと EMENU タグの間で許可されるタグ	638		87. 印刷幅が 80 でタイム・ゾーンを持つヘッダーの 2 番目の行	673
72. MENUGRP タグと EMENUGRP タグの間で許可されるタグ	641		88. PRTPNL タグと EPRTPNL タグの間で許可されるタグ	674
73. NOTE タグと ENOTE タグの間で許可されるタグ	644		89. SL タグと ESL タグの間で許可されるタグ	678
74. OL タグと EOL タグの間で許可されるタグ	646		90. TL タグと ETL タグの間で許可されるタグ	686
75. PANEL タグと EPANEL タグの間で許可されるタグ	649		91. TT タグと ETT タグの間で許可されるタグ	689
76. UIM の指紋のレイアウト	652		92. UL タグと EUL タグの間で許可されるタグ	690
77. PARML タグと EPARML タグの間で許可されるタグ	657		93. UIM によって定義される変数 (Z 変数) の属性	694
78. PDFLD タグと EPDFLD タグの間で許可されるタグ	660		94. ダイアログ・コマンドの有効な使用法の要約	699
79. ACTION 値の有効アクション・テキスト	661		95. ダイアログ・コマンドの影響の要約	700
80. PNLGRP タグと EPNLGRP タグの間で許可されるタグ	665		96. オープン・フィードバック域	721
81. PRTHEAD タグと EPRTHEAD タグの間で許可されるタグ	669		97. 装置定義リスト	724
82. 印刷幅が 132 のヘッダーの最初の行	671		98. 共通入出力フィードバック域	727
83. 印刷幅が 80 のヘッダーの最初の行	671		99. ディスプレイ・ファイルに関する入出力フィードバック域	731
85. 印刷幅が 132 でタイム・ゾーンを持つヘッダーの 2 番目の行	672		100. 取得属性	733
86. 印刷幅が 80 のヘッダーの 2 番目の行	672		101. i5/OS 編集コードの要約表	757
			102. 有効な編集コード、ソース・データ、および編集済みの出力	759
			103. IBM 提供の編集記述	759
			104. MSGID 用のメッセージ・ファイル	768
			105. MSGID 用のメッセージ・ファイル	768

「アプリケーション表示プログラミング」(SC88-4031) について

本書には、以下のトピックについての情報が記載されています。

- DDS を使用した、アプリケーションの表示の作成および保守
- システム上のディスプレイ・ファイルの作成および処理
- オンライン・ヘルプ情報の作成
- UIM を使用した、アプリケーションのパネルおよびダイアログの定義

本書は、アプリケーション表示およびヘルプ表示用のプログラムを作成するために使用します。本書は、表示装置の構成または DDS キーワードのすべてを記述しているわけではありません。特別なトピックに関する特定情報については、他の IBM 資料を参照することが必要な場合があります。関連資料のリストについては、781 ページの『参考文献』を参照してください。

本書は、アプリケーション表示およびヘルプ表示を作成または処理するアプリケーション・プログラマーを対象としています。また、原始ステートメント入力ユーティリティ (SEU) およびデータ記述仕様 (DDS) についての知識が必要です。

本書では、ご使用の表示装置をシステムに記述する装置記述がすでに存在していると想定しています。

「アプリケーション表示プログラミング」に対する変更の要約

「アプリケーション表示プログラミング」のこのエディションには、この資料が最後に出版された後に生じた多数の技術更新が含まれています。変更が加えられた場所には、変更情報の横の余白に垂直バーが示されています。さらに、オペレーティング・システムから除去された UIM キーワードへの参照のいくつかは削除されました。

第 1 部 オンライン・ヘルプ情報のあるサンプル画面の作成

第 1 章 オンライン・ヘルプ情報のあるサンプル画面 の作成	3
アプリケーション表示	3
オンライン・ヘルプ情報	3

第 1 章 オンライン・ヘルプ情報のあるサンプル画面の作成

本章では、i5/OS でオンライン・ヘルプ情報のあるサンプル画面を作成するために行う必要のあるステップの概略を示します。ステップの実行方法がよく分からない場合には、それぞれのステップで参照されている追加情報を参照してください。

アプリケーション表示

サンプル画面は、ディスプレイ・ファイル (表示装置ファイルともいう) を使用して作成されます。ディスプレイ・ファイルは、ユーザーによって作成される、ファイル記述を含むオブジェクトまたは名前付きストレージ・スペースです。ファイル記述は、使用されるディスプレイ装置を識別し、オプションで、ディスプレイ装置によって使用されるレコード様式を識別します。レコード様式は、画面上のフィールドの特性および配置を記述します。レコード様式はデータ記述仕様 (DDS) を使用して定義します。DDS は、データを処理するアプリケーション・プログラムの外部で、データ属性を記述します。

オンライン・ヘルプ情報

サンプル画面のオンライン・ヘルプ情報は、ヘルプ・パネル・グループを使用して定義されます。パネル・グループは、ユーザー・インターフェース管理機能によってオンライン・ヘルプ情報として使用されるテキストを含むオブジェクト、または名前付きストレージ・スペースです。ユーザー・インターフェース管理機能 (UIM) は、表示の一部またはすべてのヘルプ、コマンドのヘルプ、索引検索機能 (選択可能ヘルプ・トピック)、およびハイパーテキスト (オンライン・ヘルプ情報の異なる単位をリンクする機能) といった、表示に関するオンライン・ヘルプ情報を提供する、オペレーティング・システムの一機能です。

次の表に、ステップで使用されるサンプル名をリストします。

表 1. サンプル画面の作成ステップで使用される名前

名前	説明
SRCSAM	サンプル・ソース・ファイル
ADMSAM	アプリケーション表示のサンプル・ソース・メンバー
DSPSAM	サンプル・ディスプレイ・ファイル
HDMSAM	ヘルプ表示のサンプル・ソース・メンバー
PNLSAM	サンプル・パネル・グループ
LIBSAM	ソース・ファイル SRCSAM、ディスプレイ・ファイル DSPSAM、およびパネル・グループ PNLSAM を含むサンプル・ライブラリー

以下のステップでは、オンライン・ヘルプ情報のあるサンプル画面の作成方法を 1 つだけ示します。他の方法については、本章の末尾で解説します。

1. ソース物理ファイル作成 (CRTSRCPF) コマンドを使用してソース・ファイル SRCSAM を作成し、ライブラリー作成 (CRTLIB) コマンドを使用してライブラリー LIBSAM を作成します。
2. プログラム開発管理機能開始 (STRPDM) コマンドを入力して、プログラム開発管理機能 (PDM) の使用を開始します。画面が表示されたら、オプション 3 (Work with members) を選択します。

プログラム開発管理機能 (PDM)

次の1つを選択してください。

1. ライブラリーの処理
2. オブジェクトの処理
3. メンバーの処理

9. ユーザー定義オプションの処理

選択項目またはコマンド

====> 3

F3= 終了 F4= プロンプト F9= コマンドの複写 F10= コマンド入力
F12= 取り消し F18= 省略時の値の変更
(C) COPYRIGHT IBM CORP. 1981, 2005.

追加情報

プログラム開発管理機能 (PDM) は、Application Development ToolSet ライセンス・プログラムの一部分で、ライブラリー、オブジェクト、およびメンバーのリストからいくつかの操作 (コピー、削除、名前変更など) を実行できます。

ディスプレイ・ファイル (ステップ 9 で作成) は、実際にはデータを含まないのに、ディスプレイ・ファイルの DDS ソースは、ソース・ファイルに入力されます。**ソース・ファイル**は、1 つ以上の**ソース・メンバー**から成るオブジェクトです。ソース・メンバーは、ご使用の DDS ソースを構成するさまざまなデータ集合のことです。

プログラム開発管理機能について詳しくは、「AS/400 プログラム開発管理機能 (PDM)」に記載されています。

3. 「Specify Members to Work With」画面が表示されたら、ファイルおよびライブラリー情報をすべて入力して、Enter キーを押します。

処理するメンバーの指定

選択項目を入力して、実行キーを押してください。

ファイル SRCSAM 名前, リストは F4
ライブラリー LIBSAM *LIBL, *CURLIB, 名前
メンバー:
名前 *ALL *ALL, 名前, * 総称 *
タイプ *ALL *ALL, タイプ, * 総称 *, *BLANK

F3= 終了 F4= プロンプト F5= 最新表示 F12= 取り消し

4. 「Work with Members Using PDM」画面が表示されます。

PDM を使用したメンバーの処理

ファイル SRCSAM
ライブラリー LIBSAM 位置指定

オプションを入力して、実行キーを押してください。

2= 編集 3=コピー 4= 削除 5= 表示 6= 印刷 7= 名前の変更
8= 記述の表示 9= 保管 13=テキストの変更 14=コンパイル 15=モジュール作成 ...

OPT メンバー タイプ テキスト

(ファイル中にメンバーがない)

パラメーターまたはコマンド

====>
F3= 終了 F4= プロンプト F5= 最新表示 F6= 作成
F9= コマンドの複写 F10= コマンド入力 F23=オプションの続き F24=キーの続き

この画面で F6 を押して、新規メンバーを作成します。

5. 「Start Source Entry Utility」画面が表示されます。この画面で情報をすべて入力して、Enter キーを押します。

SEU開始 (STRSEU)

選択項目を入力して、実行キーを押してください。

```
ソース・ファイル . . . . . > SRCSAM      名前 , *PRV
ライブラリー . . . . . > LIBSAM       名前 , *LIBL, *CURLIB, *PRV
ソース・メンバー . . . . .          ADMSAM     名前 , *PRV, *SELECT
ソース仕様タイプ . . . . .          DSPFE      名前 , *SAME, BAS, BASP...
テキスト ' 記述 ' . . . . .          サンプル画面の DDS
```

終了

F3= 終了 F4=プロンプト F5= 最新表示 F12= 取り消し
F13= この画面の使用法 F24= キーの続き

追加情報

原始ステートメント入力ユーティリティー (SEU) は、ソース・メンバーの作成と変更に使われる Application Development ToolSet ライセンス・プログラムの機能です。原始ステートメント入力ユーティリティーについて詳しくは、「AS/400 適用業務開発ツールセット AS/400 用 原始ステートメント入力ユーティリティー 使用者の手引きと参照」に記載されています。

- 新規メンバーを作成しているのので、ブランク行の画面を持つ SEU 「Edit」画面が表示されます。画面の最後の行のテキスト *Member ADMSAM added to file SRCSAM* は、SEU が指定されたファイルに新規メンバーを追加したことを示します。

```
桁 . . . . . :   1 71          編集          LIBSAM/SRCSAM
SEU==>          ADMSAM
FMT DP . . . . . AAN01N02N03T.NAME+++++RLEN++TDPBLINPOFUNCTIONS+++++
***** データの始め *****
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
***** データの終わり *****
```

F3= 終了 F4=プロンプト 5= 最新表示 F9=コマンドの複写 F10=カーソル F11= 切り替え
F16= 検索の反復 F17= 変更の反復 F24= キーの続き
メンバー ADMSAM がファイル LIBSAM/SRCSAM に追加された。 +

- SEU 編集画面で、新規メンバーの DDS ソース・ステートメントを入力します。この章の指示に従うと、サンプル画面用のオンライン・ヘルプ情報を提供できるので、ヘルプを使用可能にしてそれにアクセスするために必要な DDS キーワードを必ず DDS ソースに含めてください。

追加情報

SEU には、DDS の入力を支援する多くの機能があります。SEU の機能について詳しくは、「AS/400 適用業務開発ツールセット AS/400 用 原始ステートメント入力ユーティリティー 使用者の手引きと参照」に記載されています。

DDS を使用した表示の記述については、本書の 23 ページの『レコード様式内での表示フィールドおよび機能の定義』を参照してください。オンライン・ヘルプ情報に必要な DDS キーワードについて詳しくは、『第 18 章 ディスプレイ・ファイルのオンライン・ヘルプ情報をアクセス可能にする』を参照してください。

8. ご使用の DDS ソースの入力を完了したら、F3 を押して、「Exit」画面の情報を完成させます。Enter キーを押します。
9. 新規ディスプレイ・ファイルを作成するには、任意のコマンド行でディスプレイ・ファイル作成 (CRTDSPF) コマンドを入力します。ご使用の DDS ソースを含むソース・ファイルおよびメンバーを必ず指定してください。

```
CRTDSPF FILE(LIBSAM/DSPSAM) SRCFILE(LIBSAM/SRCSAM) SRCMBR(ADMSAM)
```

Enter キーを押します。

追加情報

ディスプレイ・ファイルの作成について詳しくは、20 ページの『ディスプレイ・ファイルと記述の作成』を参照してください。

10. オンライン・ヘルプ情報をサンプル画面に追加するには、UIM ソースに対する 2 番目のソース・ファイル・メンバーを作成します。2 番目のソース・メンバーは、ご使用の DDS ソースを作成したのと同じソース・ファイル内に作成する必要はありません。しかし、DDS および UIM ソースの両方に対して同じソース・メンバーを使用しないでください。

2 番目のソース・ファイル・メンバーを作成するには、ステップ 2 から 4 を繰り返します。「Start Source Entry Utility」画面が再び表示されたら、ステップ 11 を継続します。

11. 以下のように指定して、「Start Source Entry Utility」画面の情報を完成させます。

プロンプト 入力する情報

Source member HDMSAM

Source type PNLGRP

Text description

 UIM help for sample display

12. SEU 編集画面で、新規メンバーの UIM ソース・ステートメントを入力します。

追加情報

UIM を使用したオンライン・ヘルプ情報の定義については、437 ページの『パネル・グループ中のオンライン・ヘルプ情報の定義』を参照してください。

13. ご使用の UIM ソースの入力を完了したら、F3 を押して、「Exit」画面の情報を完成させます。Enter キーを押します。
14. パネル・グループ作成 (CRTPNLGRP) コマンドを任意のコマンド行に入力します。
 CRTPNLGRP PNLGRP(LIBSAM/PNLSAM) SRCFILE(LIBSAM/SRCSAM) SRCMBR(HDMSAM)
- Enter キーを押します。

追加情報

パネル・グループについて詳しくは、450 ページの『パネル・グループの作成および削除』を参照してください。

前のステップでは特定のライブラリーを使用しましたが、それと同じライブラリー内に画面用の各種オブジェクトを作成する必要はありません。

システム上でユーザー独自のアプリケーション表示を作成する他の方法は、以下のとおりです。

表2. アプリケーション表示を作成する別の方法

方法	説明	追加情報の入手先
画面設計機能 (SDA)	Application Development ToolSet ライセンス・プログラムの機能で、画面およびメニューの設計、作成、および保守を支援します。SDA により、画面の外観を思いどおりに設計できます。その後システムが、ユーザー用の DDS ソースおよびディスプレイ・ファイルを作成します。	「AS/400 適用業務開発ツールセット AS/400 用 画面設計機能 (SDA)」
QUSRTOOL	ご使用のシステムにオプションでインストール可能なライブラリー。これにより、アプリケーション開発やシステム管理に役立つ、各種ツールおよびプログラミング手法の例にアクセスできます。QUSRTOOL 内の画面例は、オンライン・ヘルプ情報のある 4 つのサンプル画面を提供します。選択したライブラリーにこれらの画面のソースをコピーしてから、ユーザーの要件に合うように調整できます。	455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』
UIM パネル・グループ	ご使用のアプリケーションのパネルおよびダイアログを定義することのできる、システムのパーツ。UIM は、パネルの外観を制御し、IBM® により開発されたパネルとの整合性を確保します。	303 ページの『第 16 章 ユーザー・インターフェース管理機能の概要』

ご使用の画面のオンライン・ヘルプ情報を作成する他の方法は、以下のとおりです。

表3. オンライン・ヘルプ情報を作成する別の方法

方法	説明	追加情報の入手先
ヘルプ・レコード	DDS キーワード。これによって、DDS ソースと同じまたは別のメンバー内に、ご使用のアプリケーション表示のオンライン・ヘルプ情報を作成できます。これは、ディスプレイ・ファイルからはアクセスできますが、UIM アプリケーション・パネルからはアクセスできません。	452 ページの『DDS レコードでのオンライン・ヘルプ情報の定義』 および 411 ページの『ディスプレイ・ファイルでレコードを指定する』

オンライン・ヘルプ情報を作成する各種方法を比較および対比するには、404 ページの『ヘルプにパネル・グループとレコードのいずれかを選択する』を参照してください。

第 2 部 ディスプレイ・ファイルを使用したアプリケーション表示のプログラミング

第 2 章 ディスプレイ・ファイル内での表示の定義	17	出力専用ファイルの定義	43
ディスプレイ・ファイルの設定	17	入出力ファイルの定義	43
ファイル記述の決定	18		
フィールド・レベル記述	18		
レコード・レベル記述	18		
ファイル・レベル記述	19		
データの記述をプログラムの内部または外部のいずれで行うかの決定	19		
外部記述データ	19		
プログラム記述データ	20		
ディスプレイ・ファイルと記述の作成	20		
ファイル記述の変更	21		
ファイル記述の変更の検出	22		
レコード様式内での表示フィールドおよび機能の定義	23		
ディスプレイ・ファイルの DDS	23		
プログラムによって使用されるレコード様式	24		
画面上のレコード様式	24		
フィールド属性文字について	24		
レコード様式フィールドが使用される方法について	25		
機能キーの定義	27		
コマンド・アテンション (CAnn) およびコマンド機能 (CFnn) キーの定義	28		
代替キーの指定	29		
標識による情報の引き渡し	30		
オプション標識および応答標識をレコード域から除去する	30		
同時に異なる応答標識を使用可能にする	30		
標識をオフに設定する	31		
メッセージ記述からの固定フィールド・テキストを挿入する	31		
右から左へのカーソル移動の許可	31		
入力可能位置にのみカーソル移動を定義する	31		
入力フィールドのカーソル移動先の定義	33		
入力フィールドの属性の定義	34		
編集マスクを使用した入力フィールドの保護	34		
右から左への表示処理の指定	34		
フィールドの語の折り返しの指定	35		
フィールドの語の折り返しの指定 - ヒント	36		
フィールドの強調	36		
カラーの追加	37		
出力フィールドの編集	38		
ユーザー独自の編集コードの定義	38		
有効な画面サイズの指定	39		
画面を印刷できるようにする	40		
ウィンドウの定義	41		
プログラム記述データの使用	42		
入力専用ファイルの定義	43		
		第 3 章 アプリケーションでのディスプレイ・ファイルの処理	45
		システムがリソースを割り振る方法についての理解	45
		ディスプレイ・ファイルのオープン	46
		入出力操作作用のディスプレイ装置の獲得	46
		ディスプレイ・ファイルおよびディスプレイ装置についての情報の入手	47
		オープンおよび入出力操作についての情報の入手	47
		ディスプレイ装置についての属性情報の入手	48
		データの送信および受信	48
		ディスプレイ上のアクティブ・レコード様式の判別	49
		ディスプレイへの出力の書き込み	49
		ディスプレイへのレコードの配置	49
		ディスプレイ上のスペースを占有しないレコードについての理解	52
		ディスプレイ上のレコード様式の変更	52
		ディスプレイに書き込まれるレコード様式の順序の決定	53
		ディスプレイ上のレコード様式のオーバーレイおよび消去	53
		特定の行からのレコード様式の開始	54
		指定した行数の消去	57
		ディスプレイ上の 2 行間のデータのロール	59
		属性またはフィールドの内容のオーバーライド	61
		ディスプレイ上のすべての無保護の入力および出力入力フィールドの消去	66
		ディスプレイ上のレコードに関連付けられた変更データ・タグのリセット	66
		レコードまたはフィールドのディスプレイ上での保持	66
		読み取り要求の実行時までの書き込み操作の据え置き	68
		フィールドのデフォルト値の指定	69
		レコードを表示するモードの指示	69
		出力操作後のカーソルの位置決め	71
		アプリケーションへのカーソル位置の返送	72
		アプリケーションへのサブファイル内のカーソル位置の返送	72
		アプリケーションへのサブファイルのモードの返送	72
		出力入力フィールドの初期化	72
		ディスプレイに対する入力の送信勧誘	73
		CL プログラムからの入力の送信勧誘	73
		ディスプレイからの送信勧誘された入力の読み取り	74

送信勧誘された装置からの読み取りについての理解	75
ディスプレイからの入力の読み取り	76
プログラムによるデータ処理中のキーボードのロック解除	77
入力データの保持	78
データ変更時の標識の設定	78
レコードの初期化とキーボードのロック解除の図	79
妥当性検査機能の指定	79
入力可能フィールドの数の制限についての理解	81
負の数値入力データの処理	82
システムがディスプレイから入力を読み取る方法についての理解	82
出力書き込みおよび入力読み取りの同時実行	83
待機されなかった入力の取り消し	83
入出力操作中のキーボードのロックおよびカーソルの位置決め	83
前に表示した情報の保管	85
コマンド・キーに対する入出力操作の影響についての理解	87
5250 ディスプレイ装置上のレコード様式の問題の回避	87
獲得されたディスプレイ装置の入出力操作からの解放	90
ディスプレイ・ファイルのクローズ	91
表示操作の高水準言語へのマッピング	91
同一ジョブ内でのディスプレイ・ファイルの共用	92
ジョブ内で共用されるファイルのオープン操作についての理解	92
ジョブ内で共用されるファイルの入出力操作についての理解	93
ジョブ内で共用されるファイルのクローズ操作についての理解	93
第 4 章 サブファイルを使用したレコードのグループの表示	95
サブファイルの使用法についての理解	95
DDS ソースでのサブファイルの記述	97
プログラムでのサブファイルの使用	104
サブファイルに関する入出力操作の要求	106
サブファイル・レコード様式に関する入出力操作の要求	106
サブファイル中の指定された場所にレコードを追加する	106
サブファイル中の活動状態のレコードを更新する	106
サブファイル中の指定された場所で活動状態のレコードを読み取る	107
サブファイル中の次の変更済みレコードを読み取る	107
サブファイル制御レコード様式に関する入出力操作の要求	108
サブファイル・レコードの表示	108
処理のためにディスプレイ上のサブファイル・レコードを挿入する	109

サブファイル・レコードの表示と処理を同時に行う	109
高水準言語でのサブファイル入出力要求についての理解	109
サブファイルの外観の制御	109
表示モードでサブファイルを横方向に表示する	111
ページ・サイズと等しいサブファイル・サイズの指定	112
ページ・サイズと等しくないサブファイル・サイズの指定	114
サブファイル・データに関する妥当性検査	115
サブファイルからのエラー・メッセージの表示	116
表示されているサブファイル上のカーソルの位置決め	117
カーソルの最初の位置決め	117
ロール・キー使用時のカーソルの位置決め	117
継続または切り捨てキー使用時のカーソルの位置決め	120
複数のレコードが表示される場合のカーソルの位置決めとロール	120
サブファイル DDS およびプログラム論理の例についての理解	124

第 5 章 ディスプレイ・ファイルを使用したウィンドウの定義	127
ウィンドウの用語	128
DDS ウィンドウ・キーワード	128
ウィンドウの表示内容とハードウェア構成	128
ウィンドウの作成	129
ウィンドウ定義レコード	129
ウィンドウ参照レコード	129
ウィンドウ・サイズおよび場所	130
カーソル位置	131
エラー・メッセージ	132
サブファイル	132
DDS ヘルプ・レコード	132
ウィンドウ枠の定義	133
枠のデフォルト	133
複数の枠定義	133
UIM ヘルプ・ウィンドウ枠	134
ウィンドウ・タイトルの定義	135
ウィンドウ・タイトル例の DDS	136
ウィンドウからのデータの読み取り	136
ウィンドウの枠と内容の変更	136
ウィンドウの移動および複写	137
一度に 2 つのウィンドウを活動状態のように表示する	137
一連のウィンドウの中の 1 つを目立たせる	138
ウィンドウの除去	138
すべてのウィンドウの除去	138
新しい方のウィンドウの除去	139
アプリケーションのパフォーマンスの改善	139
システムの保管および復元の操作	139
応答時間	139
システムの保管および復元操作の迂回	140
USRRSTDSP キーワードの処理と対話	141

プログラミングの例	141	選択項目の選択可能性の制御	184
基本的なウィンドウ機能の使用法	142	単一項目選択フィールドにおける自動選択	185
DDS のフルスクリーン・ディスプレイとウィ ンドウの定義	142	単一項目選択フィールドにおける自動入力	185
RPG 表示プログラム	143	略号の定義	186
ステップ 1: 初期画面の表示	145	選択項目のカラーおよび属性の定義	187
ステップ 2: ウィンドウ 1 の表示	145	継続入力フィールドの概要	190
ステップ 3: ウィンドウ 2 の表示	145	継続入力フィールドでの語の折り返しの指定	191
ステップ 4: ウィンドウ 1 の復元	146	継続入力フィールドでの DBCS に関する考慮事 項	191
ステップ 5: 初期画面の表示	146	継続入力フィールドの DBCS データを戻す方法	192
別のディスプレイ・ファイルでのウィンドウの定 義	147	継続入力フィールドでのキーボード機能	192
DDS のフルスクリーン・ディスプレイとウィ ンドウの定義	147	文字データ	192
RPG プログラム・ソース	147	フィールド・マーク	193
WINPGM の RPG プログラム・ソース	148	自動形状判別 (ASD) 処理	193
ステップ 1: 初期画面の表示	148	削除	193
ステップ 2: ウィンドウの表示	149	EOF 消去	194
ステップ 3: 初期画面に戻る	150	入力消去	194
第 6 章 画面用のグラフィカルな表示の作成	151	反転	194
グラフィカルな表示に影響を与える要素	151	クローズ	194
ハードウェア構成	151	フィールド・エグジット	194
拡張表示パラメーター	153	フィールド・プラス	194
DDS キーワード	154	フィールド・マイナス	195
メニュー・バーの作成	157	DUP (複写)	195
メニュー・バー選択項目の定義	157	漢字	195
メニュー・バー区切り記号の抑制	158	文字後退	195
メニュー・バー区切り記号の定義	158	文字前進	196
選択フィールドの概要	160	改行	196
選択フィールド用 DDS の例	161	フィールド前進	196
垂直単一項目選択フィールドの作成	162	フィールド後退	196
垂直複数項目選択フィールドの作成	163	前方フィールド出口処理	196
水平選択フィールドの作成	163	後方フィールド出口処理	197
垂直項目選択フィールドにおけるカーソル移動	164	メニュー・バーがアプリケーションと相互作用する 方法	197
水平項目選択フィールドにおけるカーソル移動	165	アプリケーション・レコードでの MNUBARDSP キーワードの定義	197
選択フィールド内の選択標識の制御	166	メニュー・バー・レコードでの MNUBARDSP キーワードの定義	199
単一項目選択フィールドを使用したプルダウン・メ ニューの作成	167	プルダウン・メニューからの入力データの受け取 り	200
プルダウン・メニュー内の選択標識の制御	169	プルダウン入力パラメーターを使用したプル ダウン・メニューからの入力データの受け取 り	200
アクセラレーター・キーの定義	170	入力データを受け取った後のプルダウン・メニ ューの除去	202
メニュー・バー切り替えキーの定義	171	表示前のプルダウン・メニューの更新	202
取り消しキーの定義	172	アプリケーション・ヘルプの定義	203
カーソルがプルダウン・メニューの外にある場合 の機能の制限	172	選択項目レベルのヘルプの定義	203
選択リストの概要	173	フィールド用ヘルプの定義	206
選択リスト用 DDS の例	174	メニュー・バーとプルダウン・メニューのキーの相 互作用	207
選択リストの作成	175	カーソル移動	208
選択リスト内の選択標識の制御	176	タブ・キーを押す	208
スクロール・バーの概要	178	カーソル・キーを押す	209
スクロール・バーの作成	179	プログラミングの例	209
スクロール・バー用 DDS の例	181	アプリケーション・レコードでの MNUBARDSP キーワードの使用	209
スクロール・バーの操作	182		
押しボタンの概要	182		
押しボタン用 DDS の例	183		
押しボタンの作成	183		

説明	211
メニュー・バー・レコードでの MNUBARDSP	
キーワードの使用	212
説明	213
画面の外観	213
単純ホット・スポット	214
コマンド・キー・エミュレーション	215
前ページおよび次ページ・キー・エミュレーション	215
プログラマブル・マウス・ボタン - 概要	216
ポインター装置イベント	216
戻される AID コード	217
プログラマブル・マウス・ボタン - 利点	217
プログラマブル・マウス・ボタンの操作	218
プログラマブル・マウス・ボタン - NWS の考慮事項	219
プログラマブル・マウス・ボタン - イベント処理の状態	219
プログラマブル・マウス・ボタン - イベント処理の優先順位	220
非シフト左ボタン押しイベントの処理	220
非シフト左ボタン・リリース・イベントの処理	222
非シフト左ボタン・ダブルクリック・イベントの処理	223
シフト左ボタン押しイベントの処理	223
シフト左ボタン・リリース・イベントの処理	224
シフト右ボタン押しイベントの処理	225
他のすべてのポインター装置イベントの処理	226
グリッド・ライン構造の概要	226
グリッド・ライン構造用 DDS の例	227
グリッド・ライン構造およびウィンドウ	228
グリッド・ライン構造におけるハードウェア要件	228
HTML タグの挿入	229
HTML フィールドのオーバーラップの解決	229
プログラミングの例	231
第 7 章 ディスプレイ・ファイルとディスプレイ・ファイル属性のオーバーライド	235
オーバーライドを使用するかしないかの判別	235
HLL プログラムでのファイル属性のオーバーライド	235
例	236
HLL プログラムでのファイル名のオーバーライド	237
例	237
HLL プログラムでのファイル名と属性の両方のオーバーライド	238
例	238
プログラムのコンパイル時のオーバーライドの適用	238
例	239
オーバーライドの削除	240
オーバーライドの表示	240
ファイルのリダイレクトを使用したファイル名およびライブラリーまたはファイル・タイプのオーバーライド	240

同じファイル・タイプでのファイルのオーバーライド	240
異なるファイル・タイプでのファイルのオーバーライド	240
オーバーライドを無視または制限するコマンドの認識	243

第 8 章 ディスプレイ・ファイルに関するメッセージとエラーの処理	247
独自のメッセージの作成および表示	247
メッセージ行にメッセージを表示する	248
サブファイル制御レコードが書き込まれる際にメッセージ行にメッセージを表示する	248
メッセージ・フィールドを使用してメッセージ行にメッセージを表示する	249
メッセージ行にメッセージを表示する場合の優先順位	249
ディスプレイ上のフィールドにメッセージを表示する	249
プログラム・メッセージ待ち行列中のメッセージを表示する	250
サブファイルを使用したエラー・メッセージの表示	250
メッセージに関するアラームの鳴動	253
ディスプレイ装置での永続的な入出力エラーの自動処理	253
システムから送信されるエラー・メッセージの分析	254
メッセージおよびメッセージ・モニターの説明	255
メジャー/マイナー戻りコードの説明	256
エラーからの回復	257
正常完了	257
例外での完了	257
永続的なシステム・エラーおよびファイル・エラー	258
入出力操作時の永続的な装置エラーまたはセッション・エラー	258
オープン操作または獲得操作時の装置エラーまたはセッション・エラー	259
入出力操作時の回復可能な装置エラーまたはセッション・エラー	259

第 9 章 ディスプレイ・ファイルを使用したメニューの作成およびアクセス	261
システム・メニューとユーザー定義メニューの実行	261
GO コマンドを実行後、メニューに戻る	261
直前のメニューの判別	261
メニューでの取り消しキーおよび終了キーの使用	262
サインオン時に表示されるメニューの選択	263
ユーザー独自のディスプレイ・ファイル・メニューの定義	263
メニューに関する DDS およびディスプレイ・ファイル考慮事項	264
メッセージ・ファイル内のメニュー・アクションの記述	265
メニュー用のヘルプ・フォーマットの命名	265
ディスプレイ・ファイル・メニューの作成	267

メニューおよびメニュー・ヘルプ情報を記述する	267	ディスプレイ・ファイルの 1 次画面サイズの指定	295
ディスプレイ・ファイルを作成する	268	サブファイル・レコードを一度に 1 ページだけ書き込む	295
メッセージ・ファイルを作成する	269	同じジョブのオープン・データ・パス (ODP) を共用する	296
メッセージをメッセージ・ファイルへ追加する	269	入力フィールドを持つレコードを、順番に画面に送信する	296
メニュー・オブジェクトを作成する	269	繰り返し送信されるレコードを重複させ、削除しない	296
メニューを実行する	270	画面の復元	297
ユーザー独自のプログラム・メニューの定義	270	コマンド機能キーではなくコマンド・アテンション・キーの定義	297
プログラム・メニューのパラメーターの引き渡し	270	送信勧誘操作の使用	297
プログラム・メニューの作成	271	ウィンドウの使用	298
メニューを記述する	271		
ディスプレイ・ファイルを作成する	271		
ソースを入力し、制御言語プログラムを作成する	271		
メニューを作成する	272		
メニューを実行する	272		
前のメニューに戻らずに、プログラム・メニューから出る	273		
プログラム 1	273		
プログラム 2	274		
メニュー名の競合の回避	275		
メニューの命名	275		
ライブラリー・リスト内の高位ライブラリーへのメニューの配置	275		
メニューを含むライブラリーの指定	275		
一般メニュー仕様の使用	275		
コマンド複製後のコマンド・デフォルトの変更	275		
メニュー属性の表示	276		
メニュー属性の変更	276		
メニューの削除	276		
第 10 章 ユーザー定義のデータ・ストリームの使用	277		
ディスプレイ装置の違いについて	277		
ユーザー定義のデータ・ストリームの制限について	278		
第 11 章 プログラム間でのデータの受け渡し	283		
ジョブ内の同じ経路指定ステップ内でのデータの受け渡し	283		
ジョブ内の経路指定ステップ内でのデータの受け渡し	284		
第 12 章 ディスプレイ・ファイル、ICF ファイル、およびデータ待ち行列からの入力待ち	287		
第 13 章 代替文字セットおよびコード・ページの使用	289		
システムに装置で通常表示されない文字がある	289		
装置がシステム上で表示されない文字を渡す	290		
フィールド用の文字変換を指定する	290		
ご使用のディスプレイ用の文字 ID (CHRID) 値の判別	292		
第 14 章 ディスプレイでのシステム・パフォーマンスの向上	295		
ディスプレイ・ファイルの書き込み操作の据え置き	295		

第 2 章 ディスプレイ・ファイル内での表示の定義

ディスプレイ・ファイルは、ディスプレイ装置に表示される情報のフォーマット、およびその情報がディスプレイ装置とのやり取りにおいてシステムによって処理される方法を定義します。データ記述仕様 (DDS) は、ディスプレイ・ファイルによって参照されるデータを記述します。

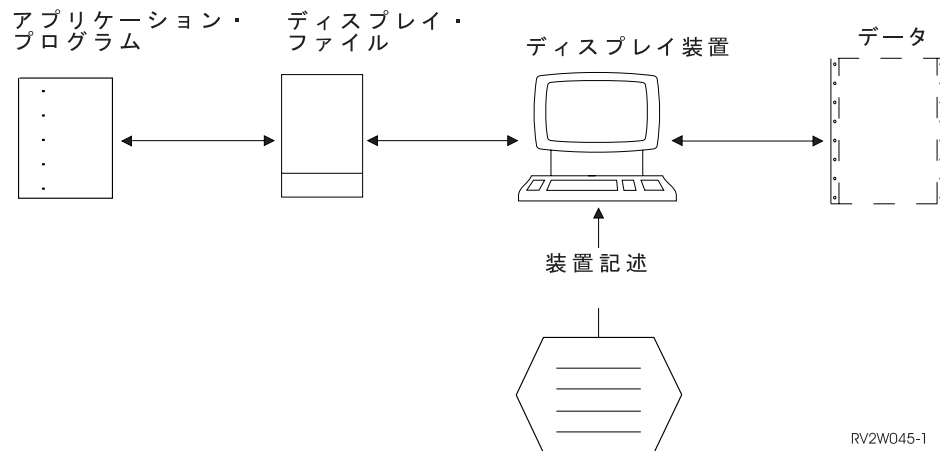
この章では、ディスプレイ・ファイルについて、それらの作成方法、およびそれらが表示を記述するための DDS ソースの提供方法などについて説明します。

ディスプレイ・ファイルの設定

ディスプレイ・ファイルは、システム上のオブジェクトです。**オブジェクト**は、それ自体を、およびディスプレイ・ファイルの場合はデータをも記述する、1 組の特性からなる、名前付きストレージ・スペースです。システム上の他のオブジェクトと同様、ディスプレイ・ファイルには以下の特性があります。

- ディスプレイ・ファイルは名前付きで、作成されるとライブラリーに置かれます。ファイル名およびライブラリー名によって、ご使用のアプリケーション内のディスプレイ・ファイルを参照できます。
- ディスプレイ・ファイルの作成後は、その変更、保護、保管、復元、削除を行えます。

アプリケーション・プログラムがディスプレイ装置と連動できるようにするには、その前に、ディスプレイ・ファイルを開き、プログラムとディスプレイ装置間をデータがフローできるようにしなければなりません。



装置記述は、ディスプレイ装置をシステムに記述するシステム・オブジェクトです。これは、ディスプレイ装置に関しても存在しなければなりません。装置記述には、装置アドレス、装置タイプ、型式番号、および機能などの情報が含まれます。装置記述は通常、システム担当者によって作成されます。または、ローカル接続された装置の場合は、システムの自動構成中に作成できます。

プログラムは、以下のいずれかを行うことによって、一度に複数のディスプレイ装置と連動できます。

- 複数のディスプレイ・ファイルを開く
- 複数のディスプレイ装置が 1 つのオープン・ファイルに接続するのを許可するディスプレイ・ファイルを開く

ディスプレイ・ファイルはそれ自身と一意的に関連したデータ集合を持たないので、データとディスプレイ・ファイルとの関係は、ディスプレイ・ファイルのオープン時に確立され、ディスプレイ・ファイルのクローズ時に終了します。

ファイル記述の決定

ファイル記述は、ディスプレイ・ファイルの作成と同時に作成され、ディスプレイ・ファイルの特性を記述し、ディスプレイ・ファイルが以下の事柄を行う仕方を決定します。

- ディスプレイ装置の制御
- ディスプレイ装置での表示用の、プログラムからの出力データのフォーマット設定
- プログラムへの表示用の、ディスプレイ装置からの入力データのフォーマット設定

ファイル記述は、プログラムがファイルをどのように使用できるかを決定します。プログラムがディスプレイ・ファイル記述と矛盾する操作を実行しようとした場合には、システムがその操作を許可しません。

ファイル記述は、それが記述するディスプレイ・ファイルと同時に作成および削除されます。ファイル記述のある部分は、ディスプレイ・ファイル変更 (CHGDSPF) コマンドによって永久的に、またはディスプレイ・ファイル・オーバーライド (OVRDSPF) コマンドによって一時的に変更できます。

ファイル記述は、以下の 3 つのレベルでデータを記述します。

- フィールド・レベル
- レコード・レベル
- ファイル・レベル

以下のセクションで、これらのレベルについて説明します。

フィールド・レベル記述

フィールドは、システムのデータ管理機能サポートによって認識および処理される、データの最小単位です。フィールド・レベル記述により、以下のような、フィールドの詳細な特性をシステムに知らせることができます。

- 画面上でフィールドの表示される位置
- フィールドに有効なデータ・タイプ
- フィールドが何らかの仕方で強調表示されるか
- プログラムからシステムへの出力、およびシステムからプログラムへの入力、それがどのように表示されるか
- 各フィールドのレコード先頭からの相対的な位置
- システム内での各フィールドの特性
- 各フィールドに関して、出力用のデータをどこから獲得するか
- ディスプレイ装置からの入力を、プログラムが使用できるように配置する場所と方法
- フィールドが入力可能フィールドか、それとも出力のみ可能なフィールドか

フィールド・レベル記述だけが、画面上の個々のフィールドに有効なデータが指定されていることを判別できます。

レコード・レベル記述

レコードは、1 つ以上のフィールドの順序セットです。レコード・レベル記述によって、特定のレコードの外観、またはそのレコード様式をシステムに知らせることができます。

レコード・レベル記述は、以下の 2 つの方法のうちの 1 つで指定されます。

- フィールド・レベル記述も使用される場合、レコード様式を構成するフィールド、およびレコード様式内でのこれらのフィールドの順序を識別します。その後システムは、レコード・レベル記述内のフィールド・レベル記述で記述された各フィールドに対して、個別の操作を実行できます。たとえば、1 つのフィールドを強調表示し、別のフィールドは強調表示しない、ということができます。
- フィールド・レベル記述が使用されない場合、レコード様式は、レコードの長さを指定することによって与えられます。システムはレコード全体を 1 つの単位として処理するので、レコードのある部分に対してはある方法で操作を実行し、別の部分に対しては別の方法で操作を実行する、ということはありません。

レコードは、システムとアプリケーション・プログラム間でデータを転送するために使用されるので、ディスプレイ・ファイルにはレコード・レベル記述が必要です。

ファイル・レベル記述

ファイルは、ゼロ個以上のレコードの編成された集合です (レコードがゼロ個のファイルは空です)。ファイル・レベル記述は、ファイルに全体として適用される記述です。ディスプレイ・ファイルの場合、ファイル・レベル記述で以下の事柄を指定できます。

- ディスプレイ・ファイルに対して有効なレコード様式
- ディスプレイ・ファイルで使用可能となるディスプレイ装置
- ディスプレイ・ファイルを介して入力されるデータに対して想定される図形文字セット

データの記述をプログラムの内部または外部のいずれで行うかの決定

ディスプレイ・ファイルおよびそれが参照するデータの詳細記述が、プログラムに組み込まれているのではなく、ディスプレイ・ファイルに含まれている場合、そのデータは**外部記述データ**と呼ばれます。データがソース・プログラム内に記述されている場合、そのデータは**プログラム記述データ**と呼ばれます。

外部記述データ

外部記述データは、そのファイルを使用するプログラムとは独立して存在します。外部記述データを使用すると、ディスプレイ・ファイルおよびそのディスプレイ・ファイルを介して処理できるデータの両方の、詳細記述および標準記述を作成できます。

外部記述データを使用するには、ディスプレイ・ファイルが外部記述ファイルとして使用されることを宣言する必要があります。言語コンパイラまたはインタープリターが、表示からファイル記述を抽出した後、それをプログラムに取り込みます。

外部記述データを使用すると、以下のような利点があります。

- プログラマーの生産性の増大。追加のコーディングをしなくても、言語がレコード・レイアウトを自動的に記述します。レコードおよびフィールドを記述する必要があるのは一度だけです (ファイルの作成時)。その後は、プログラム内でこれらのフィールドを参照できます。
- ファイルおよびプログラムの保守の容易さ。フィールドを追加、削除、または変更する場合、そのファイルを使用する各プログラム内のレコード・レイアウトを保守する代わりに、1 つの場所でそのことを行えます。
- データ保全性の増加。フィールドおよびレコードが 1 つの中央の場所で記述されるので、ファイル内のデータをプログラムに記述する際のプログラミング・エラーが生じる可能性が少なくなります。ファイルを使用するすべてのアプリケーション・プログラムは、データの同じ視点を持ちます。さらに、データのシステム視点が、アプリケーション・プログラム視点と同じになります。

- レベル検査の提供。レベル検査は、プログラムの実行時に使用される、プログラムの最後のコンパイル以降にファイル記述が変更されたかどうかを判別する自動方式です。変更のタイプによっては、プログラムを変更せず、再コンパイルするだけでよいこともあります。これによって、プログラム保守に対してより良い制御が行えます。22 ページの『ファイル記述の変更の検出』には、レベル検査機能についての詳細が記載されています。

プログラム記述データ

表示についてプログラム内で記述する場合、外部記述を使用する必要はありません。外部記述データを使用しない場合は、コンパイラまたはインタープリターに対してデータの外觀を定義する変数を、ソース・プログラム内で宣言しなければなりません。

プログラム記述データが使用される場合、プログラムとシステムは、データの同じ視点を持たない可能性があります。

- ファイルがフィールド・レベル記述を持たない場合、システムはレコード・レベルで作動しなければなりません。この場合、ただ一つ注意する点は、プログラムが使用しているレコード長が、システムが使用しているものと同じであることです。必ずしも同じである必要はありませんが、システムは常に、自分の持つレコード長で作動します。システムの使用しているレコード長が、プログラムの使用しているレコード長と異なる場合、システムは必要に応じて切り捨てまたは埋め込みを実行します。
- ファイルにフィールド・レベル記述があるが、プログラムはそれらを使用しない場合、プログラムが使用しなくてもシステムはフィールド・レベル記述を使用します。システムは、プログラムがファイル記述に従ってデータを表示することを期待します。また逆に、その記述に従ってデータをプログラムに提供します。

プログラム記述データについて詳しくは、42 ページの『プログラム記述データの使用』に記載されています。

ディスプレイ・ファイルと記述の作成

ディスプレイ・ファイルは、ディスプレイ・ファイル作成 (CRTDSPF) コマンドを使用して作成されます。

ディスプレイ・ファイルの DDS は、以下の 2 つの方法のうちの 1 つで定義できます。

- 画面設計機能 (SDA) ユーティリティーを使用する
- ユーザー独自に DDS ソースを入力する

ご使用のディスプレイ・ファイルについて特定の属性を指定できます。これらの属性については、以下に記載されています。

- 68 ページの『読み取り要求の実行時までの書き込み操作の据え置き』
- 85 ページの『前に表示した情報の保管』
- iSeries Information Center の **CL** のトピック
- *iSeries* 機密保護解説書

以下の例は、ディスプレイ・ファイルを作成する 2 つの方法を比較します。

Two Ways to Define Display Files

Enter your own DDS and then use the
Create Display File (CRTDSPF) Command ...

```

A
A
A      R GOTO                ALTHLP
A                                BLNK
A                                CFQ3(03)
A                                CF12(12)
A                                HELP
A                                HLPCLR
A                                1 32'Go To Another List'
A                                DSPATR(HI)
A      3 2'Select one of the following:'
A                                COLOR(BLU)
A      49 5 71'. Work With documents in folder'
A      50 6 712'. Work With documents to be print-
A      51 ed'
A      52 7 713'. Work With folders'
A      53 8 714'. Work With nontext document data'
A      9 715'. Work With text profiles'
A      20 2'Selection'
A      01 1A B 21 7
A                                DSPATR(RI PC)
A                                23 2'F3=Exit F12=Cancel'
A                                COLOR(BLU)
    
```



CRTDSPF command compiles DDS and creates
display file object.

Go To Another List

Select one of the following:

1. Work with documents in folder
2. Work with documents to be printed
3. Work with folders
4. Work with nontext document data
5. Work with text profiles

Selection

-

F3=Exit F12=Cancel

Design your display as it will look
using the screen design aid (SDA) utility ...

Go To Another List

Select one of the following:

1. Work with documents in folder
2. Work with documents to be printed
3. Work with folders
4. Work with nontext document data
5. Work with text profiles

Selection

-

F3=Exit F12=Cancel



SDA generates DDS and runs the CRTDSPF command.

```

A
A
A      R GOTO                ALTHLP
A                                BLNK
A                                CFQ3(03)
A                                CF12(12)
A                                HELP
A                                HLPCLR
A                                1 32'Go To Another List'
A                                DSPATR(HI)
A      3 2'Select one of the following:'
A                                COLOR(BLU)
A      49 5 71'. Work With documents in folder'
A      50 6 712'. Work With documents to be print-
A      51 ed'
A      52 7 713'. Work With folders'
A      53 8 714'. Work With nontext document data'
A      9 715'. Work With text profiles'
A      20 2'Selection'
A      01 1A B 21 7
A                                DSPATR(RI PC)
A                                23 2'F3=Exit F12=Cancel'
A                                COLOR(BLU)
    
```

RV2W000-5

2つの方法を結合することもできます。つまり、SDA を使用して初期表示画面を作成した後、生成された DDS を調整できます。

ファイル記述の変更

ディスプレイ・ファイルの作成後、ファイル記述を以下のように変更できます。

- 元々 CRTDSPF コマンドで指定されたファイル記述を変更するには、ディスプレイ・ファイル変更 (CHGDSPF) コマンドを使用します。
- DDS ソースに含まれるファイル・レベル、レコード・レベル、またはフィールド・レベルの情報を変更するには、まず最初に DDS ソースを更新した後、CRTDSPF コマンドを使用してディスプレイ・ファイルを再作成します。CRTDSPF コマンドで REPLACE(YES) を指定すれば、既存のディスプレイ・ファイルを削除しなくても、新規ディスプレイ・ファイルを作成できます。
- CL コマンドのファイル・レベル記述および DDS ソースの両方を変更するには、ディスプレイ・ファイルを再作成する際に、新規値を指定します。

ディスプレイ・ファイル記述の変更は、以下の事柄に従って適用されます。

- CL コマンドによってファイル・レベル記述が変更された場合は、ファイルを使用するプログラムが自動的に、それらの新規記述を使用ようになります。

- DDS 記述が変更され、プログラムがファイルをプログラム記述ファイルとして使用する場合には、システムは新規ファイル・レベル記述を使用します。しかし、DDS 記述が変更され、プログラムがファイルを外部記述ファイルとして使用する場合は、プログラムのコンパイル時に使用されるレコード・レベル記述およびフィールド・レベル記述が、変更後のファイルと一致しないことがあります。プログラムがファイルを開いた時点で、システムが不一致を検出した場合、エラーが生じます。『ファイル記述の変更の検出』を参照してください。

ディスプレイ・ファイルを開く時点で、ファイル・レベル記述を一時的に変更することもできます。これらの一時的な変更について詳しくは、235 ページの『第 7 章 ディスプレイ・ファイルとディスプレイ・ファイル属性のオーバーライド』を参照してください。

ファイル記述の変更の検出

外部記述ファイルを使用するプログラムのコンパイル時には、高水準言語コンパイラが、プログラムで参照するファイルのレコード・レベル記述およびフィールド・レベル記述を取り出し、それらの記述をコンパイルされるプログラムの一部とします。プログラムの実行時に、プログラムのコンパイル時に使用した記述が、現行の記述であることを確認できます。

システムは、各レコード様式と関連したファイルを作成する時に、レコード様式ごとに固有のレベル識別コードを割り当てます。システムは、以下の情報を使用して、レベル識別コードを決定します。

- レコード様式名
- フィールド名
- レコード様式の全長
- レコード様式内のフィールド数
- フィールド属性 (長さ、小数点以下の桁数など)
- レコード様式内のフィールドの配列

ディスプレイ・ファイルはまた、標識と呼ばれる特殊なフィールドの数および配列を使用して、レベル識別コードを判別できます。

レコード様式の DDS を変更し、また上記リスト内の項目のいずれかを変更した場合には、レベル識別コードが変更されます。

プログラムの実行時にレベル識別コード内の変更を確認するには、CRTDSPF または CHG DSPF コマンドの LVLCHK パラメーターに *YES を指定します。ディスプレイ・ファイルを開いた時点で、ディスプレイ・ファイルのレベル識別コードと、コンパイル済みプログラムの一部であるファイル記述のレベル識別コードとが、様式ごとに比較されます。識別コードが異なっている場合、またはプログラムに指定された様式のいずれかがファイルに存在していない場合には、状態を示すメッセージがプログラムに送信されます。

識別コードが異なっている場合、様式が変更されたか、またはご使用のプログラムが、変更された様式を使用していないかのいずれかです。変更された様式がご使用のプログラムに影響を与える場合、以下のいずれかを行うよう選択できます。

- プログラムを再コンパイルして、変更が組み込まれるようにする
- 取るべき処置を決定する前に、変更がご使用のプログラムに影響を与えるかどうかを判別する

レコード様式に対する変更を確認するには、以下のコマンドのいずれかを実行します。

- ファイル・フィールド記述表示 (DSPFFD) コマンド。これは、レコード・レベル記述およびフィールド・レベル記述を表示します。
- SEU 開始 (STRSEU) コマンド。これは、ファイルの DDS を含むソース・ファイルを表示します。

- ファイル記述表示 (DSPFD) または DSPFFD コマンド。これは、ファイル内で定義された様式レベル識別コードを表示します。
- プログラム参照表示 (DSPPGMREF) コマンド。これは、プログラムの作成時に使用された様式レベル識別コードを表示します。

レコード様式内での表示フィールドおよび機能の定義

ディスプレイ・ファイル内のレコード様式は、アプリケーション・プログラムで使用されるレコードの様式と、それが表示されるときレコードの様式の両方を記述します (24 ページの図 2 を参照)。

レコード様式には、フィールド記述が含まれます。これは、データ記述仕様 (DDS) を使用して定義されます。レコード様式内のそれぞれのフィールドごとに、以下を記述します。

- フィールドの画面上の位置
- フィールドの長さ
- フィールドに含まれるデータのタイプ (文字、ゾーン 10 進数、または浮動小数点)
- フィールド・タイプ (出力、入力、または入出力)

DDS キーワードについて

このセクションでは、DDS キーワードを使用して画面上に情報を記述する仕方を説明します。特定の DDS キーワードについて詳しくは、iSeries Information Center の『DDS』のトピックを参照してください。

ディスプレイ・ファイルの DDS

以下のソースは、サンプル・ディスプレイ・ファイルの DDS を示します。

```
|...+....1....+....2....+....3....+....:833.+....5....+....6....+....7...
A          R RECORD
A          3 2'Customer Number:'
A          CUST          5 0 3 20
A          3 27'Customer Name:'
A          NAME          20 3 44
A          4 27'Address:'
A          ADDR          20 4 44
A          CITY          20 5 44
A          STATE         2 5 66
A          ZIP           5 0 5 70
```

図 1. ディスプレイ・ファイルのサンプル DDS ソース

表 4 は、DDS 仕様の桁の位置と説明を示します。

表 4. サンプル DDS の桁の位置

列	定義	開始位置
17	名前のタイプ	17
19 - 28	フィールド名	19
30 - 34	長さ	34
36 - 37	小数点以下の桁数	37

表 4. サンプル DDS の桁の位置 (続き)

列	定義	開始位置
39 - 41	行位置	41
42 - 44	桁位置	44
45 - 80	機能	45

プログラムによって使用されるレコード様式

プログラムはレコード内のフィールドを、DDS ソースでそれらが記述されたのと同じ順序で渡します。

CUST	NAME	ADDR	CITY	STATE	ZIP
1	5 6	25 26	45 46	65 66	67 68
72					

RV2W028-1

画面上のレコード様式

フィールドは、ユーザーが DDS ソース内でそれらに割り当てた表示位置に従って表示されます。

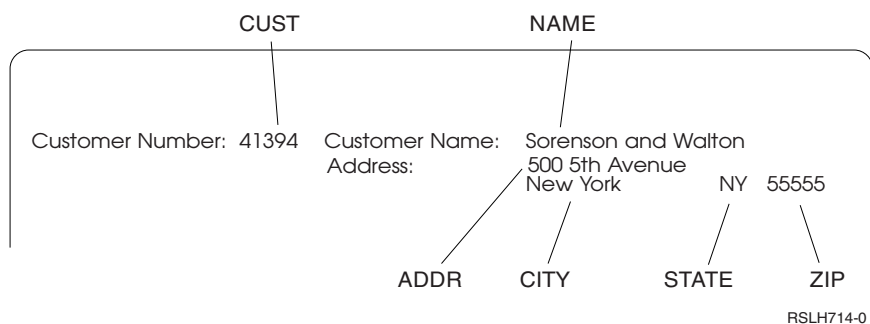


図 2. プログラム内および画面上のレコード様式

フィールド属性文字について

表示される各フィールドには、表示フィールドを定義する、関連した**開始属性文字**と**終了属性文字**があります。開始文字は、フィールドの先頭文字に先行し、空白として表示されます。終了属性文字は、フィールドの最後の文字の後に続き、これも空白として表示されます。たとえば、位置 2 から 8 のフィールドを指定する場合、開始属性文字は位置 1、終了属性文字は位置 9 になります。これらの文字は、ユーザーが DDS で指定するフィールド長には含まれません。開始属性文字は終了属性文字とオーバーラップできます。つまり、それらは表示上の同じ位置を占めることができます。ただし他のものは、開始属性文字にオーバーラップできません。したがって、表示を設計する際には、各フィールドの開始属性文字のために 1 スペース取らなければなりません。表示されるときにフィールド間に間隔を空けるため、空白の属性文字を使用できます。

フィールド・レベル記述が使用されない場合、レコード全体が、開始属性文字と終了属性文字のある 1 つのフィールドとして扱われます。

レコードの最後のフィールドが行の最後の位置で終了するようにレコードが表示される場合は、そのフィールドの終了属性文字は、次の行の最初の位置になります。次のレコードの最初のフィールドの開始属性文字は、終了属性文字と重ね合わせることができます。たとえばレコード 1 の最後のフィールドの終了属性文字が、行 5 の位置 1 にある場合に、レコード 2 の開始属性文字も、行 5 の位置 1 に置くことができます。この場合、最初のレコードはオーバーラップされているとみなされません。ただし、レコードの最初のフィールドが位置 1 で開始する場合、つまり開始属性文字が前の行の最後の位置にある場合には、前のレコードがオーバーラップされ、表示から消去されます。

プログラムで使用される入力レコードと出力レコード内のフィールドの位置を参照するには、CRTDSPF コマンドを使用してディスプレイ・ファイルを作成時に生成された、印刷された DDS 出力を参照してください。

レコード様式フィールドが使用される方法について

レコード様式で記述するフィールドは、以下の方法で使用できます。

注: DDS フォーム上の桁の位置を確認するには、23 ページの表 4 を参照してください。

- **入力フィールド**は、プログラムがレコードを読み取る時に、ディスプレイ装置からプログラムへ渡されるフィールドです。入力フィールドは、デフォルト値 (ディスプレイ・ファイルのレコード様式で指定される) で初期化できます。ユーザーがフィールドを変更せず、フィールドが入力のために選択されている場合には、デフォルト値がプログラムに渡されます。初期化されていない入力フィールドは、ブランクとして表示され、そこにユーザーはデータを入力できます。デフォルトで、画面上で入力フィールドには下線が引かれます。

注: 入力フィールド上の末尾ブランクは、ブランク文字ではなくヌルで置き換えられます。したがって、Insert キーを使用して、ブランクで終了するすべての入力フィールドに文字を挿入できます。

- **出力フィールド**は、プログラムがレコードを画面に書き込む時に、プログラムからディスプレイ装置に渡されるフィールドです。出力フィールドには、ユーザーではなくプログラムによって提供されるデータが含まれます。名前付き出力フィールドの初期値を指定するには、69 ページの『フィールドのデフォルト値の指定』を参照してください。

サブファイルの場合、これは情報のリストを表示するために使用される特殊レコードで、出力フィールドは、それらが出力/入力フィールドであるかのようにプログラムに戻されます。

- **出力/入力フィールド**は、プログラムがレコードを画面に書き込む時にプログラムから渡され、プログラムがレコードを画面から読み取り、フィールドが入力のために選択される時にプログラムへ渡されるフィールドです。デフォルトで、画面上でこれらのフィールドには下線が引かれます。出力/入力フィールドは通常、ユーザーが変更できるデータをプログラムが表示するときに使用されます。名前付き出力フィールドの初期値を指定するには、69 ページの『フィールドのデフォルト値の指定』を参照してください。
- **隠しフィールド**は、プログラム間で受け渡しされるものの、画面には送られないフィールドです。隠しフィールドは通常、サブファイルの関係するアプリケーションで有用です。たとえば、サブファイル・レコードには、隠しフィールド内のレコード・キー情報を含めることができます。隠しフィールドは、ユーザーは見ることができませんが、サブファイル・レコードとともにプログラムに戻され、プログラムがそのレコードをデータベースに戻すことができます。
- **固定フィールド**は、画面に渡されるものの、プログラムには不明なフィールドです。これらのフィールドには名前がなく、その固定値はファイルの DDS で定義されます。DATE、TIME、および MSGCON は、固定フィールドだけで許可されるキーワードの例です。それらの固定値は、プログラムの実行時 (DATE および TIME) または DDS のコンパイル時 (MSGCON) に判別されます。
- **メッセージ行**は、メッセージとして処理される出力フィールドです。

- **プログラム-システム間フィールド**は、名前付き、数値、または英数字の出力専用フィールドです。それらを使用して、アプリケーション・プログラムとシステム間で通信します。プログラム-システム間フィールドは、画面上に表示されません。つまり、ご使用のプログラムはこれらのフィールドにデータを入れることができ、システムはそのデータを使用して、出力操作での処理を制御しますが、ユーザーはこれらのフィールドの内容を見ることはできません。

フィールドは、入力フィールドまたは出力/入力フィールドであれば、入力可能です。それぞれの入力可能フィールドには、**変更データ・タグ (MDT)** と呼ばれる特殊属性があります。MDT はフィールドにデータが入力されると、ディスプレイ装置によってオンに設定されます。また、アプリケーション・プログラムによっても、オンに設定したりクリアしたりできます。

各レコード様式に指定できるフィールドの最大数は 32 763 です。(指定できる入力可能フィールドの数については、81 ページの『入力可能フィールドの数の制限についての理解』を参照してください。) レコード様式内のすべてのフィールドおよび標識を結合した最大長が 32 763 です。

次の画面は、ユーザーからの要求に対する応答で表示される、出力フィールドおよび入力フィールドを示しています (入力フィールドにカスタマー番号を入力するフォーム)。

```
Customer number: 41394
Order number:    41882
Order date:      11/01/81
Order amount:    $580.00
A/R balance:     $580.00
```

```
Enter next customer number: _____
```

プロンプト *Customer number:*、*Order number:*、*Order date:*、*Order amount:*、*A/R balance:*、および *Enter next customer number:* は固定情報です。これらのフィールドに関連したデータ (41394、41882、11/01/81、\$580.00、および \$580.00) は、出力フィールドに表示されます。データはアプリケーション・プログラムからシステムへ渡され、システムがそれを表示します。固定情報 *Enter next customer number:* の後に続くフィールドは、入力フィールドです。ユーザーはこのフィールドにデータを入力しなければなりません (カーソルは入力フィールドの先頭に置かれます)。入力フィールドには、デフォルトで下線が引かれます。フィールドの編集は通常、DDS 内で定義されます。

フィールドが隠しフィールド、メッセージ行、またはプログラム-システム間フィールドのとき以外、またはフィールドがサブファイル・メッセージ・レコード様式内にあるとき以外には、各フィールドの位置を指定しなければなりません。任意の行で開始できるレコードを定義するとき以外は、位置として行 1、桁 1 は指定できません。

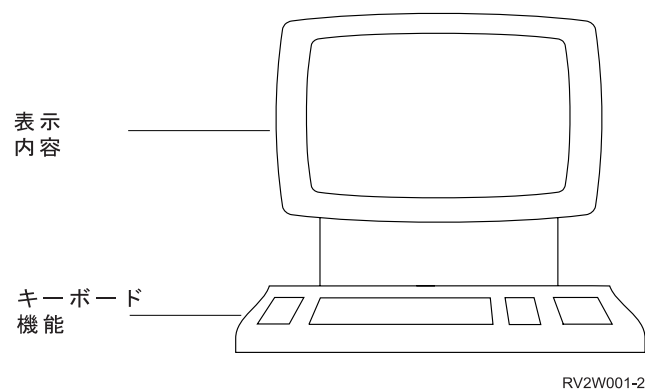
文字フィールドまたは数値 (ゾーン 10 進数) フィールドの最大長は、画面上の残り桁数 (フィールドの開始位置による相対数) から1 を引いたものです。数値 (ゾーン 10 進数) フィールドの別の制約事項は、31 より多い桁が画面上に残っている場合であっても、31 以下でなければならないことです。

記述するフィールドの仕様は、以前に記述したフィールドから取得できます。以前に記述したフィールドは、データベース・ファイル内にあるか、またはディスプレイ・ファイルの DDS ソース内ですでに定義されているかのいずれかです。データベース・ファイルからのフィールド・レベル記述を使用する場合、バイナリー・フィールドおよびパック 10 進数フィールドが、ゾーン 10 進数フィールドに変更されます。他のフィールドを定義するために使用するこれらのフィールドは、**参照フィールド**と呼ばれます。

画面上の同じ桁にある 2 つのフィールドを定義できます。その場合、オプション標識を使用して重複フィールドのどちらを表示するか選択できます。同じ出力操作で複数の重複フィールドが選択される場合、選択された最初のフィールドだけが表示されます。

機能キーの定義

ディスプレイ装置を使用してアプリケーションを書き込むには、キーボードのキーの機能および画面のコンテンツの両方を制御する必要があります。



Enter キーは常時、ユーザーが使用できます。ユーザーが他の機能キーを使用できるように、以下の DDS キーワードを指定して、対応する機能キーを使用可能にしなければなりません。

- CAnn (nn は 1-24)
- CFnn (nn は 1-24)
- CLEAR
- HELP (画面上でメッセージ・ヘルプを取得するためだけにヘルプ・キーが必要な場合には、不要)
- HOME
- PRINT
- ROLLDOWN または PAGEUP (サブファイルのページがサブファイルのサイズと等しくない場合に、サブファイルのロールを可能にするための指定は、不要)
- ROLLUP または PAGEDOWN (サブファイルのページがサブファイルのサイズと等しくない場合に、サブファイルのロールを可能にするための指定は、不要)
- MOUBTN (プログラマブル・マウス・ボタン) により、アテンション ID を各種ポインター装置イベントに関連付けることができます。
- PSHBTNFLD (押しボタン・フィールド) により、アテンション ID を押しボタンに関連付けることができます。

読み取り操作の実行時にどの機能キーが渡されるかを知らせるために、以下のうちのいずれか 1 つを使用して、機能キーを定義する必要があります。

- 機能キーの応答標識を定義します。**応答標識**は、アプリケーションに情報を戻す標識です。99 個の応答標識が使用可能です。

注: 応答標識は、機能キー以外に対しても使用されます。たとえば、それらを使用して、画面上のフィールド内のデータがいつ変更されたかを知らせることができます。

- 入出力フィールドバック域を調べます。**入出力フィールドバック域**は、開いたファイルに対して実行される操作に関して、システムによって提供される状況情報です。入出力フィールドバック域からの情報を取得する方法については、使用しているプログラミング言語のマニュアルを参照してください。フィールドバック域から入手できる情報については、721 ページの『付録 C. ディスプレイ・ファイルのフィールドバック域のレイアウト』を参照してください。

コマンド・アテンション (CAnn) およびコマンド機能 (CFnn) キーの定義

コマンド機能 (CFnn) キーおよびコマンド・アテンション (CAnn) キーは、1 から 24 の番号が付けられ、キーボード上では物理的に同じセットのキーです。これらのキーボード・キーは通常、*Cmdnn* または *PFnn* または *Fnn* というラベルが付けられます (nn は関連したキー番号)。それらを使用して、応答標識の設定や、特定の機能の実行を行います。

各種コマンド・キーは、以下の事柄を実行します。

コマンド機能 変更されたフィールドを含むレコードを、プログラムに戻します。

コマンド・アテンション

レコードはプログラムに戻されますが、そのレコードにはユーザーが入力したデータは含まれず、フィールド妥当性検査は実行されません。

あるファイル内で CFnn キーとして指定されたキーは、同時にそのファイル内で CAnn キーとして指定できません。同様に、CAnn キーとして指定されたキーは、同時に CFnn キーとして指定はできません。たとえば、機能キー 01 が CAnn キー (CA01) として指定された場合、同じファイル内のどこにおいても CF01 を指定できません。

応答標識が CFnn キーに対して指定された場合、そのキーが押されると、応答標識がオンに設定され、入力データと共にプログラムに渡されます。応答標識が CFnn キーに対して指定されない場合には、入力データだけが渡されます。

注: 入出力フィールドバック域には、1 文字のアテンション ID (AID) が含まれます。これは、押されたキーをも示します。入出力フィールドバック域については、721 ページの『付録 C. ディスプレイ・ファイルのフィールドバック域のレイアウト』を参照してください。

応答標識が CAnn キーに対して指定された場合、そのキーが押されると、応答標識がオンに設定され、プログラムに渡されます。画面に送られたフィールドおよび隠しフィールドが、プログラムに戻されます。CFnn キーまたは Enter キーが以前に押された場合には、入力専用フィールドが、以前に入力されたデータとして戻されます。データが入力専用フィールドに入力されなかった場合には、フィールドがブランク (文字フィールド) またはゼロ (数字フィールド) として戻されます。CFnn キーまたは Enter キーが最後に押されたとき以降、ユーザーが変更したフィールドは戻されません。

CAnn キーを使用すれば、プログラムの入力バッファに、ユーザーの入力した、ディスプレイ・ファイル内で指定された妥当性検査に適合しないデータを入れることができます。たとえば、ユーザーがデータを入力して CFnn キーまたは Enter キーを押すと、データがユーザーの DDS で定義されたように妥当性検査されます。入力データは一度に 1 つのフィールドずつ処理され、データ操作は妥当性検査の前に実行されます。妥当性検査エラーが生じた場合、メッセージが選択され、他のすべての入力データが処理されます。すべての入力データの処理後、1 つ以上のエラーが生じていた場合には、メッセージがユーザーに送信され

ます。その後、ユーザーが有効な CAnn キーを押すと、変更されていないデータが画面から送信されます。データが、入力バッファ保管域から入力バッファに移されます。入力バッファには現在、エラーのあるデータが含まれています。CAnn キーが押されたとき、ユーザーのプログラムがこのデータを処理中でない場合には、問題はありません。これが問題である場合には、CAnn キーの使用を避けてください。CFnn キーだけを使って、無効なデータを検出できるようにしてください。

CAnn キーを使用したい場合には、以下の妥当性検査 DDS キーワードを指定してはなりません。

CHECK(M10)
CHECK(M11)
CHECK(VN)
CHECK(VNE)
CHKMSGID
COMP/CMP
RANGE
VALUES

Print、Help、Clear、および Home キーは、CAnn キーと同様の仕方で作動します。Roll Up、Roll Down、Page Up、および Page Down キーは、CFnn キーと同様の仕方で作動します。

代替キーの指定

コマンド・アテンション・キーまたはコマンド機能キーが、Help、Page Up (または Roll Down) および Page Down (または Roll Up) の各キーの機能を実行するようにも定義できます。キーワードで指定される機能キーは、使用される代替キーを識別します。

DDS キーワードは以下のとおりです。

- **ALTHELP:** Help キーまたは ALTHELP キーワードで指定されたキーのいずれかが押されたときに、ヘルプ機能が開始されることを示します。ALTHELP キーワードが指定されているが、代替キーが指定されていない場合には、デフォルトが CA01 となります。Help キーはアテンション・キーであって機能キーではないことに注意してください。なぜならそれは入力に戻さないからです。
- **ALTPAGEUP** および **ALTPAGEDWN:** ページ・キーまたはキーワードで指定されたキーが押されたときに、ページ送り機能が開始されることを示します。ALTPAGEUP または ALTPAGEDWN キーワードで代替キーが指定されない場合には、デフォルトがそれぞれ CF07 および CF08 となります。ページ・キーが機能キーであることに注意してください。なぜなら、それらは入力に戻すからです。

ALTHELP、ALTPAGEUP、および ALTPAGEDWN キーワードで指定される代替キーは、実際のキーと同じ機能を提供します。たとえば、Help キーを押してヘルプ機能を開始する場合には、ALTHELP キーワードによって定義された代替キーを押しても、ヘルプ機能を開始します。同様に、Page Up または Page Down キーを押すと制御がアプリケーション・プログラムに戻る場合には、代替キーを押しても制御がアプリケーション・プログラムに戻ります。これらの例のいずれも、プログラムには、実際のキーが押されたかのように映ります。

ページ送り (USROPT(*ROLLKEY)) のユーザー・プロファイル・オプションは、PAGEUP、PAGEDOWN、ALTPAGEUP、ALTPAGEDWN、ROLLUP、および ROLLDOWN キーワードに適用されます。

代替ヘルプ・キー機能は、キーボードがロックされているときには作動しません。たとえば、入力可能でないフィールドに情報を入力すると、コントローラ検出エラーが生じ、明滅する数値が表示されます。

Help キーを使用して、エラーの詳細を取得できます。代替ヘルプ・キーとして指定される機能キーは、Reset キーが押されるまで無効です。押されると、ヘルプ情報がもはや使用できなくなります。

標識による情報の引き渡し

標識は、プログラムによって使用される入力レコードおよび出力レコード内、または特殊な標識域内のいずれかに存在する 1 文字のフィールドです。値 1 を持つ場合は標識がオンになり、値 0 を持つ場合にはオフになります。標識を使用して、プログラムからシステムへ、またはシステムからプログラムへ情報を渡すことができます。ディスプレイ・ファイル用の標識が DDS によって使用される方法を指定します。

ディスプレイ・ファイル用の標識には、2 つのタイプがあります。

オプション標識:

アプリケーション・プログラムからシステムへ情報を渡します。これらは一般的に、システムによる特定のレコード様式の処理を制御するために使用されます。

応答標識:

入力要求の完了時に、システムからアプリケーション・プログラムへ情報を渡します。応答標識は、ユーザーがどの機能キーを押したか、またはユーザーがどのデータを変更したかを、プログラムに知らせることができます。

オプション標識と応答標識のいずれも、ファイル・レベル、レコード様式レベル、およびフィールド・レベルで指定できます。ファイル・レベルで指定される標識は、ファイル内のすべてのレコード様式に適用されます。

オプション標識および応答標識をレコード域から除去する

標識域 (INDARA) キーワードを使用して、プログラムによって使用される入出力レコードから、オプション標識と応答標識を分離できます。INDARA キーワードを使用すると、標識が別個の 99 個の文字の領域に置かれます。この 99 個の文字域がどのように定義されているかについては、ご使用の適切な高水準言語の資料を参照してください。

同じ標識番号を応答標識とオプション標識の両方として使用する場合、応答標識の状況を使用して、後続の出力操作のためのオプション標識を設定できます。たとえば、標識 15 は、応答標識とオプション標識の両方として使用されます。入力操作がレコード様式に対して実行されるときに応答標識がオンの場合、オプション標識 15 がオンに設定され、そのレコード様式のための出力操作が実行されるときにオンになります。

ディスプレイ・ファイルに対して定義できるレコード様式の最大数は 1024 です。INDARA キーワードを使用しない場合、指定できるフィールドの最大数は、使用する標識の数 (各 1 文字) および記述する各フィールドの長さに依存しています。レコード様式内のすべてのフィールドおよび標識を結合した合計の長さが、32 763 文字を超えることはできません。INDARA キーワードを使用して、個別の標識域を指定する場合、指定できるフィールドの最大数は、各フィールドの長さだけに依存します。すべてのフィールドの総数は 32 763 を超えることはできません。

同時に異なる応答標識を使用可能にする

同時に表示されるレコード様式上の ROLLUP/ROLLODOWN キーワードに対して、異なる応答標識を持つことができます。たとえば、レコード A には 52 のロールアップ標識を指定し、レコード B には 25 のロールアップ標識を指定し、両方のレコードが表示されるとします。プログラムでレコード A に対して読み取り操作が要求されると、オペレーターは Roll Up キーを押し、制御がプログラムに戻ります。レコード A がプログラムに渡され、応答標識 52 がオンに設定されます。応答標識 25 は設定されません。その後、プログラムはレコード B に対する読み取り操作を実行できます。レコード B がプログラムに渡されると、応答標識 25 がオンに設定されます。応答標識 52 は設定されません。読み取り操作の対象となるレ

コード様式で指定された応答標識だけが、設定されます。 Roll Up キーが押されたときにカーソルが配置されていたレコード様式は、ROLL キーワードに関連した応答標識の設定に影響を与えません。

標識をオフに設定する

SETOF または SETOFF キーワードで指定された標識は、オフに設定される応答標識になり、プログラムに戻されます。入力操作が実行されるまで、標識はオフに設定されません。レコード様式内の任意の場所で、同じ標識が応答標識として指定される場合、標識は、関連したキーワード条件の状況に基づいて、プログラムに戻されます。たとえば、応答標識 01 が SETOF/SETOFF キーワードおよび CF5 キーの両方に対して指定される場合は、CF5 キーが押されると、標識 01 がオン条件に戻されます。標識が任意の場所で応答標識として指定される場合は、SETOF/SETOFF キーワードを使用する必要はありません。

メッセージ記述からの固定フィールド・テキストを挿入する

メッセージ固定情報 (MSGCON) キーワードを使用して、固定フィールドのテキストがメッセージ記述に含まれることを指定できます。

固定テキストに使用されるメッセージ記述が、画面上のフィールドよりも短い場合には、フィールドの残りの部分にブランクが埋め込まれます。メッセージ記述がフィールドより長い場合には、メッセージ記述が切り捨てられます。

DDS のコンパイル時にメッセージ記述が存在しない場合には、ファイルが作成されません。メッセージ記述を変更する場合、更新済みメッセージがディスプレイ・ファイルに含まれるようにしたいならば、ファイルを再作成する必要があります。

右から左へのカーソル移動の許可

カーソルがフィールド間、および入力フィールドで、画面上を右から左へ移動するようにできます。 DDS CHECK キーワードの 2 つのパラメーターを使用して、以下のことを行えます。

- CHECK (RL): 画面上の指定された非数値の入力フィールドまたはすべての非数値の入力フィールドで、カーソルを右から左に移動させます。
- CHECK (RLTB): フィールド間でカーソルを右から左に移動させます。

これらのパラメーターの使用時には、以下の事柄を覚えておいてください。

- モジュラス・チェック・ディジット検査がサポートされますが、チェック・ディジットは引き続き、フィールドの右端までのバイトです。
- 右から左へのカーソル移動が指定されるフィールドは、画面上の複数の行を占めることができます。ただし、カーソルは引き続き、画面の上から下へ移動します。
- ユーザー定義のデータ・ストリームでは、右から左へのカーソル移動は使用できません。

注: カーソル位置がディスプレイ・ファイル内で、またはプログラムによって指定されない場合は、カーソルは入力可能フィールドの一番上の行の左端に置かれます。

入力可能位置にのみカーソル移動を定義する

入力可能位置にのみカーソル移動 (CSRINPONLY) キーワードを使用して、カーソル移動を入力可能位置のみに制限します。このキーワードは、カーソル矢印キーにのみ影響を与えます。この機能は、画面上の、矢印キーの向きにある最初の入力可能位置にカーソルを移動させます。ユーザーは適切な方向にあるカーソル・キーを一度押すだけで、カーソルを入力可能位置に移動させることができます。

このキーワードは、ファイル・レベルまたはレコード・レベルで指定します。

カーソルが移動できる入力可能位置には、以下のものが含まれます。

- 入力フィールド (保護フィールド以外)
- 選択フィールドの選択項目 (その選択項目制御 (CHCCTL) 値のため、カーソルを置けないもの以外)
- 選択リストの選択項目 (その選択項目制御 (CHCCTL) 値のため、カーソルを置けないもの以外)
- メッセージ行 (メッセージが表示され、キーボードがロックされていない場合)
- サブファイル・メッセージ・レコード (SFLMSGRC) キーワードで定義されたメッセージ・サブファイル

いくつかの DDS キーワード (DSPATR(PC) および CSRLOC など) を使用して、カーソルを任意の表示位置に置くことができます。これは、CSRINPONLY キーワードが指定される場合にも当てはまります。最初の後続カーソル移動キー・ストロークにより、カーソルを置くことのできるロケーションにカーソルが移動します。カーソルを置くことのできる位置が画面上にない場合、カーソルは行 1、列 1 に置かれます。カーソルがこの位置から移動した場合、Home キーを押すと、カーソルは元の初期位置に戻ります。

入力フィールドのないウィンドウが表示された場合、カーソルはウィンドウの行 1、列 1 に置かれます。カーソル移動キーが押されると、カーソルは全画面の行 1、列 1 (ウィンドウの外部) に移動します。ウィンドウが *RSTCSR を指定して定義された場合には、コマンド・キーはウィンドウの外部では無効です。Home キーを押すと、カーソルがウィンドウに戻ります。任意のコマンド・キーまたは Enter キーを押すと、アラーム音が鳴り、カーソルがウィンドウに戻ります。この問題を回避するには、ウィンドウの上部の隅で入力禁止入力フィールドを指定するか、またはウィンドウ・キーワードで *NORSTCSR を指定することを考慮してください。

注:

1. メッセージ・サブファイルが 1 より大きい SFLPAG キーワードを指定して定義され、かつ CSRINPONLY キーワードが有効な場合、エラーのため反転表示になったフィールドは、メッセージ・サブファイルがメッセージの部分ページにロールされると、反転でないイメージになります。
2. エラーのため反転表示になったフィールドは、以下の条件が真の場合、反転でない表示になります。
 - メッセージ・サブファイルが 1 より大きい SFLPAG キーワードを指定して定義されている
 - CSRINPONLY キーワードが有効である
 - メッセージ・サブファイルがメッセージの部分ページにロールされる
3. PUTOVR、ERRMSG、または ERRMSGID キーワードを有効に指定してレコードが書き込まれると、CSRINPONLY キーワードの状態が変更されません。PUTOVR、ERRMSG、または ERRMSGID キーワードを指定した書き込み操作に先立ち、CSRINPONLY キーワードが有効な場合には、CSRINPONLY が有効なままとなります。このことは、PUTOVR、ERRMSG、または ERRMSGID キーワードを割り当てられたレコード上の CSRINPONLY キーワードのオプションに関係なく当てはまります。これはまた、CSRINPONLY キーワードを割り当てられたレコード上の PUTOVR、ERRMSG、または ERRMSGID キーワードのオプションにも関係なく当てはまります。
4. CSRINPONLY キーワードが有効な場合、PROTECT キーワードを指定したレコードの書き込みは、メッセージに関連した入力フィールドに影響しません。表示されるメッセージは、保護されません。したがって、カーソルは引き続きメッセージを移動します。
5. CSRINPONLY キーワードは、非プログラマブル・ワークステーションの拡張インターフェースをサポートするコントローラーに接続されたディスプレイ装置に対してのみ有効です。他のコントローラーに接続されたディスプレイ装置上では、無視されます。

入力フィールドのカーソル移動先の定義

FLDCSRPRG キーワードにより、ユーザーは、カーソルがフィールドを離れたときに次に移動するフィールドを指定できます。

フィールドの DDS は、以下のようになります。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          F1          10A B 3 4FLDCSRPRG(F3)
A          F2          10A B 13 4FLDCSRPRG(F1)
A          F3          10A B 16 4FLDCSRPRG(F2)
```

図3. フィールド・レベルのカーソル移動先の DDS

FLDCSRPRG キーワードのパラメーターは、前方フィールド出口処理の実行時にカーソルが移動するフィールドの名前です。フィールド出口キーのためカーソルが F1 を離れると、F3 に進みます。このキーワードで名前を付けられたフィールドのオプションがオフの場合には、このフィールドのカーソル移動先が無視されます。

注: カーソルが後方フィールド出口処理を使用してフィールドを離れる場合、カーソルは画面上の、FLDCSRPRG キーワードで指定された終了したフィールド名を持つ、最初のフィールドに移動します。図3の DDS の場合、後方フィールド出口処理を使用してフィールド 2 を離れると、カーソルはフィールド 3 に移動します。

SFLCSRPRG は、サブファイル・カーソル移動に使用されるキーワードです。サブファイルの DDS は、以下のようになります。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R SFL01          SFL
A          S1          10A B 5 5SFLCSRPRG
A          S2          10A B 5 25
A          R CTL01          SFLCTL(SFL01)
A          SFLPAG(5) SFLSIZ(20) SFLDSP
```

図4. サブファイル・レベルのカーソル移動先の DDS

SFLCSRPRG キーワードにより、カーソルは、あるサブファイル・レコード内のフィールドから、次に表示されるサブファイル・レコード内の同じフィールドに移動します。SFLCSRPRG の指定がなければ、カーソルはレコード内のフィールドから、同じレコード内の次のフィールドへ移動します。カーソルがサブファイルの最初のレコードのフィールド S1 を離れると、サブファイルの 2 番目のレコードの S1 に進みます。SFLCSRPRG キーワードの指定がなければ、カーソルは最初のレコードのフィールド S2 に進みます。カーソルが S2 を離れると、S2 は SFLCSRPRG キーワードを持たないため、次のレコードの S1 に進みます。このキーワードは、フィールド選択を使用するサブファイルでは指定できません。これは水平方向のサブファイルでは使用できません。表示される最後の SFL レコードの S1 上にカーソルがあるとき、カーソルは最後の SFL レコードの下にある次の入力フィールドに移動します。残りの SFL フィールドがない場合には、カーソルは画面の先頭に移動します。

注: 非プログラマブル・ワークステーションの拡張インターフェースをサポートしないコントローラーに接続されたディスプレイ上では、FLDCSRPRG キーワードと SFLCSRPRG キーワードは無視されます。

入力フィールドの属性の定義

入力フィールドの先行フィールド属性は、カーソルがフィールドに入った時点で、指定された属性に変更されます。

フィールドの DDS は、以下のようになります。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
  A          F1          10A B 3 4ENTFLDATR(*CURSOR (*DSPATR CS))
  A          F2          10A B 13 4ENTFLDATR(*NOCURSOR (*COLOR RED))
```

ENTFLDATR は、カーソルがフィールドに入った時点で、フィールドの属性を変更することをシステムに知らせます。*CURSOR および *NOCURSOR を使用して、カーソルがフィールドに入ったときにカーソルが可視になるかどうかを指定します。*NOCURSOR オプションを指定した場合、入出力フィールドバック域内のカーソルの行値および列値は、フィールド内の最初の位置を示しています。また、カラーや属性も指定できます。

注: 非プログラマブル・ワークステーションの拡張インターフェースをサポートしないコントローラーに接続されたディスプレイ上では、ENTFLDATR キーワードは無視されます。

編集マスクを使用した入力フィールドの保護

EDTMSK キーワードは、EDTCDE または EDTWRD キーワードを指定したフィールドに対して使用されます。フィールドが表示される時、フィールドの特定の領域が保護されます。どの領域が保護されるかを定義します。

フィールドの DDS は、以下のようになります。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
  A          F1          6 0B 3 4EDTWRD(' / / ')
  A          F1          6 0B 3 4EDTMSK(' & & ')
  A          F2          6 0B 4 4EDTCDE(Y)
  A          F2          6 0B 4 4EDTMSK(' & & ')
```

アンパーサンド (&) は、フィールドの保護される部分を表します。ブランクは、フィールドの無保護の部分を表します。編集マスクの長さは、フィールドの表示長に等しくなければなりません。無保護の位置の数は、少なくとも、フィールドのプログラム長に等しくなければなりません。フィールドが変更された場合、保護データは戻されないため、保護するのは非数値データだけにしなければなりません。&; がある箇所は、そのフィールドにどのデータがあるかに関わらず、その部分が保護されます。

最初のフィールドの日付内の斜線 (/) 文字が保護されています。2 番目のフィールドでは、日付内の / が常時保護されています。

非プログラマブル・ワークステーションの拡張インターフェースをサポートするコントローラーに接続されたディスプレイ上のキーボード機能は、編集マスク・フィールドに対するものと継続入力フィールドに対するものとで同じです。

注: 非プログラマブル・ワークステーションの拡張インターフェースをサポートしないコントローラーに接続されたディスプレイ装置上では、EDTMSK キーワードは無視されます。

右から左への表示処理の指定

右から左へ表示 (DSPRL) キーワードを使用して、ディスプレイ・ファイル内のレコードが右から左の方向に書き込まれることを指定できます。このキーワードは、ファイル・レベルでのみ許可されます。

図 5 は、DDS コーディングの例を示します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...  
A                                     DSPRL  
A          R RECORD  
A          FLD1          20A    5 5')Customer Name(:'
```

図 5. 右から左への表示処理の DDS 例

図 5 内の DDS は、画面上で次のような出力を生成します。

```
: (emaN remotsuC)
```

DDS で左右の括弧が逆になっていることに注意してください。これは、それらが画面上で正確に表示されるためです。すべての対称文字は、このように指定する必要があります。

ご使用のアプリケーション・プログラムが、DSPRL キーワードを指定した 1 つのディスプレイ・ファイルと、WINDOW キーワードを指定した別のディスプレイ・ファイルを使用する場合、WINDOW キーワードを指定したディスプレイ・ファイルが、必ず DSPRL キーワードをも指定するようにしてください。指定されていない場合、ディスプレイは WINDOW キーワードの指定されているディスプレイ・ファイルの方向を想定します。

DSPRL キーワードを指定する場合、データの入力時にカーソルは右から左へ移動します。したがって、CHECK(RL) キーワードを使用する必要はありません。CHECK(RL) および DSPRL を指定する場合、CHECK(RL) キーワードは無視されます。

DSPRL キーワードを指定すると、ディスプレイ・ファイル内のすべてのレコードが、右から左の方向に書き込まれます。個々のレコードが左から右の方向に書き込まれるよう指定することはできません。

ERRMSG または ERRMSGID キーワードを DSPRL キーワードとともに指定すると、これらのキーワードに関連したメッセージが、左から右の方向に表示されます。

フィールドの語の折り返しの指定

語の折り返しは、フィールド内の最後の語が、フィールドの右マージンを超える場合に、フィールド内の次の行に自動的に移動する機能です。名前付きフィールドに語の折り返し機能を指定するには、語の折り返し (WRDWRAP) キーワードを使用します。このキーワードは、ファイル・レベル、レコード・レベル、またはフィールド・レベルで使用できます。それを使用できるのは、入力専用 (I) フィールドまたは入出力 (B) フィールドに限定されます。

注:

1. この機能は、非プログラマブル・ワークステーションの拡張インターフェースをサポートするコントローラーに接続されたディスプレイ装置に対してのみ使用可能です。
2. Reverse キーおよび Close キーは、語の折り返しフィールドでは使用できません。
3. 語の折り返しを使用し、またキーボードが挿入モードであるとき、ヌル文字は右方にシフトしません。それらは置換されます。

語の折り返しは、以下のフィールドでは許可されません。

- DBCS 専用フィールド

- 純粋フィールド
- 択一フィールド (2 バイトを持つ)
- SBCS データを持つオープン・フィールド

語の折り返しは、以下のフィールド・タイプまたは機構とともに使用できません。

- 符号付き数値
- 数値のみ
- 数字のみ
- 磁気ストライプ読取装置 (DSPATR(OID))
- ライト・ペン (DSPATR(SP))
- 右揃え
- 全桁入力必須
- セルフ・チェック (M10F/M11F)
- 重複許可
- 右から左へのカーソル移動 (CHECK(RL))
- 右から左、上から下へのカーソル移動 (CHECK(RLTB))

語を分割しなければ、すべてのデータが語の折り返しフィールド内に収まらない場合には、そのフィールドの語の折り返し機能は無視されます。データは、語の折り返しが指定されなかったかのように書き込まれます。フィールドの後続の操作も、語の折り返しが指定されなかったかのように行われます。

語の折り返しは、単一行に含まれるフィールド上で指定できます。この場合、キーボードが挿入モードであれば、ヌル文字は右方にシフトしません。それらは置換されます。

フィールドの語の折り返しの指定 - ヒント

以下は、語の折り返し機能を使用するときには考慮すべき事柄です。

- 入力フィールドの全長には、折り返しが生じたときに埋め込みに使用される、行またはセグメントの終わりにある文字位置を含める必要があります。フィールドが短すぎると、エラーが生じるか、または語の折り返しがオフになります。
- 各行またはセグメントの長さは、フィールドに入力される可能性のある語のうち最も長いものの長さ以上でなければなりません。ある行またはセグメントが短すぎると、エラーが生じるか、または最後の行またはセグメントにデータがシフトダウンされる可能性があります。
- 折り返しが生じるように挿入される余分のブランクは、データがプログラムに戻されるときに除去されます。

フィールドの強調

ファイルの DDS 内で以下を指定することにより、画面上のレコードのフィールドを強調できます。

注: ご使用のディスプレイ装置でサポートされない機能は、無視されます。

表 5. フィールドを強調する DDS

強調のタイプ	DDS キーワード
フィールドに下線を引く (入力フィールドのデフォルト)	DSPATR(UL)
フィールド内の文字間に縦区切り線を置く	DSPATR(CS)

表 5. フィールドを強調する DDS (続き)

強調のタイプ	DDS キーワード
画面上で通常使用されるよりも高い輝度で表示することにより、フィールドを強調表示する	DSPATR(HI)
「暗い背景に明るい前景」から「明るい背景に暗い前景」へ、または「明るい背景に暗い前景」から「暗い背景に明るい前景」へ、フィールドのイメージを反転させる	DSPATR(RI)
フィールド内のデータを、ディスプレイ装置ユーザーに対して不可視にする	DSPATR(ND)
特定のフィールドにカーソルを置く	DSPATR(PC)
フィールドが表示されるとき、明滅させる	DSPATR(BL)

フィールドの属性を指定する別の方法は、DSPATR キーワードでプログラム-システム間フィールド・パラメーターを使用することです。ご使用のアプリケーション・プログラムは、プログラム-システム間フィールドを使用して、DSPATR キーワードが適用されるフィールドの表示属性または保護属性を設定します。

図 6 に、プログラム-システム間フィールドの DDS コーディングの例を示します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R RECORD
A          FLD1          5A    2  2DSPATR(&PFLD1)
A          FLD2          5A    2  9DSPATR(&PFLD2)
A          PFLD1         1A    P
A          PFLD2         1A    P
```

図 6. プログラム-システム間フィールドの DDS 例

1 つのプログラム-システム間フィールドを、レコード内の複数のフィールドに対して使用できます。フィールドあたり 1 つのプログラム-システム間フィールドだけ使用できます。プログラム-システム間フィールドを使用している場合、以下の属性は指定できません。

MDT 表示されるとき、変更データ・タグを設定する

OID オペレーター識別番号

PC 位置カーソル

SP ライト・ペンによる選択

ご使用のプログラムがプログラム-システム間フィールドに渡すことのできる有効な 16 進値については、iSeries Information Center の **DDS** のトピックを参照してください。

カラーの追加

カラーを表示するディスプレイ装置上で使用するために、画面を設計できます。DDS キーワード **COLOR** によって、フィールドに対して次のカラーを指定できます。すなわち、緑、白、赤、青緑、黄、ピンク、および青。カラーをサポートしないディスプレイ装置上で表示されるフィールドに対して選択された場合、このキーワードは無視されます。

ディスプレイ・ファイルの DDS で **COLOR** キーワードが指定されないが、ディスプレイ装置はディスプレイ装置記述でカラー・ディスプレイ装置として指定されている場合、カラーをサポートしないディスプレイ装置用に設計された画面は、カラー・ディスプレイ装置にも使用できます。キーワード **DSPATR(UL)** および **DSPATR(RI)** は、別個のフィールドで指定されている場合、5250 ディスプレイ装置に対するのと同様に機能します。ただしキーワード **DSPATR(CS)**、**DSPATR(HI)**、および **DSPATR(BL)** はカラー・ディスプレイ装置上で以下のカラーを生成します (指定された表示属性 **CS**、**HI**、および **BL** が抑制されます)。

COLOR キーワードが指定されない場合に、カラー・ディスプレイ装置上で生成されるカラー	選択される表示属性: DSPATR(CS)	選択される表示属性: DSPATR(HI)	選択される表示属性: DSPATR(BL)
緑 (通常)			
青緑	X		
白		X	
赤、明滅しない			X
赤、明滅する		X	X
黄	X	X	
ピンク	X		X
青	X	X	X

出力フィールドの編集

システムには、フィールドの表示時にさらに読みやすくするための、編集サポートが用意されています。システム編集サポートを使用すれば、次のことができます。

- 先行ゼロを消去する。
- 小数桁を示したり、数字を 3 桁ずつ区切ったりするために、コンマおよびピリオドを句読点としてフィールドに入れる。
- 左側または右側に負符号 (-) を付けて、負の値を表示する。
- 右側に文字 CR (クレジット) を付けて、負の値を表示する。
- ゼロの値を、ゼロまたはブランクとして表示する。
- 有効数字の左側にアスタリスクを表示して、アスタリスク保護を行う。
- システム値の QCURSYM と対応する、通貨記号を表示する。

システムは、この編集サポートとともに編集コードと編集語を提供します。**編集コード**とは、定義済みの 1 組の編集パターンです。システムによって提供されたものに加えて、ユーザー独自の編集コードも定義できます。編集コードは名前によって識別され、システムは、指定された編集コードによって定義されたパターンに従ってフィールドを編集します。**編集語**とは、必要な結果を作成するために自分で定義する編集パターンです。編集コードは、一般に使用される編集要件をカバーしています。編集語サポートが必要になるのは、編集コードがカバーしていない編集要件に対してだけです。

編集コードは次のように使用されます。

- プログラム記述データを使用しているアプリケーションでは、高水準言語により編集コードを指し示すか、あるいは独自の編集語を作成することができます。
- アプリケーションが外部記述データを使用している場合には、編集コード (EDTCDE) の DDS キーワードを使用すれば、編集コードを識別することができ、編集語 (EDTWRD) の DDS キーワードを使用すれば、独自の編集パターンを定義できます。

システムは、いくつかの編集コードを提供します。これらのコードによって定義される編集パターンは、『付録 E. 編集コード』に記載されています。

ユーザー独自の編集コードの定義

5 つの編集コードを定義して、i5/OS 編集コードで使用できる機能よりも多くの編集機能を提供し、それが無い場合には編集語を使用する必要がある共通した編集機能进行处理できます。これらをユーザー定義編集コ

ードと呼びます。たとえば、ハイフンの組み込まれた数値 (たとえば、電話番号)、あるいは複数の小数点を編集することが必要になることがあります。これらの機能に、ユーザー定義の編集コードを使用することができます。これらの編集コードは QEDIT5、QEDIT6、QEDIT7、QEDIT8 および QEDIT9 という名前をもち、DDS または高水準言語プログラムの中では番号 (5、6、7、8、または 9) によって参照できます。

これらの編集コードは、編集記述作成 (CRTE DTD) コマンドを使用して作成されます。編集記述は、常にライブラリー QSYS に入れられます。編集記述の移動または名前の変更はできず、それぞれの出現は、1 回だけ許されています。編集記述のオブジェクト・タイプは、*EDTD です。

それらはユーザー定義の編集コードですが、それらの各バージョンはご使用のシステムによって提供されます。これらの編集記述をそのまま使用したり、あるいは削除して独自のものを作成することもできます。これらの編集記述の IBM 提供のバージョンによって実行される編集、またユーザー定義の編集コードの内容の定義およびそれらを使用するため規則は、『付録 E. 編集コード』に記載されています。

ユーザー定義の編集コードを使用する前に、システム上のその内容を調べる必要があります。IBM 提供のバージョンが変更されている場合もあるからです。編集記述表示 (DSPEDTD) コマンドは、ユーザー定義の編集コードの内容を表示します。

ユーザー定義の編集コード記述を変更しても、その編集記述を使用してすでに作成されたアプリケーションまたはディスプレイ・ファイルに影響はありません。アプリケーションに変更後の編集記述を使用させたい場合には、再度高水準言語プログラムを作成するか (編集コードがプログラムで参照される場合)、あるいは再度ファイルを作成しなければなりません (EDTCDE キーワードが入っている外部記述ファイルをアプリケーションが使用している場合)。

有効な画面サイズの指定

いくつかのケースでは、以下の画面サイズ条件名を使用して、画面サイズを基にした表示ロケーションおよびキーワードを選択できます。

- *DS3、24 x 80 (5251 モデル 11 および 12、5291、5292、3179 モデル 2、3180-2、3196、および 3197)
- *DS4、27 x 132 (3180-2; 3197 モデル D1、D2、W1、W2; 3477 モデル FA、FC、FD、FE、FG、FW; 3487 モデル HA、HG、HW、HC)

注: 27 x 132 モードで表示する機能は、6040 または 6041 または 2638 ローカル・ディスプレイ装置コントローラーに接続された、またはリモート側で 5294 または 5394 コントローラーに接続された、3180-2、3197、3477 モデル FA、FC、FD、FE、FG、FW、および 3487 モデル HE、HD、HW、HC ディスプレイ装置上で使用できます。27 x 132 モードの画面サイズは、これらのコントローラーが指定されるのでない限り、DSPSIZ キーワードに対して無視されます。

これらの条件名を使用して、異なるサイズの画面上の異なるロケーションにフィールドを置くことができます。ただしフィールドは、どのサイズの画面上でも同じ順序で指定しなければなりません。たとえば、以下の DDS は 24 x 80 と 27 x 132 の両方の画面をフォーマット設定します。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     DSPSIZ(*DS3 *DS4)
A                                     .
A                                     .
A                                     .
A      R RECORD
A      NAME          20      5  2
A *DS4
A      ADDR          30      6  2
A *DS4
A                                     4102

```

図7. 2 つの画面サイズ用の DDS 例

通常、ディスプレイ・ファイルは 24 x 80 画面 (デフォルトのサイズ) 用にセットアップされます。DSPSIZ キーワードは、ファイルに対して有効な画面サイズを指定し、どのサイズが 1 次および 2 次の画面サイズであるかを示します。(1 次画面サイズは、最初の、またはただ一つの DSPSIZ 値です。)
DSPSIZ キーワード上で、画面サイズは *DS3、*DS4、24 80、または 27 132 として指定できます。たとえば、DSPSIZ (24 80) は 24 x 80 の画面サイズを指定します。1 次および 2 次画面サイズを指定すると、ディスプレイ・ファイルが両方のサイズに対して妥当性検査されます。

1 次画面サイズとして指定された画面サイズは、ディスプレイ・ファイルがそれと共に最も頻繁に使用されるものである必要があります。このような方法で DSPSIZ キーワードをコーディングすることによって、パフォーマンスが向上します。実際の画面サイズが 2 次画面サイズのとときには、追加処理が実行されま

す。
画面サイズ条件名を使用すれば、任意のサイズの画面に対して、単一のディスプレイ・ファイルを使用できるようにします。たとえばサブファイルを使用すれば、24 x 80 画面用にページあたり 22 のレコード、または 27 x 132 画面用にページあたり 25 のレコードを指定できます。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     DSPSIZ(*DS4 *DS3)
A                                     .
A                                     .
A                                     .
A      SFLPAG(25)
A *DS3      SFLPAG(22)
A                                     .
A                                     .
A                                     .

```

図8. 2 つの画面サイズ用のサブファイル用の DDS 例

ユーザー独自の画面サイズ条件名も定義できます。

画面を印刷できるようにする

Print キーが画面に対して使用可能である場合は、ユーザーは Print キーを押すことにより、現行画面を印刷できます。DDS PRINT キーワードに指定するパラメーター (または指定しないパラメーター) は、ユーザーの画面が印刷される仕方を制御します。

表 6. Print キーを使用する PRINT キーワードの結果

パラメーター	処置
パラメーターなし	出力は、このディスプレイに関連したディスプレイ装置プリンターに送られます (表示装置記述作成 (CRTDEV DSP) または表示装置記述変更 (CHGDEV DSP) の PRINTER パラメーター)。ディスプレイ装置プリンターに対する操作が失敗した場合、またはディスプレイ装置プリンターが指定されていない場合には、出力はディスプレイ装置記述の PRTFILE パラメーターで指定されたプリンター・ファイルに送られます。PRTFILE パラメーターのデフォルトは QSYSPRT です。
ファイル名および、オプションでライブラリー名	印刷操作は指定されたプリンター・ファイルに送られます。操作が失敗すると、デフォルトのプリンター・ファイルに送られます。デフォルトのプリンター・ファイルは、CRTDEV DSP または CHGDEV DSP コマンドの PRTFILE パラメーターで指定されます。
応答標識 *PGM	応答標識がオンに設定され、制御がプログラムに戻ります。制御がプログラムに戻り、プログラムは入出力フィードバック域内のアテンション ID を調べて、どのキーが渡されたかを判別しなければなりません。

PRINT キーワードは、ファイル・レベルで、またレコード・レベルでも使用できます。PRINT がレコード・レベルで指定されると、異なる形式の PRINT キーワードを持つ (または PRINT キーワードのない) いくつかのレコードが、画面上に同時に表示されることがあります。画面に書き込まれた最後のレコード形式が、画面全体に対する Print キーの使用を制御します。

任意の形式で PRINT キーワードを指定すると、ユーザーはメッセージ・ヘルプを含む画面を印刷できません。

PRTKEYFMT パラメーターにより、画面の印刷時にどの情報を含めるかを制御できます。

- 出力のみ
- 出力とヘッダー情報 (行と列)
- 出力と枠情報 (システム名を含むタイトル行、日時、フォーマット済みユーザーおよびディスプレイ装置名)
- 出力と枠情報とヘッダー情報

ジョブ変更 (CHGJOB) およびジョブ属性検索 (RTVJOBA) コマンドの PRTKEYFMT パラメーターにより、Print キー出力の外観を選択できます。

ディスプレイ装置プリンターの装置記述を変更する場合は (CHGDEV PRT コマンド、または DLTDEV D および CRTDEV PRT コマンドを使用)、CHGDEV DSP コマンドの PRINTER パラメーターを使用して、関連したディスプレイ装置の装置記述も変更する必要があります。これは、装置記述を変更したプリンター名が同じままの場合でも行う必要があります。

ウィンドウの定義

ユーザーのデータ入力を支援するため、画面上でウィンドウを使用できるアプリケーションがあります。

ウィンドウは、現行画面の一部にオーバーレイする情報で、ユーザーはウィンドウ内部の情報を読み取ることができます。画面の残りの部分はウィンドウによってオーバーレイされず、引き続きユーザーは読み取ることができます。

標準 DDS を使用することによって、またはユーザー定義のデータ・ストリームを使用することによって、ウィンドウを作成できます。標準 DDS を使用するには、127 ページの『第 5 章 ディスプレイ・ファイルを使用したウィンドウの定義』を参照してください。QUSRTOOL ライブラリーには例も用意されています。

プログラム記述データの使用

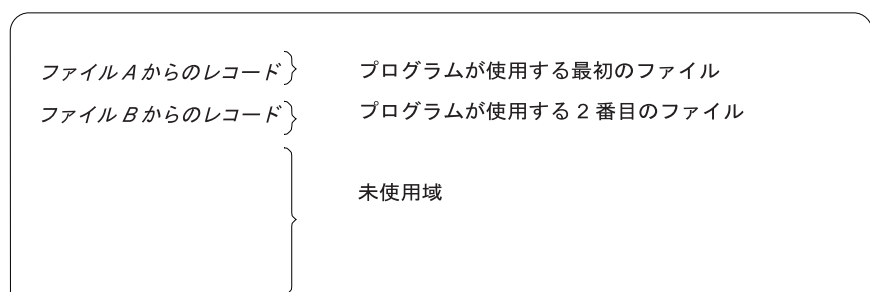
データ記述仕様を使用しなくても、ディスプレイ・ファイルを作成できます。その場合、そのようなディスプレイ・ファイルはプログラム記述データを使用し、レコードやそれ独自のフィールド・レベル記述はありません。

ディスプレイ・ファイルでプログラム記述データを使用して、1 つ以上のディスプレイ装置とのコミュニケーションを行うときには、単純な表示フォーマットだけが実行でき、そのフォーマットはファイルを使用している高水準言語プログラム内で指定しなければなりません。すべてのフィールド記述が定義され、すべての処理が、ファイルを使用するプログラム内で実行されます。同じディスプレイ装置に対して同じプログラム内で同時に複数のディスプレイ・ファイルを開くことができますが、同じディスプレイ装置上で同時に使用できるのは 2 つだけです。つまり、1 つは入力用、1 つは出力用です。

プログラム記述データを使用するディスプレイ・ファイルが開いているとき、システムは画面上の領域を、単一フィールドとして扱います。つまり、フィールド長がレコード長と同じです。レコード長は、ファイルを使用しているプログラムによって定義され、ファイルが開いたときからクローズするまで、同じです。レコードがプログラムからディスプレイ装置に、またはディスプレイ装置からプログラムに渡される時、標識は渡すことはできません。またコマンド・キーは、プログラム記述ディスプレイ・ファイルに対して使用できません。

画面上のスペースは、以下の例で示されたように、プログラム記述ディスプレイ・ファイルに割り当てられます。

プログラムによって使用される最初のファイルのレコードは、画面の先頭 (上部) に表示されます。2 番目のファイルのレコードは、最初のファイルのレコードによって使用される領域のすぐ下に表示されます。



RSLH705-0

ファイル B のレコードは、ファイル A のレコードの最後の行の次の、最初の新たな行の先頭から開始します。ファイル A のレコードが最後の行を完全に埋めていない場合、そのスペースはどちらのレコードによっても使用されず、最大レコード長の計算時に考慮に入られます。プログラム記述ディスプレイ・ファイルでは、最大レコード長は以下のようになります。

- 入力ファイルの場合、画面サイズ - 2
- 出力ファイルの場合、画面サイズ - 2
- 入出力ファイルの場合、画面サイズ - 2
- 2 つのファイル (1 つは出力、1 つは入力) の場合、画面サイズ - 3

プログラム記述ディスプレイ・ファイルが開いているとき、それは次のように定義できます。

- 入力専用
- 出力専用
- 入出力

入力専用ファイルの定義

入力専用ファイルを開くと、レコードは画面上で単一のブランク・フィールドに初期化されます。カーソルがフィールドの先頭位置に置かれると、ユーザーは任意のタイプのデータを入力できます。

プログラムがレコードを読み取るとき、入力はプログラムに渡されます。レコードは画面から消去されません。カーソルを再びレコード (フィールド) の先頭位置に置くと、プログラムが次のレコードを読み取るとき、キーボードがアンロックされます。その後ユーザーは、前のレコードの上に次のレコードを入力できます。

出力専用ファイルの定義

出力専用ファイルを開くと、レコードは画面上で単一のブランク・フィールドに初期化されます。プログラムがレコードをファイルに書き込むと、レコードが表示され、キーボードがロックされます。ユーザーが Enter キーを押さなければ、別のレコードをファイルに書き込むことができません。ファイルに書き込む後続のレコードは、現在表示されているレコードを消去します。なぜなら、出力ファイルに表示できるレコードは 1 つだけだからです。

入出力ファイルの定義

入出力ファイルを開くと、レコードは画面上で単一のブランク・フィールドに初期化されます。カーソルがファイルの先頭位置に置かれると、ユーザーは任意の種類 of データを入力できます。

ファイルを使用しているプログラムは、任意の順序でレコードの読み取りや書き込みを行えます。レコードがファイルに書き込まれるときには必ず、変更データ・タグがオフに設定され (データがフィールドに入力されなかったことを示す)、キーボードはアンロックされます。その後、ユーザーがデータをフィールドに入力すると、変更データ・タグがオンに設定されます。次の操作が読み取り操作ではなく書き込み操作である場合には、ユーザーによって入力されたデータが上書きされ、変更データ・タグが再びオフに設定されます。

第 3 章 アプリケーションでのディスプレイ・ファイルの処理

ディスプレイ・ファイルをアプリケーションで使用するには、最初にディスプレイ・ファイルを定義する必要があります。この章では、ディスプレイ・ファイルをアプリケーションで使用する場合に実行される操作について説明します。

システムがリソースを割り振る方法についての理解

高水準言語プログラムがディスプレイ・ファイルを使用する場合、実行に必要なリソースをシステムが割り振らなければならない操作がいくつかあります。ファイル・リソースを割り振ると、ディスプレイ・ファイルが開かれるときにシステムがディスプレイ・ファイルに対してファイルのロックを取得します。したがって、複数のユーザーが同じディスプレイ・ファイルを競合して使用することはできません。例えば、アプリケーションがファイルを使用中の場合は、そのファイルを削除できません。

ファイル・リソースは、以下の 2 つの方法で割り振られます。

- システムは、割り振りを必要とする操作が要求されるたびに割り振りを実行します。ディスプレイ・ファイルに対する以下の操作は、割り振りを必要とします。
 - オープン: ファイル・リソースには、ファイル記述およびディスプレイ装置が含まれます。オープン操作の詳細については、46 ページの『ディスプレイ・ファイルのオープン』を参照してください。
 - 獲得: ディスプレイ装置がリソースとして割り振られます。獲得操作の詳細については、46 ページの『入出力操作のディスプレイ装置の獲得』を参照してください。
 - リモート・システムでのプログラムの開始: ファイル・リソースには、APPC/APPN で必要なセッション・リソースが含まれます。
- プログラムを実行する前にプログラムに必要なすべてのリソースを使用可能にする場合は、プログラムを実行する前にジョブでオブジェクト割り振り (ALCOBJ) CL コマンドを使用します。

リソースの割り振り中にリソースが即時に使用可能にならない場合、システムは事前定義した時間待機します。制限時間内にリソースが使用可能にならない場合は、以下のようになります。

- ALCOBJ コマンドを使用している場合、コマンドは失敗します。
- プログラムがファイル操作を実行中の場合、その操作は失敗し、エラー・メッセージがプログラム・メッセージ待ち行列に送信されます。

高水準言語のエラー処理機能を使用して、操作を再試行してみてください。例えば、別のジョブがファイルに関連付けられたディスプレイ装置を使用しているためにオープン操作が失敗する場合は、後でディスプレイ装置が使用されなくなってからオープン操作を再試行してみてください。

リソースの割り振り中にシステムが待機する時間は、ALCOBJ コマンドで指定するか、CRTDSPF コマンドの WAITFILE パラメーターで指定します。プログラムを実行する前に ALCOBJ コマンドを使用する場合、リソースが使用可能であるため WAITFILE パラメーターの値は無視されます。ディスプレイ・ファイルで発生するディスプレイ装置エラー用のエラー処理手順がアプリケーションに用意されている場合は、*IMMED 以外の値を指定してシステムがディスプレイ装置エラーから回復できるようにする必要があります。ディスプレイ装置のシステム回復手順を完了してからでなければ、プログラムはリソースの割り振りに起因するエラーから回復できません。

ディスプレイ・ファイルのオープン

オープン操作により、ディスプレイ・ファイルがプログラムに接続されて処理できるようになります。

ディスプレイ・ファイルが開かれると、通常は、ディスプレイ・ファイル用に 1 つ以上のディスプレイ装置が暗黙的に獲得されるか、入出力操作に備えて自動的に準備されます。

表 7. ディスプレイ・ファイルのオープン時に暗黙的に獲得されるディスプレイ装置

オープン時にディスプレイ・ファイルが定義される方法	他の指定内容	暗黙的に獲得されるディスプレイ装置
単一のディスプレイ装置で定義	*REQUESTER の指定あり	ユーザーがプログラムを要求したディスプレイ装置
	*REQUESTER の指定なし	CRTDSPF、CHGDSPF、または OVRDSPF コマンドの DEV パラメーターで指定するディスプレイ装置
複数のディスプレイ装置で定義	CL プログラムでオープン	CRTDSPF、CHGDSPF、または OVRDSPF コマンドの DEV パラメーターで指定するすべてのディスプレイ装置
	CL 以外の高水準言語でオープン	CRTDSPF、CHGDSPF、または OVRDSPF コマンドの DEV パラメーターで指定する最初のディスプレイ装置
ディスプレイ装置なしで定義	-	なし

CRTDSPF、CHGDSPF、または OVRDSPF コマンドの WAITFILE パラメーターで指定する値は、ファイル・リソースが使用可能になって割り振り可能になるまでオープン操作が待機する時間を決定するために使用されます。ファイル・リソース (ディスプレイ装置など) が使用可能にならない間に指定した待機時間が経過すると、オープン操作は失敗します。

ファイルのオープン時にディスプレイ装置が暗黙的に獲得されると、以下のようになります。

- 画面が完全に消去され、カーソルがディスプレイの左上隅に置かれる。
- キーボードがロック解除される。

オープン操作時にディスプレイ装置を暗黙的に獲得するには、オンに変更する必要があります。交換回線ディスプレイ装置が「オンに変更保留」の状態になっていれば、交換回線ディスプレイ装置も獲得することができます。また、*REQUESTER ディスプレイ装置以外のディスプレイ装置は、サインオンされている場合は獲得することができません。

入出力操作のディスプレイ装置の獲得

システムは、ディスプレイ・ファイルのオープン時にディスプレイ・ファイルに対してディスプレイ装置を暗黙的に獲得します (または自動的に割り振ります)。ただし、獲得操作を使用して、プログラム用に追加のディスプレイ装置を獲得することもできます。獲得操作は、複数のディスプレイ・ファイル・アプリケーションで使用されます。あるいは、アプリケーションでエラー回復を実行している場合に使用されます。

獲得操作が成功すると、以下のようになります。

- 画面が完全に消去され、カーソルがディスプレイの左上隅に置かれる。
- キーボードはアンロック状態になる。

CRTDSPF、CHGDSPF、または OVRDSPF コマンドの WAITFILE パラメーターで指定する値は、ディスプレイ装置が使用可能になって割り振り可能になるまで獲得操作が待機する時間を決定するために使用されます。ディスプレイ装置が使用可能になる前に待機時間が経過すると、ディスプレイは獲得されません。

ディスプレイ装置は、オンに変更しない限り割り振ることはできません。交換回線ディスプレイ装置は、「オンに変更保留」の状態になっている場合に割り振ることができます。また、*REQUESTER ディスプレイ装置以外のディスプレイ装置は、サインオンされている場合は割り振ることができません。

ディスプレイ・ファイル (複数装置ディスプレイ・ファイルを含む) に対して獲得できる *REQUESTER ディスプレイ装置は 1 つだけです。

獲得操作が失敗すると、ディスプレイ装置に対して行える操作は解放操作だけです。

ディスプレイ・ファイルおよびディスプレイ装置についての情報の入手

開いているディスプレイ・ファイルに対して実行されるオープン操作および入出力操作についての情報と、使用しているディスプレイ装置についての属性情報を取得できます。

オープンおよび入出力操作についての情報の入手

ディスプレイ・ファイルが正常に開くと、システムはフィードバック域でファイルの状況を追跡します。

オープン・フィードバック域には、正常に開かれた後のディスプレイ・ファイルについての以下のような情報が入ります。

- ディスプレイ・ファイルの名前とライブラリー
- ディスプレイ上の行と列の数
- オーバーライド適用後のファイルの名前とライブラリー
- ファイル用に定義されたディスプレイ装置についての情報

オープン・フィードバック域の**装置定義リスト**には、ディスプレイ・ファイルに接続された各装置についての情報が入ります。

入出力フィードバック域には、正常に開かれた後のディスプレイ・ファイルに対して実行される入出力操作についての情報が入ります。I/O フィードバック域には、以下の 2 つのセクションがあります。

- **共通フィードバック域**には、ファイルに対して実行された入出力操作についての情報が入ります。これには、操作数と最後の操作が含まれます。
- **ファイル依存フィードバック域**には、ファイル固有の情報 (メジャー/マイナー戻りコードおよびディスプレイ装置から受け取ったデータ量など) が入ります。

ディスプレイ・ファイルに対して操作を実行すると、フィードバック域が更新されて最新の状況が反映されます。

開いているファイルごとに 1 つのフィードバック域があります。共用ファイルは例外です。共用ファイルは、プログラム-ファイル間のデータ・パスだけでなくフィードバック域も共用します。共用ファイルについて詳しくは、92 ページの『同一ジョブ内でのディスプレイ・ファイルの共用』を参照してください。

フィードバック域は、エラーが発生した場合に情報を提供するために使用することができます。例えば、ディスプレイ・ファイルでエラーが発生すると、プログラムはファイル依存フィードバック域内のメジャー/マイナー戻りコードに基づいて事前定義エラー処理操作を判別できます。メジャー/マイナー戻りコードについての詳細は、『第 8 章 ディスプレイ・ファイルに関するメッセージとエラーの処理』を参照してください。

システム上の一部の高水準言語では、操作の実行対象となるファイルについての状況およびその他の情報にアクセスできます。

フィードバック域で使用可能なすべての情報については、『付録 C. ディスプレイ・ファイルのフィードバック域のレイアウト』を参照してください。

ディスプレイ装置についての属性情報の入手

属性取得操作では、特定のディスプレイ装置についての以下の情報を入手できます。

表 8. 属性取得操作で使用可能な情報

情報	詳細
ディスプレイ装置の特定のモデル	ファイルに対してディスプレイ装置が獲得されている場合にのみ有効
ディスプレイ装置の画面サイズ	ファイルに対してディスプレイ装置が獲得されている場合にのみ有効
装置獲得標識	ファイルに対してディスプレイ装置が現在獲得されているかどうかを示す
装置送信勧誘状況	ディスプレイ装置が送信勧誘されているかどうかを示し、送信勧誘されている場合はディスプレイ装置でデータが使用可能になっているかどうかを示す。ファイルに対してディスプレイ装置が獲得されている場合にのみ有効
*REQUESTER ディスプレイ装置標識	ディスプレイ装置が *REQUESTER ディスプレイ装置かどうかを示す

提供される情報は、ファイルが開かれるときに暗黙的に獲得されるディスプレイ装置のオープンおよび入力フィードバック域でも使用可能であるため、属性取得操作は複数のディスプレイ・ファイル・アプリケーションで最も有用です。属性取得操作の実行方法の詳細については、該当する高水準言語の資料を参照してください。

属性取得操作で使用可能なすべての情報については、『付録 C. ディスプレイ・ファイルのフィードバック域のレイアウト』を参照してください。

データの送信および受信

アプリケーション・プログラムによって行われるディスプレイを対象としたデータの書き込み/読み取りは、レコードで行われます。レコードはフィールド (データの個々の項目) で構成されます。アプリケーション・プログラムが作成されている高水準言語には、ディスプレイに書き込むデータをシステムに提供し、ディスプレイからレコードの形式で読み取られたデータをシステムから受け取る I/O ステートメントがあります。

高水準言語の I/O ステートメントは、レコード様式 (DDS を使用して定義される) も参照します。出力時に、レコード様式は、プログラムによって提供されるデータをディスプレイにどのように表示するか、表示する前にデータをどのように処理するかについて記述します。入力時に、レコード様式は、一部の表示機能を制御し、ディスプレイ上のすべてのデータからプログラム・データを抽出し、そのデータをアプリケーション・プログラムに戻すために使用されます。

DDS キーワードについて

ここでは、ディスプレイを対象とした情報の送信および受信を制御する DDS キーワードを使用します。特定の DDS キーワードについて詳しくは、iSeries Information Center の『DDS』のトピックを参照してください。

ディスプレイ上のアクティブ・レコード様式の判別

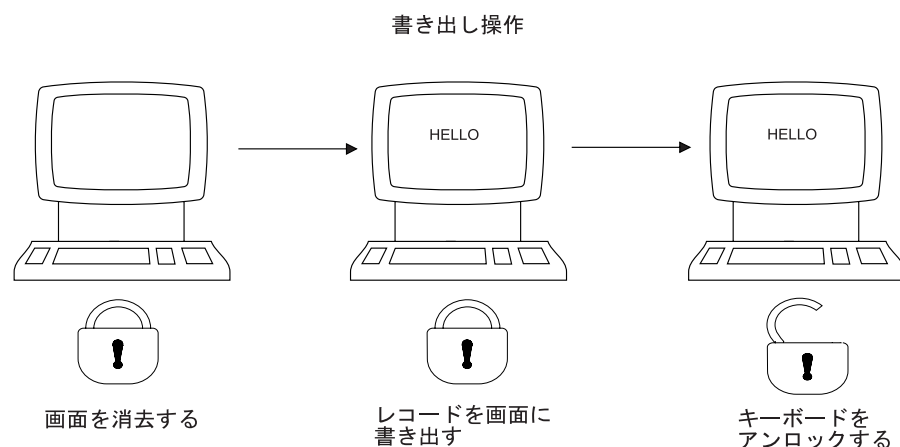
アクティブ・レコード様式テーブル (現在ディスプレイ上にあるすべてのレコード様式のテーブル) がシステムで保守されます。読み取り操作は、アクティブ・レコード様式テーブル内のレコード様式に対してのみ実行可能です。特定の DDS キーワードは、テーブル内のレコードを変更します。

レコード様式が入っている書き込み操作が実行されると、レコード様式がテーブルに追加されます。レコードに対して読み取り操作を正常に実行できなくなると、レコード様式はテーブルから除去されます。

アクティブ・レコード様式テーブルの消去は、ディスプレイが消去されるたびに行われます。

ディスプレイへの出力の書き込み

書き込み操作は、レコードをプログラムからシステムに渡します。ディスプレイ・ファイル内のレコード様式には、システムがレコードを処理するのに必要な情報が入っています。書き込み操作が実行されると、以下のようになります。



RV2W010-1

ディスプレイへのレコードの配置

1 つのレコード様式が画面全体を占有する場合もあれば、複数のレコード様式を表示するために画面を分割することもできます。

複数の行にレコードを表示する場合は、以下の規則が適用されます。

- 行はディスプレイ上の連続した行でなければなりません。例えば、1 つのレコードが 2 行を占有する場合にレコードが行 2 から開始する場合、レコードは行 3 に続きます。
- 行 1、桁 1 を占有できるのは開始属性文字だけです。
- 次の行の桁 1 にあるのがレコードの最後のフィールドの終了属性文字だけの場合、別のレコード様式はその同じ行から開始することができます。
- ディスプレイ上の行を同時に占有できるのは 1 つのレコードだけです。すでにディスプレイ上にあるレコード様式の複数の行にオーバーラップするレコード様式を表示する場合は、行の消去 (CLRL) キーワードに *NO を指定します。CLRL(*NO) は、新しいレコード様式を表示する前に、共通の行を消去します。

以下の 3 つの図は、CLRL キーワードを指定しない場合に画面がどのように分割されるかを示しています。

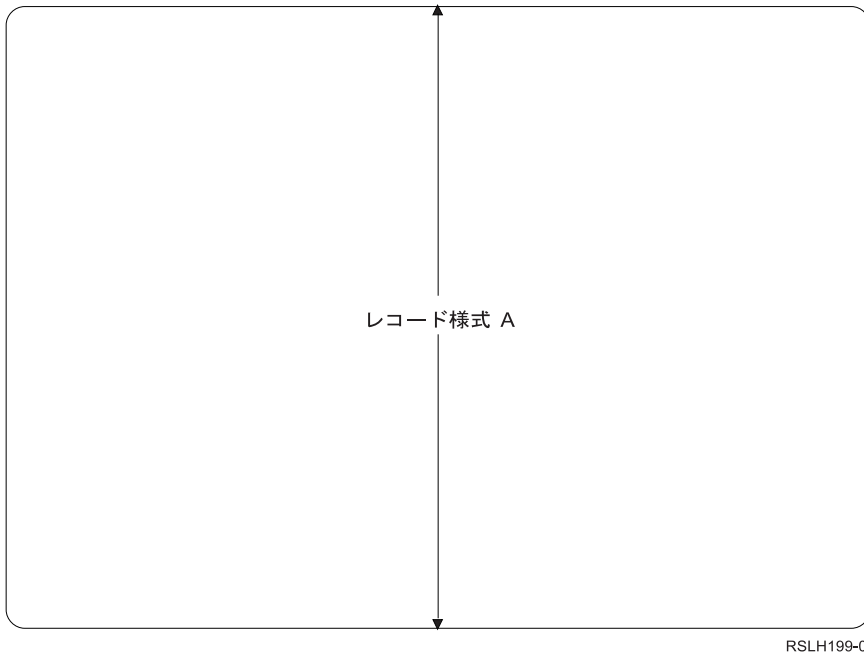


図9. CLRL キーワードを使用しない場合の画面上のレコードの有効な配置 (1/3)

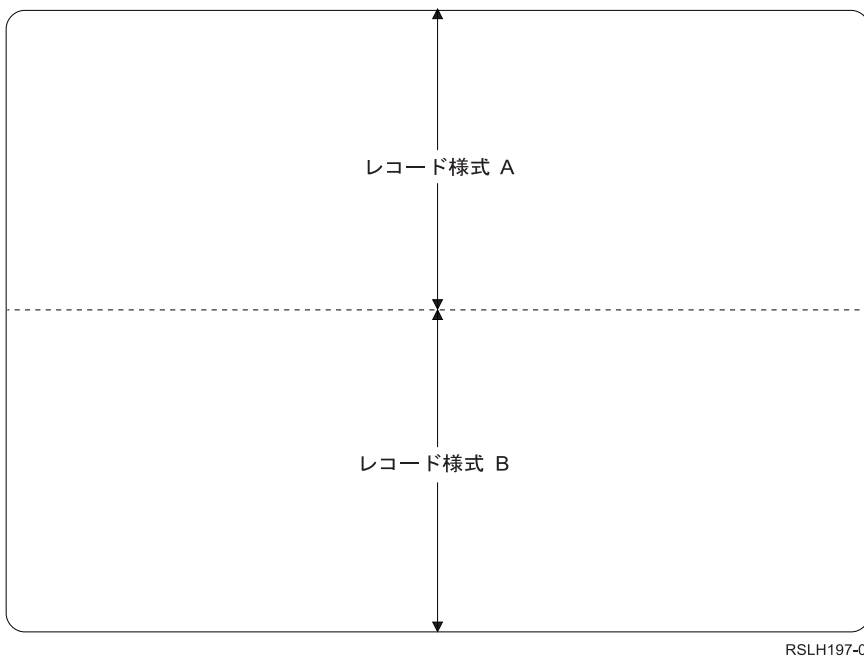


図9. CLRL キーワードを使用しない場合の画面上のレコードの有効な配置 (2/3)

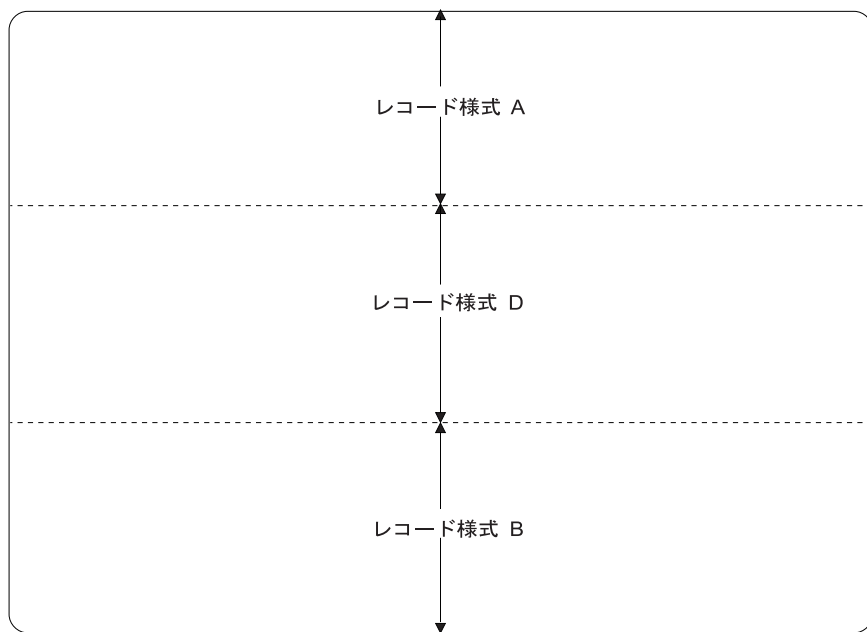
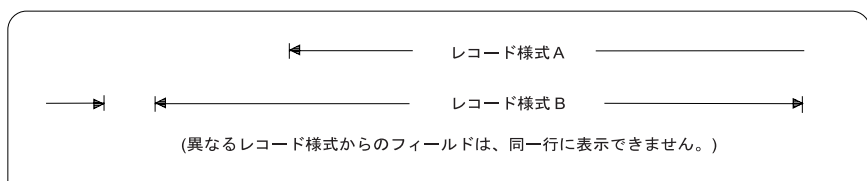
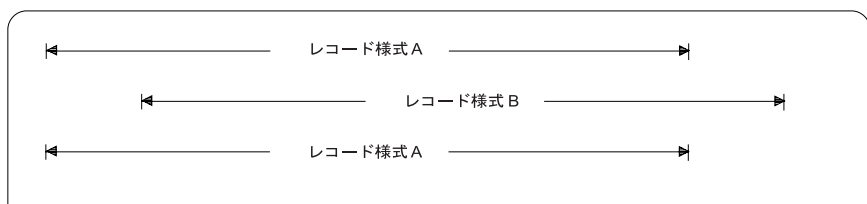
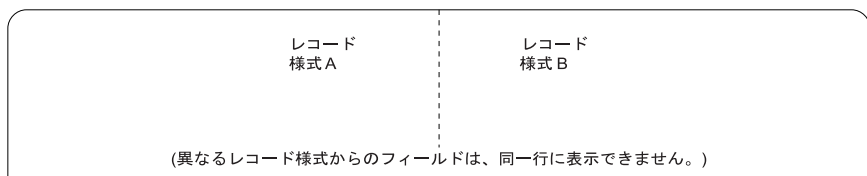


図 9. CLRL キーワードを使用しない場合の画面上のレコードの有効な配置 (3/3)

以下の図は、CLRL キーワードを指定しない場合に画面をどのように分割できないかを示しています。



RV2W048-0

図 10. CLRL キーワードを使用しない場合の画面上のレコードの間違った配置

ディスプレイ上のスペースを占有しないレコードについての理解

以下のタイプのレコードは、ディスプレイ上のスペースを占有しませんが、行 0 に存在すると見なされません。

- フィールドが定義されていないレコード
- 隠しフィールド、プログラム-システム間フィールド、メッセージ・フィールドだけを含むレコード
- CLRL キーワードを指定したレコードで、入力可能なフィールドがないレコード

システムが一度に追跡できるのは、これらのレコードのいずれか 1 つだけです。行 0 に存在すると見なされるレコードの出力操作が、行 0 に存在すると見なされる別のレコードを置き換える場合、置き換えられたレコードに対して入力操作を行うことはできなくなります。

ディスプレイ上のレコード様式の変更

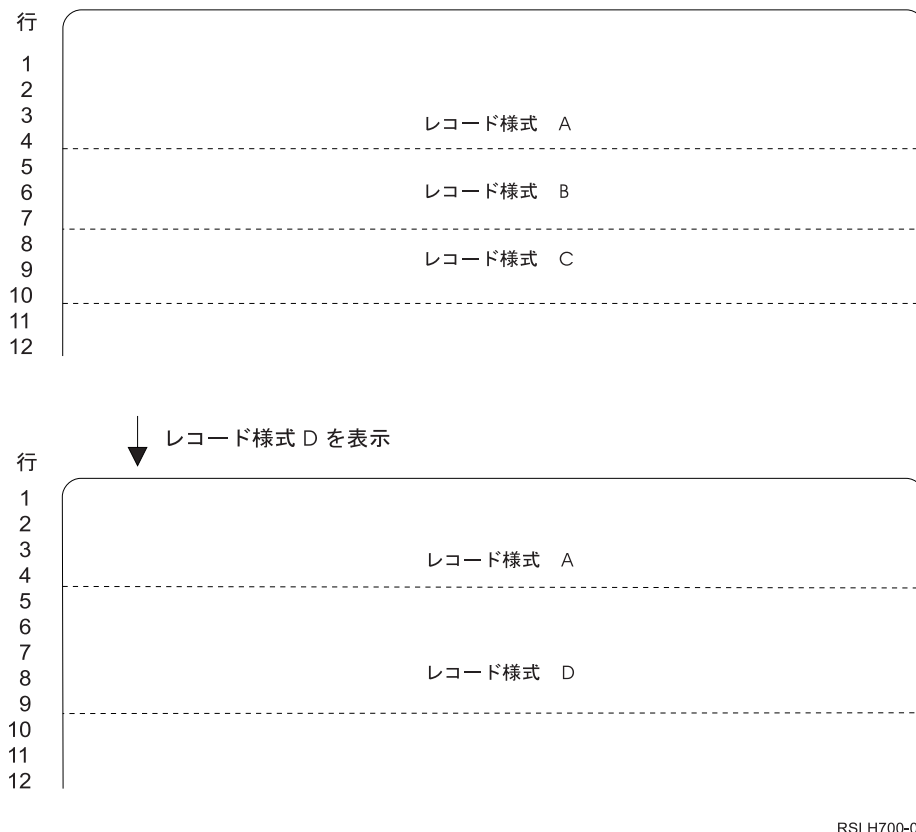
新しい様式が表示される時にディスプレイ上の情報は削除される可能性があるため、表示されている様式が、ファイルの処理中に変更される場合があります。

プログラムが出力用に、または入力を許可するために新しいレコード様式を表示する場合、通常、既存のディスプレイは新しいレコード様式が表示される前に消去されます。例えば、3 つのレコード様式が同時にディスプレイ上に存在する場合に、別のレコードをディスプレイに書き込む場合は、ディスプレイ上の 3 つのレコード様式は消去されます。いくつかの DDS キーワード (OVERLAY キーワードなど) を使用すると、入出力操作時にレコードおよび入力フィールドの表示を制御できます。これらの DDS キーワードの詳細については、53 ページの『ディスプレイ上のレコード様式のオーバーレイおよび消去』および 57 ページの『指定した行数の消去』を参照してください。

以下の例では、レコード様式のフィールドは以下のように定義されています。

- レコード様式 A のフィールドは、行 1 から 4 を占有する。
- レコード様式 B のフィールドは、行 5 から 7 を占有する。
- レコード様式 C のフィールドは、行 8 から 10 を占有する。
- レコード様式 D のフィールド (CLRL キーワードが指定されている) は、行 5 から 9 を占有する。

以下の図では、レコード様式 A、B、および C が最初に表示されています。レコード様式 D が表示されると、レコード様式 B および C が置換されます。



RSLH700-0

図 11. レコード様式の置換

レコード様式 D に OVERLAY キーワードが指定されていなかった場合は、前述の例で以下のことが生じた可能性があります。

- レコード様式 A も削除されていた。
- レコード様式 B の行 5 から 7 がディスプレイに表示されたままになっていた。レコード様式 D によってオーバーレイされたレコード様式 B のフィールド内のデータが変更されていた。(詳細については、57 ページの『指定した行数の消去』を参照。)

ディスプレイに書き込まれるレコード様式の順序の決定

パフォーマンス上の理由により、入力フィールドを含むレコードは、ディスプレイに表示される順序でディスプレイ装置に送信する必要があります。

図 11 で、レコード様式 A と B が両方とも入力フィールドを含んでいて、同じディスプレイに表示される場合、レコード様式 A を先にディスプレイに送信する必要があります。

ディスプレイ上のレコード様式のオーバーレイおよび消去

プログラムが出力用に、または入力を許可するために新しいレコード様式を表示するときに、既存のディスプレイが消去されないよう、OVERLAY キーワードを指定することができます。OVERLAY キーワードを指定すると、完全にまたは部分的に重複するレコードだけが消去され、他のすべてのレコードはディスプレイに表示されたままになります。

注: ファイルが開かれてから最初の書き込み操作で OVERLAY キーワード有効になっていても、DDS キーワード ASSUME がディスプレイ・ファイル内のいずれかのレコード様式に指定されていない限り、画面の消去を回避することはできません。

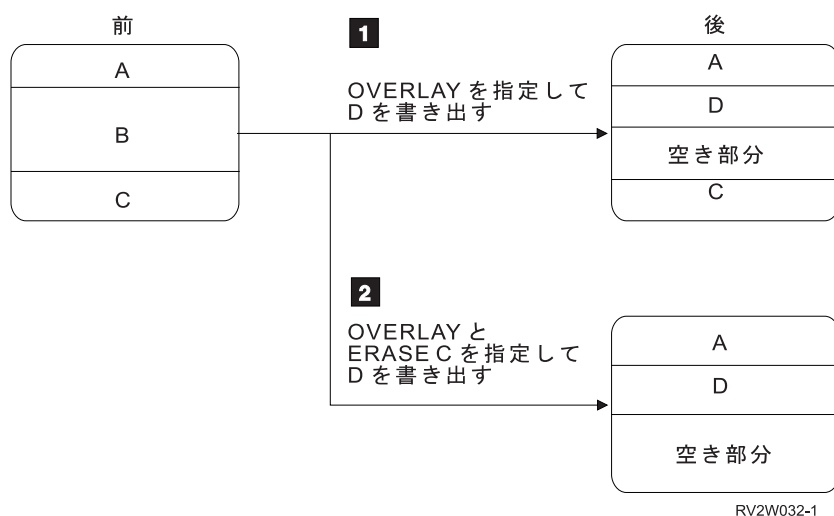
OVERLAY キーワードを使用すると、まとめて提供する必要があるアプリケーションの情報 (本来は 2 つ以上の部分に分かれる) を表示できます。例えば、アプリケーションで、あるレコード様式を使用して状態の情報をディスプレイ上部で提供し、別のレコード様式を使用してその状態内の特定の領域の情報を提供できます。

複数のレコードをディスプレイに配置するには、ディスプレイの書き込み操作と読み取り操作を分離します。次いで、各書き込み操作を実行すると、システムはデータを提供したレコードからデータを取り、レコード様式に指定された情報と結合してディスプレイに配置します。ディスプレイで読み取り操作を実行する準備が整うまでキーボードをロックするには、以下のいずれかを行います。

- すべてのレコード様式に LOCK キーワードを指定する
- ディスプレイ・ファイルの書き出し据え置き (DFRWRT) パラメーターに *YES を指定する

情報のリストをサブファイルでユーザーに対して提供する場合は、複数のレコード様式を使用できます。サブファイルは同じレコード様式を持つレコードのグループで、ディスプレイ装置を対象として 1 つの操作で読み取りおよび書き込みが行われます。サブファイルについては、95 ページの『第 4 章 サブファイルを使用したレコードのグループの表示』を参照してください。

レコードをオーバーレイする場合に特定のレコードをディスプレイから消去するには、ERASE キーワードを OVERLAY キーワードと一緒に使用します。以下の図は、出力操作に対する OVERLAY および ERASE キーワードの影響を示しています。



- 1** レコード様式 B はレコード様式 D がオーバーラップするため消去され、レコード様式 D が表示されます。レコード様式 D は、レコード様式 B が以前に使用していたスペース・レコードのすべてを使用しないため、レコード様式 C をオーバーラップしません。
- 2** レコード様式 B はレコード様式 D がオーバーラップするため消去され、ERASE C が指定されているためレコード様式 C が消去されます。レコード様式 D が表示され、ディスプレイの一部は使用されません。

特定の行からのレコード様式の開始

特定の行からレコード様式を開始するには、開始行番号 (SLNO) キーワードを使用します。SLNO キーワードで、以下のいずれかを指定できます。

- レコード様式の実際の開始行番号 (1 から 27 の値)。実際の行番号を指定すると、システムはレコード内のすべてのフィールドの行番号を指定した値から 1 を減算して調整します。

- 変数開始行番号 (*VAR)。実行時に高水準言語プログラムで開始行番号値を指定できます。プログラムで指定する値によって、以下のようになります。

表 9. SLNO(*VAR) 値の結果

指定する値	結果
0 または値を指定しない	開始行番号 1 が想定されます。
画面上の行数を超える値、または負の値	システムはプログラムにメッセージを送信し、入出力要求は実行されません。
少なくとも 1 つのディスプレイ・サイズのフィールドの開始位置が行 1、列 1	ファイル作成時に警告メッセージ (重大度 10) が発行されます。実行時に、表示中の画面サイズに行 1、列 1 で開始するフィールドが含まれていて、変数開始行番号がプログラムによって 1 に設定されている場合は、エラー・メッセージが発行されます。

開始行番号の設定方法および追加方法は、各プログラミング言語によって異なります。ご使用の言語の該当する資料を参照してください。

システムは、レコード様式内の各フィールドの行番号を指定した値から 1 を減算して調整します。結果行番号が画面サイズを超えている場合、フィールドは表示されません。さらに、フィールドの一部が画面上の最後の行を超える場合、フィールドは表示されません。

SLNO キーワードは、レコード・レベル・キーワード ASSUME、KEEP、USRDFN、SFL、または SFLCTL を含むレコード様式、またはファイル・レベル・キーワード PASSRCD を含むディスプレイ・ファイル内で使用することはできません。

ただし、SLNO キーワードは、以下の場合に他のいくつかの DDS キーワードと一緒に使用することができます。

- SLNO キーワードと一緒に CLRL キーワードを使用する場合で、消去する行数を CLRL キーワードで指定する場合、消去は SLNO キーワードの開始行番号から開始されます。
- CLRL キーワードではなく OVERLAY キーワードと一緒に SLNO(*VAR) キーワードを使用する場合に、毎回異なる開始行番号でレコードを複数回書き込む場合は、新しいレコードが表示される前に直前のレコードが削除されます。
- 有効な PUTOVR、PUTRETAIN、ERRMSG、または ERRMSGID キーワードと一緒に SLNO キーワードを使用する場合、システムは開始行番号を検査して、レコードに対する直前の出力操作の開始行番号が同じだったかどうかを判別します。
 - 開始行番号が同じだった場合、PUTOVR、PUTRETAIN、ERRMSG、または ERRMSGID キーワードで指定するアクションが実行されます。
 - 開始行番号が同じではない場合、PUTOVR、PUTRETAIN、ERRMSG、または ERRMSGID キーワードは無視され、新しい値によって調整された行でレコード様式が表示されます。

以下の DDS は、SLNO(*VAR) キーワードの使用例を示しています。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R ORDENT
A          1 36'ORDER ENTRY'
A          3 2'Enter customer number:'
A          CUST          5  B  +2
A          5 2'Enter order number:'
A          ORDNR          6  B  +2
A          7 2'ITEM NUMBER'
A          7 18'DESCRIPTION'
A          7 44'QUANTITY'
A          R LINITM
A          OVERLAY
A          SLNO(*VAR)
A          CLRL(*NO)
A          ITEM          6 00 9 2
A          DESCRP        20 0 9 18
A          QTYORD         3 00 9 44
A          R INPFMT
A          OVERLAY
A          23 2'Enter item number:'
A          ITMNR          6 0I  +2
A          +5'Enter Qty:'
A          QTY           3 0I  +2

```

図 12. SLNO(*VAR) キーワードの使用法を示すサンプル DDS ソース

この例では、レコード様式 ORDENT に「Order Entry」画面のプロンプトが含まれています。ユーザーがカスタマー番号とオーダー番号を入力すると、以下のようになります。

1. プログラムがレコード様式 INPFMT を画面に書き込み、ユーザーが項目番号と注文数を入力できるようになります。
2. ユーザーが項目番号と数量を入力すると、プログラムは項目の説明をファイルから取得し、レコード様式 LINITM を画面に書き込みます。
3. ユーザーが別の項目番号を入力できるよう、プログラムは INPFMT レコード様式を画面に書き込みます。

画面がこのように設計されているため、ユーザーは項目番号と数量を同じ行に入力できます。行項目を入力すると、プログラムは LINITM レコード様式を使用して画面上でオーダーを作成します。LINITM レコード様式が書き込まれるたびにプログラムが行を画面に追加できるよう、SLNO(*VAR) キーワードを使用します。新しいレコードが書き込まれるときに直前のレコードが削除されないよう、LINITM レコード様式に CLRL(*NO) キーワードを指定する必要があります。

LINITM レコード様式が最初に画面に書き込まれるときは、フィールドが行 9 に表示されるよう *VAR の値は 1 になります。このレコード様式に対する出力操作が成功するたびに、新しい行項目が画面に追加されるようプログラムは開始行番号に 1 を加算します。

ユーザーが 2 つの項目番号と数量を入力すると、画面は以下のようになります。


```

                                ORDER ENTRY

Enter customer number: 34785

Enter order number: 1J2340

ITEM NUMBER   DESCRIPTION   QUANTITY
96321         Pliers           115
86768         Saws              125

Enter item number:  ___ Enter Qty:  ___

```

ユーザーがデータを常に同じ行に入力するようにし、前の例に示されているとおり、以前に入力されたレコードの画面を作成する場合は、SLNO キーワードが最も効果的です。ただし、同時に複数のレコードを表示する標準の照会機能の場合は、サブファイルを使用する方が効果的です。

指定した行数の消去

レコード様式を画面に書き込む前に画面上の特定の行数を消去するには、行の消去 (CLRL) キーワードを使用します。表示されるフィールドがレコードに含まれていない場合でも CLRL キーワードを指定できます。消去は開始行番号から開始し、消去される行数は CLRL キーワードに指定する値 (1 から 27 の任意の値) で決定されます。開始行番号は、以下のように決定されます。

- SLNO キーワードを指定しない場合、開始行番号はフィールド位置によって決定されます。
- SLNO(nn) キーワードを指定する場合、nn が開始行番号になります。
- SLNO(*VAR) キーワードを指定する場合、開始行番号はディスプレイ・ファイルの作成時にデフォルトで 1 になりますが、実行時にアプリケーション・プログラムによって変更可能です。

以下の値を指定して、特定の行を消去することもできます。

値 消去される行

***END** 開始行から画面の最後まですべての行

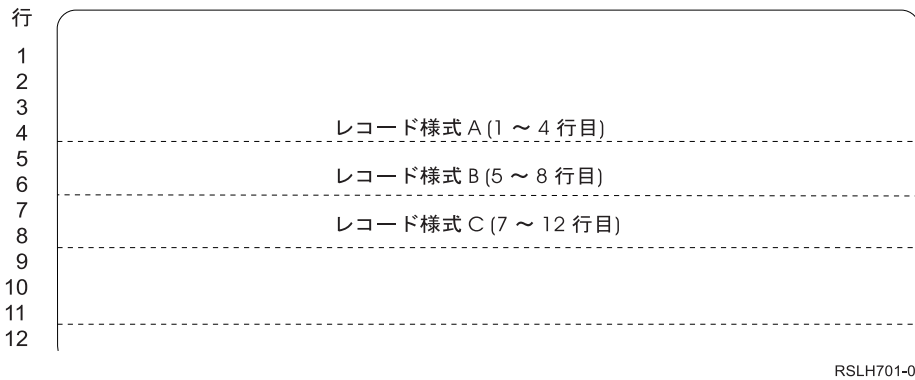
***NO** オーバーラップするレコード様式で使用されている画面の行のみ

***ALL** 画面のすべての行。デフォルトのアクションでは画面のすべての行が消去されますが、このデフォルトを変更する DDS キーワード (USRDSPMGT など) も指定しない限り、通常は CLRL(*ALL) を指定する必要はありません。

注: CLRL キーワードを使用する場合、CRTDSPF または CHGDSPF コマンドの RSTDSP パラメーターに *YES を指定する必要があります。そうしないと、ファイルが中断状態になる場合にディスプレイ上のデータが失われる可能性があります。

CLRL(*NO) キーワードを使用すると、オーバーラップするレコードがディスプレイに書き込まれるときに、オーバーラップされるレコードが削除されないようにすることができます。このキーワードを使用する場合は、オーバーラップされる表示中のレコードは画面から削除されず、新しいレコードが表示中のレコードを完全にまたは部分的にオーバーレイします。画面に繰り返し送信される定数およびデータがディスプレ

イに含まれている場合、CLRL(*NO) を使用すると、パフォーマンス上の利点が得られます。定数を別の様式で送信し、データのみを含む様式に CLRL(*NO) を指定することにより、レコード様式をディスプレイに送信するのに必要な時間を節約できます。以下に例を示します。



レコード様式 C に CLRL(*NO) を指定する場合、レコード様式 C が画面に書き込まれるときに、レコード様式 C によってオーバーラップされないレコード様式 B のすべてのフィールドは画面上に残ります。この同じ状態で OVERLAY または PUTOVR キーワードを使用する場合、レコード様式 C がレコード様式 B をオーバーラップするため、レコード様式 C が画面に書き込まれるときにレコード様式 B は削除されます。

以下の考慮事項は、CLRL キーワードを他の DDS キーワードと一緒に使用する場合に適用されます。

- 入力可能フィールドが含まれているレコード様式に CLRL キーワードを指定する場合、オーバーラップされるレコード内の入力可能フィールドは入力不能になります。オーバーラップされない他のすべてのレコード様式内のフィールドには入力可能です。これらのフィールドを入力可能のままにしない場合は、PROTECT キーワードを CLRL(nn) キーワードと一緒にレコード様式で使用する必要があります。
- CLRL キーワードを指定したレコードで、入力可能フィールドを持たないレコードは、行 0 に存在すると見なされます。したがって、入力可能フィールドを持たないレコード様式に CLRL(nn) キーワードを指定すると、既にディスプレイ上にあるすべてのレコードはディスプレイに残り、入力可能フィールドにも入力可能です。行 0 から開始するレコードはシステムに認識されないため、ROLLUP および ROLLDOWN キーワードはこれらのレコードに対して機能しません。また、これらのレコードは、OVERLAY キーワードが指定された他のレコードによってオーバーラップされるときに完全に消去されない可能性があります。オーバーラップするレコードに必要な行は消去されますが、オーバーラップするレコードで不要な行は画面に残ります。
- CLRL(nn) キーワードは、レコード・レベル・キーワード ASSUME、KEEP、USRDFN、SFL、または SFLCTL を含むレコード様式、またはファイル・レベル・キーワード PASSRCD を含むディスプレイ・ファイル内で使用することはできません。
- ERRMSG または ERRMSGID キーワードが有効な場合、CLRL(nn) キーワードは無視されます。
- CLRL(nn) キーワードを使用する場合で、PUTOVR または PUTRETAIN キーワードが有効な場合、行の消去は PUTOVR または PUTRETAIN 機能と競合します。PUTOVR または PUTRETAIN キーワードを使用する場合、オーバーライドされるフィールドがディスプレイ上に存在することが必要ですが、CLRL(nn) キーワードはこれらのフィールドを最初に消去する場合があります。CLRL(nn) キーワードを使用するためにレコードが入力不能になる場合、PUTOVR キーワードが有効であれば入力可能フィールドは入力可能のままです。ただし、プログラムがこのようなレコードを読み取ろうとする場合、システムはメッセージを発行します。CLRL(nn)、CLRL(*NO)、および CLRL(*END) キーワードは OVERLAY キーワードを暗黙指定しますが、CLRL キーワードと OVERLAY キーワードには、以下の例に示すような相違があります。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R RECORD1
A          FLD1          10  I  4  5
A          FLD2          10  I  5  5
A                                     5 21'Enter employee number'
A          R RECORD2A   OVERLAY
A          FLD3          1  B  5  2
A                                     6 2'Required field'
A                                     7 2'Enter 1, 2, or N'
A          FLD4          19  0  8  2
A          R RECORD2B   CLRL(4)
A          FLD5          1  B  5  2
A                                     6 2'Required field'
A                                     7 2'Enter 1, 2, or N'
A          FLD6          19  0  8  2
A          R RECORD3   OVERLAY
A          FLD7          10  0  8 15
A          .
A          .
A          .
A          FLD8          10  B 12  4
A          R RECORD4   CLRL(*NO)
A          FLD9          42  I 11  2

```

図 13. CLRL と OVERLAY の相違を示すサンプル DDS ソース

プログラムが以下の順序でレコード様式に対して出力操作を実行する場合は、以下ようになります。

表 10. CLRL 例の結果

レコード様式の順序	結果
RECORD1 RECORD3 RECORD2A	RECORD2A が RECORD1 および RECORD3 をオーバーラップし、OVERLAY キーワードだけが RECORD2A に指定されているため、RECORD2A がディスプレイに書き込まれるときに行 4 から 12 が削除されます。
RECORD1 RECORD3 RECORD2B	CLRL(4) キーワードが指定されているため、RECORD2B がディスプレイに書き込まれる前に行 5 から 8 が消去されます。RECORD1 内の FLD1 および RECORD3 内の入力可能フィールド (行 9 から 12) は画面に残りますが、RECORD1 および RECORD3 の一部が RECORD2B によってオーバーラップされるため、入力不能になります。
RECORD1 RECORD3 RECORD4 RECORD2A	RECORD3 に OVERLAY キーワードが指定されているため、RECORD3 が画面に書き込まれるときに RECORD1 が画面に残ります。RECORD4 は、画面に書き込まれるときに行 11 (RECORD3 でも使用される) の一部を使用し、RECORD4 に CLRL(*NO) が指定されているため、RECORD3 が画面に残ります。ただし、システムは RECORD3 が画面上にあることを認識しないため、RECORD2A が書き込まれるときに行 4 から 8 だけが消去され、RECORD3 の行 8 より下の部分は画面に残ります。

ディスプレイ上の 2 行間のデータのロール

高水準言語を使用している場合は、ロール可能 (ALWROL) キーワードを指定することによって、ディスプレイ上の 2 行間のデータをロールアップまたはロールダウンできます。データがロールされたために空になった行はヌルに設定され、それらの行に別のレコード様式を書き込むことができます。

プログラムで、以下を指定する必要があります。

- ロールする行の開始行番号と終了行番号。開始行番号と終了行番号によって、画面上のウィンドウが定義されます。

- **ロールする行数。**ロールする行数を正数で指定すると、データはロールアップされます。ロールする行数を負数で指定すると、データはロールダウンされます。
- **ロールアップまたはロールダウン。**

ウィンドウでは、プログラムに指定した行数のデータ行がロールアップ (またはロールダウン) します。ウィンドウの外へロールしたデータは失われます。レコード様式の一部または全部がウィンドウ内にある場合、そのレコード様式の入力可能フィールドは入力不能になります。ロール後、プログラムはウィンドウ内のレコード様式に対して入力操作を実行できなくなります。

以下の例は、プログラム制御ロールが行われる前のディスプレイと、プログラム制御ロールが行われた後のディスプレイを示しています。プログラムでの指定は、以下のとおりです。

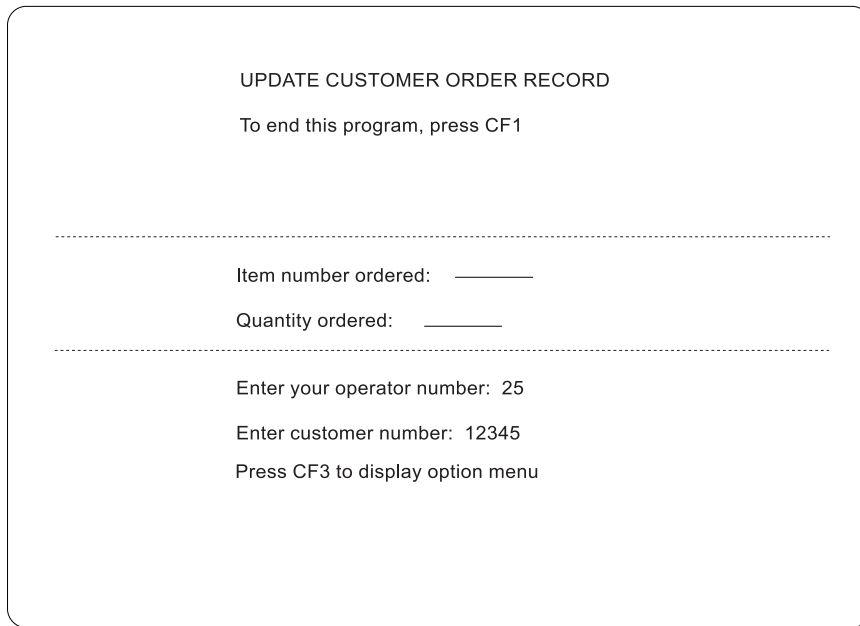
- 開始行番号、8
- 終了行番号、18
- ロールダウンする行数、6

ロール操作前のディスプレイ

<pre> UPDATE CUSTOMER ORDER RECORD To end this program, press CF1 Enter your operator number: _____ Enter customer number: _____ Press CF3 to display option menu </pre>	<div style="display: flex; align-items: center;"> <div style="margin-right: 5px;">1 行目</div> <div style="font-size: 2em;">}</div> </div> <div style="margin-right: 5px;">3 行目</div> <div style="margin-right: 5px;">8 行目</div> <div style="margin-right: 5px;">10 行目</div> <div style="margin-right: 5px;">12 行目</div>	<div style="display: flex; align-items: center;"> <div style="margin-right: 5px;">レコード 様式 1</div> <div style="font-size: 2em;">}</div> </div>
---	--	---

RSLH165-0

ロールダウン操作後のディスプレイ



1 行目 }
 3 行目 } 変化なし

9 行目 } レコード
 11 行目 } 様式 2

14 行目 }
 16 行目 }
 18 行目 }
 以前の 8 ~ 12 行目
 がロールダウン
 したところ

RSLH171-0

ALWROL キーワードは、ファイル・レベル・キーワード PASSRCD、またはレコード・レベル・キーワード KEEP、ASSUME、USRDFN、SFL、または SFLCTL と一緒に使用することはできません。

ALWROL キーワードが有効になっているのと同じ出力操作で ERRMSG、ERRMSGID、PUTOVR、または PUTRETAIN キーワードが有効になっている場合、システムはメッセージ CPF5014 を発行します。ERRMSG、ERRMSGID、PUTOVR、またはレコード・レベル PUTRETAIN キーワードが有効になっていない場合、メッセージは発行されません。ただし、PUTRETAIN キーワードをフィールド・レベルでオプション標識と一緒に指定する場合、PUTRETAIN キーワードのオプション標識をオンまたはオフにするとメッセージ (CPF5014) が発行されます。

属性またはフィールドの内容のオーバーライド

レコードの一部のデータおよび属性だけをディスプレイに送信するには、以下のキーワードを使用します。

- 明示的な一時変更を伴う PUT (PUTOVR)
- データ一時変更 (OVRDTA)
- 属性一時変更 (OVRATR)

送信するデータまたは属性を少なくすると、ディスプレイ (特にリモート接続されたディスプレイ) の応答時間を短縮できます。

PUTOVR キーワードを指定すると、以下のようになります。

- OVRATR キーワードが有効になっているフィールドの表示属性がオーバーライドされます。
- OVRDTA キーワードが有効になっているフィールドのデータ内容がオーバーライドされます。
- OVERLAY キーワードを指定していない場合でも、出力操作は OVERLAY キーワードが有効になっているかのように機能します。

PUTOVR キーワードは、PUTRETAIN キーワードを含むレコード様式に指定したり、サブファイル・レコードに使用することはできません。

OVRATR キーワードによってオーバーライド可能な表示属性は以下のとおりです。

CHECK(ER) レコードの終わり
 CHECK(ME) 入力必須
 DSPATR(MDT) 変更データ・タグの設定
 DSPATR(PR) 保護
 DSPATR(BL) 明滅
 DSPATR(CS) 桁の区切り記号
 DSPATR(HI) 高輝度
 DSPATR(ND) 非表示
 DSPATR(PC) カーソルの配置
 DSPATR(RI) 反転表示
 DSPATR(UL) 下線
 DUP 複写キー対応

PUTOVR キーワードの例を以下に示します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R ITMRVW
A
A          PUTOVR
A          1 35'ITEM REVIEW'
A          3 2'Item number:'
A          ITMNBR      5  B  +2
A          5 2'Item description:'
A          ITMDSC      20  +2
A 10          OVRDTA
A          7 2'Item price:'
A          ITMPRC      8  2  +2
A 15          OVRDTA
A          9 2'Warehouse location:'
A          WHSLOC      3    +2
A 20          OVRDTA
A          11 2'Quantity on hand:'
A          QTYOH       5  0  +2
A 25          OVRDTA
A          OVRATR
A N25 30      DSPATR(HI)
```

図 14. PUTOVR キーワードの使用法を示すサンプル DDS ソース

DDS は、ユーザーが項目番号を入力したり、項目の説明、項目の価格、ウェアハウスの場所、および在庫数量を確認できるディスプレイを記述します。

1. 最初の出力操作時に、すべてのフィールドがディスプレイに送信され、すべてのオプション標識がオフになります。レコードがまだ画面上にないため、PUTOVR キーワードは無視されます。最初の出力操作時に、プログラム内の現行フィールド値が出力フィールドに表示されます。プログラムがこれらのフィールドを設定していない場合、値は高水準言語が出力バッファを初期化するために使用した値になります。

最初の出力操作時に常に特定の値が出力可能フィールドに入っている必要がある場合、DFT または DFTVAL キーワードを使用して、フィールドをその値に初期化できます。PUTOVR および OVRDTA

キーワードが指定された出力可能フィールドで DFT キーワードを使用すると、レコードが最初にディスプレイに配置されるときに、システムはプログラム値ではなくデフォルト値をディスプレイに配置します。

2. ユーザーが項目番号を入力します。プログラムは標識 10、15、20、および 25 をオンに設定し、書き込み・読み取り操作を実行して出力フィールドを表示します。書き込み操作時に、レコードがすでに画面上にあるため、PUTOVR キーワードが有効になります。ITMDSC、ITMPRC、WHSLOC、および QTYOH フィールドに OVRDTA キーワードが指定されていて、これらのフィールドのオプション標識がオンになっているため、これらのフィールドはディスプレイに送信される唯一のデータです。

ユーザーが別の項目番号を入力しても、すでに表示されているフィールドのデータが変更されない場合、プログラムはオプション標識をオフに設定し、そのフィールドを再表示することはありません。例えば、2 番目の項目番号について、WHSLOC が最初の項目番号と同じであるとします。2 番目の項目番号の情報を表示する出力操作時に、プログラムは標識 20 をオフに設定します。したがって、標識 10、15、および 25 がオンになっているため、ディスプレイに送信されるフィールドは ITMDSC、ITMPRC、および QTYOH だけです。

QTYOH フィールドの場合、プログラムは、出力操作の前に標識 25 をオフに設定し、標識 30 をオンに設定することにより、データを変更することなくフィールドの属性を変更できます。

OVRDTA キーワードでオプション標識を使用して、ディスプレイに送信するフィールドを制御できます。オプション標識を使用しない場合は、PUTOVR キーワードが有効なときに OVRDTA キーワードが有効なため、OVRDTA キーワードが指定されたすべてのフィールドが各出力操作時にディスプレイに送信されます。前の例では、オプション標識が使用されていない場合、各出力操作時に 4 つのすべてのフィールドがディスプレイに送信されます。同じ標識を使用して、複数のフィールドを制御することもできます。

この同じアプリケーションの代わりに設計として、2 つのレコード様式を使用して、一方のレコード様式で定数をディスプレイに送信し、もう一方のレコード様式で変数を送信する方法があります。定数が入っているレコード様式が消去されないよう、CLRL(*NO) キーワードを使用する必要があります。ただし、レコード様式がすでにディスプレイ上にある場合、PUTOVR キーワードを使用するのが最も効果的な方法です。

以下の例は、PUTOVR キーワードを使用して効果的なコーディングを行う方法を示しています。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R PROMPT
A          CF03(91 'Return')
A          PUTOVR
A          ERASEINP
A          OVERLAY
A          1 28'Efficient Coding Example'
A          3 2'FLD1'
A          FLD1      5  I 3 7
A          5 2'FLD2'
A          FLD2      5  5 7
A          OVRDTA
A          7 2'FLD3'
A          FLD3      5  7 7
A 15          OVRDTA
A          9 2'FLD4'
A          FLD4      5  9 7
A          OVRDTA
A 16          DSPATR(HI)
A          11 2'FLD5'
A          FLD5      5  11 7DFT('ABCDE')
A          OVRDTA
A 17          DSPATR(HI)
A          13 2'Constant 1'
A          OVRATR
A 18          DSPATR(BL)
A          15 2'Constant 2'
A          OVRATR
A N19        DSPATR(ND)
A          17 2'Constant 3'
A 20          OVRATR DSPATR(RI)

```

図 15. PUTOVR キーワードの効果的な使用法を示すサンプル DDS ソース

前の例では、以下のようになります。

- レコード様式が現在ディスプレイ上にない場合、レコード様式が表示されるときに PUTOVR、OVRATR、および OVRDTA キーワードは無視されます。レコード様式がすでにディスプレイ上にあり、PUTOVR キーワードが有効な場合、後続の出力操作時に、OVRATR または OVRDTA キーワードで定義されるフィールドまたは定数だけがディスプレイに送信されます。すべての入力フィールドを消去するのに最も効果的な ERASEINP キーワードが使用され、ERASEINP キーワードで必須の OVERLAY キーワードが使用されます。
- FLD1 は、レコード様式が表示されるたびに消去される入力フィールドです。
- FLD2 は、関連する OVRDTA キーワードが無条件に指定されるため、レコード様式が表示されるたびにディスプレイに送信されます。
- FLD3 は、最初の出力操作時にディスプレイに送信されます。後続の出力操作時に、FLD3 は、標識 15 (OVRDTA キーワードの条件設定で使用される) がオンになっていない限りディスプレイに送信されません。
- FLD4 は、関連する OVRDTA キーワードが無条件に指定されるため、各出力操作時にディスプレイに送信されます。OVRDTA キーワードが有効な場合、フィールドの属性が常にディスプレイに送信されます。標識 16 は、FLD4 の DSPATR(HI) キーワードを制御するために使用されます。
- 最初の出力操作時に、デフォルト値の ABCDE が FLD5 に表示されます。後続の出力操作時に、関連する OVRDTA キーワードが無条件に指定されるため、プログラムの値が FLD5 に表示されます。標識 17 は、FLD5 の DSPATR(HI) キーワードを制御するために使用されます。

7. 定数 1 は常に表示されますが、最初の出力操作時のみディスプレイに送信されます。ただし、フィールドの属性は、レコード様式が書き込まれるたびにディスプレイに送信され、オプション標識 18 は、フィールドが明滅するかどうかを制御するために使用されます。
8. 定数 2 は、最初の出力操作時のみディスプレイに送信されます。ただし、フィールドの属性は、レコード様式が書き込まれるたびにディスプレイに送信され、オプション標識 19 がオフの場合、定数 2 は表示されません。
9. 定数 3 は、最初の出力操作時のみディスプレイに送信されます。ただし、このフィールドの属性は、標識 20 がオンになっていない限り、後続の出力操作時にディスプレイに送信されません。出力操作の実行時に標識 20 がオンになっている場合、定数 3 は反転表示され、後続の出力操作時に標識 20 の状況に関係なく反転表示されたままです。

以下の例は、ユーザーがレコード・グループに共通の情報を入力し、その後グループ内の特定のレコードに関連する詳細情報を繰り返し入力するアプリケーションに、PUTOVR キーワードをどのように使用できるかを示しています。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R HEADING          TEXT('Header Display')
A                               SETOF(88 'ERASEINP CTL')
A                               CF03(91 'Return')
A                               1 2'HEADING INFORMATION'
A                               DSPATR(HI)
A                               1 60'CF3-End of Program'
A                               2 60'CF2-New heading'
A          HDING              5  I  +2
A          R DETAIL          TEXT('Detail display')
A                               OVERLAY
A                               PUTOVR
A                               PROTECT
A          88                ERASEINP
A                               CF02(92 'New header')
A                               8 2'DETAIL DISPLAY'
A                               DSPATR(HI)
A                               10 2'Input'
A          FLDA              5  I  +2
A                               12 2'Output'
A          FLDB              5    +2DFT(' ')
A                               OVRDTA

```

図 16. PUTOVR キーワードの別の使用法を示すサンプル DDS ソース

前の例では、以下のようになります。

1. プログラムは HEADING レコード様式を表示してから、HDING フィールドを入力として受け取るためにレコード様式に対して入力操作を実行します。HEADING レコード様式内の SETOF キーワードは標識 88 をオフに設定します。この標識は、DETAIL レコード様式内の ERASEINP キーワードの条件を設定するために使用されます。
2. 次に、プログラムは DETAIL レコード様式を表示します。OVERLAY キーワードが有効であるため、HEADING レコード様式はディスプレイ上に残ります。PROTECT キーワードも有効であるため、HEADING レコード様式内の入力フィールド (HDING) が保護されます。したがって、ユーザーは、DETAIL レコード様式が表示されるときにこのフィールドを変更できません。
3. ERASEINP キーワードの条件は、オプション標識 88 によって設定されます。DETAIL レコード様式が最初に表示されるときに標識 88 はオフであるため、ERASEINP キーワードは有効ではありません。後続の出力操作時に、標識 88 はオンに設定され、ERASEINP キーワードが有効になります。したがって、後続の出力操作時に FLDA が消去されます。DETAIL レコード様式が最初に表示されるときに有効にならないよう、ERASEINP キーワードでオプション標識が使用されています。ERASEINP キーワ

ードは PROTECT キーワードの前に処理されるため、DETAIL レコード様式が最初に書き込まれるときに有効であった場合は、HEADING レコード様式内の HDING フィールドが消去されます。

4. FLDB は、OVRDTA キーワードが無条件に指定されるため、各出力操作時にディスプレイに送信されます。DETAIL レコードがグループ用に最初に表示されるときに、フィールドにデータが入らないよう、ブランクの値を指定した DFT キーワードが使用されます。

ディスプレイ上のすべての無保護の入力および出力/入力フィールドの消去

無保護のすべての入力可能フィールドを消去するには、入力消去 (ERASEINP) キーワードを使用します。ERASEINP キーワードは、常に OVERLAY キーワードと一緒に使用する必要があります。

変更データ・タグがオンになっている無保護のすべての入力可能フィールドを消去するには、ERASEINP キーワードに *MDTON を指定します。変更データ・タグがオンになっているかどうかにかかわらず、無保護のすべての入力可能フィールドを消去するには、ERASEINP キーワードに *ALL を指定します。

ERASEINP キーワードは、ディスプレイにブランクを送信するのではなく、フィールドを消去するため、応答時間が短縮されます。画面の消去されるフィールドの、次の読み取り操作の変更データ・タグがオンに設定されていない場合には、それらのフィールドに対して入力保管域からデータが戻されます。これは、ディスプレイ装置から前回フィールドが戻されたときにシステムによって保管されたデータです。

INZINP キーワードをレコード・レベルで ERASEINP(*ALL) および PUTOVR と一緒に使用すると、消去されたフィールドのデータをディスプレイに送信することなく、入力保管域を初期化できます。

ディスプレイ上のレコードに関連付けられた変更データ・タグのリセット

変更データ・タグをリセットするには、変更データ・タグ・オフ (MDTOFF) キーワードを使用します。MDTOFF キーワード (常に OVERLAY キーワードと一緒に使用する必要がある) は、次のレコードが表示される前に処理されます。

無保護フィールドの変更データ・タグだけをリセットするには、MDTOFF キーワードに *UNPR を指定します。すべての入力可能フィールドの変更データ・タグをリセットするには、MDTOFF キーワードに *ALL を指定します。

レコードまたはフィールドのディスプレイ上での保持

PUTRETAIN キーワードを使用すると、ディスプレイに送信される文字数を減らすことができます。このキーワードは、常に OVERLAY キーワードと一緒に使用する必要があります。フィールドの表示属性だけを変更できます。PUTRETAIN キーワードを指定と、データの送信以外の他のすべての機能がサポートされます。

PUTRETAIN キーワードをレコード様式レベルまたはフィールド・レベルのいずれで使用しても、以前にディスプレイに書き込まれたこのレコードのフィールドは、この書き込み操作に選択されていない場合でも、ディスプレイ上に残ります。これを回避するには、PUTRETAIN キーワードをフィールド・レベルで使用し、フィールドを 2 回定義します。1 回目はフィールドをディスプレイに表示するときにオプション標識を指定して定義し、2 回目はオプション標識を指定しないでブランクの値を持つ定数として定義します。最初のフィールドが選択されない場合は、2 番目のフィールドが選択されます。2 番目のフィールドが表示され、選択されていないフィールドの内容がブランクによって消去されます。

注: ERRMSG および ERRMSGID キーワードは、レコード様式レベルで PUTRETAIN キーワードを指定したかのように機能します。つまり、ERRMSG および ERRMSGID キーワードが有効な場合、フィールドはディスプレイに送信されず、他のフィールドのフィールド属性は変更されず、コマンド・キーは変更されません。

以下は、PUTRETAIN キーワードをレコード様式レベルで使用する場合の例です。以下の DDS は、3 つのオプションがある生徒検索メニューを記述しています。選択されたオプションは強調表示されます。例えば、オプション 1 が選択される場合、文字ストリング「1. By number」が強調表示されます。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R SELECT          OVERLAY
A          PUTRETAIN ERASEINP
A N44      1 2'STUDENT SEARCH MENU'
A N44      3 10'1. By number'
A 10      DSPATR(HI)
A N44      4 10'2. By name'
A 11      DSPATR(HI)
A N44      5 10'3. By address'
A 12      DSPATR(HI)
A N44      10 2'Select the number of the item to +
A          search by:'
A          INPUT          1  I 10 47
A 44      DSPATR(RI)
```

図 17. PUTRETAIN キーワードの使用法を示すサンプル DDS ソース

以下のようになります。

1. 最初の出力操作時に、すべてのフィールドがディスプレイに送信され、すべてのオプション標識がオフになります。レコードがまだディスプレイ上にないため、PUTRETAIN キーワードは無視されます。
2. ユーザーが項目 1、2、または 3 を選択します。プログラムが入力を受け取ると、選択された項目に基づいて標識 10、11、または 12 がオンに設定されます。1、2、または 3 以外の項目が選択されると、プログラムは標識 44 をオンに設定します。

次の出力操作時に、どちらの標識がオンになっているかに基づいて、フィールド 1、2、または 3 が強調表示されるか、入力フィールドが反転表示されます。

すべてのフィールドのデータはディスプレイに再送されませんが、フィールド属性は再送されます。定数のデータは送信されません。各出力フィールドまたは定数の属性を再送するには、4 バイト必要です。各入力可能フィールドの属性を再送するには、9 バイト必要です。PUTRETAIN キーワードを使用することにより、ディスプレイに送信される文字数を 138 文字から 42 文字に 96 文字減らすことができます。(この数には、データを構成するのに必要なプロトコル制御文字は含まれていません。)

ERASEINP キーワードを指定すると、ユーザーによる選択が消去されます。

以下は、PUTRETAIN キーワードをフィールド・レベルで使用する場合の例です。ここでは、PUTRETAIN キーワードを使用して、有効ではない入力を保持し、ディスプレイに送信される文字数を減らしています。以下の DDS は、項目の名前、色、形状、サイズ、および数量の入力を促す表示が含まれているディスプレイを記述しています。ユーザーは、色、形状、およびサイズの値を変更できます。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R CHANGE          OVERLAY
A          1 2'CHANGE MENU'
A N43      3 2'Change the underlined fields to  +
A          change the description.'
A          4 2'Item:'
A          ITEM          20 0 4 12
A 44      PUTRETAIN
A          5 2'Quantity:'
A          QTY          4Y 0I 5 12
A 44      PUTRETAIN
A 09      DSPATR(BL PC)
A          6 2'Color:'
A          COLOR      10 B 6 12
A 44      PUTRETAIN
A 10      DSPATR(BL PC)
A          7 2'Shape:'
A          SHAPE      10 B 7 12
A 44      PUTRETAIN
A 11      DSPATR(BL PC)
A          8 2'Size:'
A          SIZE      10 B 8 12
A 44      PUTRETAIN
A 12      DSPATR(BL PC)
A 44      9 2'Choice:'
A 44      CHOICE      20 0 9 12
A 15      9 12'

```

図 18. PUTRETAIN キーワードの使用法を示すサンプル DDS ソース

以下ようになります。

1. 最初の出力操作時に、すべての標識がオフになっているため、すべての定数および CHOICE 以外のフィールドと、CHOICE の後ろにあるフィールドがディスプレイに送信されます。
2. ユーザーが数量を入力します。プログラムが標識 43 をオンに設定します。次の出力操作が実行されると、標識 43 が設定されているため、2 番目の定数フィールドは再送されません。
3. ユーザーが別の項目の数量を入力しようとする、プログラムは別の出力操作を実行します。フィールド QTY、ITEM、COLOR、SHAPE、および SIZE の属性がディスプレイに送信されます。フィールドが選択されると、CHOICE フィールドはディスプレイに送信されません。

レコード域全体が消去されないよう、少なくとも 1 つのフィールド (この場合は QTY) を保持する必要があります。

4. ユーザーが無効な数量、色、形状、またはサイズを入力すると標識 44 がオンに設定され、入力フィールド (QTY、COLOR、SHAPE、および SIZE) は消去されず、出力フィールド CHOICE がディスプレイに送信されます。さらに、適切な標識 9、10、11、または 12 が送信され、エラーのある入力フィールドが明滅し、カーソル位置はそのフィールドの下になります。(CHOICE フィールドは、エラーのあるフィールドの有効な選択項目をユーザーに示します。)
5. CHOICE フィールドおよびブランクの定数フィールドは、同じ場所に定義されます。ユーザーが有効なデータを入力すると、標識 15 がオンに設定され、標識 44 がオフに設定され、定数フィールドは CHOICE フィールドをすべてブランクに初期化します。

読み取り要求の実行時までの書き込み操作の据え置き

ディスプレイ・ファイルの作成 (CRTDSPF) またはディスプレイ・ファイルの変更 (CHGDSPF) コマンドに DFRWRT パラメーターを使用すると、システムが書き込み操作をどのように扱うかを指定できます。DFRWRT(*NO) を指定すると、書き込み操作がデータを表示して入出力フィールドバック域を更新するまで、制御はプログラムに戻りません。

ファイルにデフォルトの DFRWRT(*YES) を指定すると、出力レコードが処理された後で制御がプログラムに戻ります。その後、プログラムは、出力が保管されたレコード域を使用して、次の書き込みまたは書き込み・読み取り操作の処理を開始します。データが実際にディスプレイに送信されるのは、読み取りまたは書き込み・読み取り操作が実行される時、または書き込み専用操作で FRCDDTA DDS キーワードが有効になっている場合だけです。

ディスプレイ・ファイルで DFRWRT(*YES) を使用するとシステム・パフォーマンスは向上しますが、以下の状況では DFRWRT(*YES) を使用しないでください。

- 書き込み操作が成功したかどうかをすぐに調べる場合。DFRWRT(*YES) が指定されたファイルへの書き込み操作に関連するエラーは、データが実際にディスプレイに送信される時にのみ出されます。
- 書き込み操作と読み取りまたは書き込み・読み取り操作の間の時間が長い。例えば、プログラムが書き込み操作後に (読み取りまたは書き込み・読み取り操作を実行する前に) 複数のデータベース操作を実行する場合、データベース操作の実行中にユーザーに対してデータは表示されません。
- 書き込み専用操作後にファイルを閉じる場合で、KEEP キーワードが指定されていない場合。ディスプレイ・ファイルのレコードに DDS キーワード KEEP が指定されている場合、書き込み専用操作から集計されたデータはファイルのクローズ時に表示されます。ただし、KEEP キーワードが指定されていない場合、データは表示されません。

DFRWRT パラメーターは、以下に対して影響を与えることはありません。

- ユーザー定義のデータ・ストリームを使用する書き込み操作
- プログラム記述データを使用するディスプレイ・ファイルに対する書き込み操作
- FRCDDTA DDS キーワードが有効になっているレコード様式

フィールドのデフォルト値の指定

DFT と DFTVAL の両方のキーワードを使用して、フィールドに表示するデフォルト値を指定できます。ただし、2 つのキーワードを使用する方法は異なります。

DFT キーワードは、定数、入力、出力、および出力/入力フィールドで使用することができ、オプションを指定できません。出力または出力/入力フィールドで使用する場合は、OVRDDTA および PUTOVR キーワードも指定する必要があります。レコードがディスプレイ上にない場合、キーワードをこのように組み合わせると、デフォルト値が画面上に表示されます。レコードがすでにディスプレイ上にある場合、PUTOVR キーワードが有効になり、デフォルト値ではなくプログラムのデータがディスプレイに表示されます。

DFTVAL キーワードは、出力および出力/入力フィールドでのみ使用することができ、オプションを指定できません。出力操作で有効になっている場合は、プログラムの値ではなくキーワードの値がフィールドに置かれます。レコードがディスプレイ上にあり、PUTOVR および OVRDDTA キーワードが有効になっている場合、デフォルト値ではなくプログラム値が使用されます。

DFT および DFTVAL キーワードを同じフィールドに指定できません。

レコードを表示するモードの指示

一部のディスプレイ装置 (3180-2 ディスプレイ装置など) は、代替画面サイズをサポートします。この代替サイズを指定するには、DSPMOD キーワードを使用します。DSPMOD キーワードは、特定のレコードに対して、レコードを表示するためのモードを指示します。DSPMOD キーワードが指定されていないレコードは、デフォルトの表示モードで表示されます。デフォルト・モードは、DSPSIZ キーワードの *DS3 または *DS4 ディスプレイ・サイズのうち最初に指定されている方です。

DSPMOD キーワードは、*DS3 と *DS4 の両方を DSPSIZ キーワードに指定する場合にのみ有効です。このキーワードは、レコード・レベルでのみ有効です。オプション標識を使用できます。DSPMOD キーワードをレコード内で重複させることはできません。

注: 27 × 132 モードで表示する機能は、ローカル・ディスプレイ装置コントローラーに接続された、または 5294 または 5394 コントローラーにリモート接続された 3180-2、3197、3477 モデル FA、FC、FD、FE、FG、FW、および 3487 モデル HE、HD、HW、HC ディスプレイ装置でのみ使用できます。これらのコントローラーが使用されない限り、DSPMOD キーワードは無視されます。

例えば、以下の DDS は、RECORD1 を 27 × 132 モード、RECORD2 を 24 × 80 モード (DSPSIZ キーワードで設定されるデフォルト・モード) で表示します。RECORD3 はオプション標識 03 がオンになっている場合は 27 × 132 モード、オプション標識 03 がオンになっていない場合は 24 × 80 モードで表示されます。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     DSPSIZ(*DS3 *DS4)
A           R RECORD1                 DSPMOD(*DS4)
A           R RECORD2
A           R RECORD3
A 03                                     DSPMOD(*DS4)
```

図 19. DSPMOD キーワードの使用法を示すサンプル DDS ソース

DSPMOD キーワードを使用すると、表示モードが大幅に変更される可能性があります。DSPMOD キーワードを指定する場合は注意してください。DSPMOD がアクティブになっているレコードによってモードが変更される場合は、現在ディスプレイ上にあるすべてのレコードは消去され、アクティブ・レコード・テーブルから削除されます。その後、DSPMOD がアクティブになっているレコードがディスプレイに送信されます。このレコードのモードは、DSPMOD キーワードがアクティブになっている限りディスプレイで維持されます。DSPMOD をオフに設定するか、DSPMOD を指定していない別のレコードに対する書き込み操作によって、表示モードはディスプレイ装置の基本表示画面サイズに戻されます。

前のサンプル DDS ソースを使用して DSPMOD キーワードを指定すると、レコードが以下の順に画面に書き込まれる場合に、以下の結果になります。

- RECORD1 が *DS4 モードで表示される。
- 表示画面が消去され、RECORD2 が *DS3 モードで表示される。
- 標識 03 がオフの場合、RECORD3 が *DS3 モードで表示される。OVERLAY キーワードが指定されている場合、RECORD2 はディスプレイに表示されたままになる。
- 標識 03 がオンの場合、RECORD2 が消去され、RECORD3 が *DS4 モードで表示される。

注: 表示モードを変更すると、表示されているサブファイル・データがディスプレイから除去されます。ただし、サブファイル・データはサブファイル・テーブルから消去されません。

表示モードが変更されると、以下のキーワードは無視されます。

表 11. 表示モードが変更される場合に無視されるキーワード

キーワード	追加情報
ALWROL	レコードが画面上にない場合はロールできません。

表 11. 表示モードが変更される場合に無視されるキーワード (続き)

キーワード	追加情報
ASSUME	ASSUME キーワードを指定したレコードは、ファイルが開かれるときに画面に残ります。表示モードが変更されると、画面上のレコードは消去されます。これは、OVERLAY キーワードを指定しないで ASSUME キーワードを指定することと似ています。KEEP キーワードを指定したファイルのディスプレイ・サイズは、ASSUME キーワードを指定したファイルのディスプレイ・サイズと等しくなります。
CLRL	表示モードが変更されると、すべての行が消去されます。
ERASEINP/INZINP	表示モードが変更されると、表示モードの変更前からレコードが画面上にあった場合でも、PUTOVR が無効になった状態でレコードが表示されます。
ERRMSG	
ERRMSGID	
KEEP	
OVERLAY	
PROTECT	
PUTOVR	
PUTRETAIN	-
SFLMSG	
SFLMSGID	

出力操作後のカーソルの位置決め

出力操作後にカーソルを配置する場所を指定するには、CSRLOC または DSPATR(PC) キーワードを使用します。

レコード・レベル・キーワード CSRLOC では、カーソル位置の正確な行および位置が入る 2 つの 3 バイト・ゾーン 10 進数隠しフィールドの名前を指定できます。CSRLOC キーワードを指定すると、表示しているレコードの外側にカーソルを配置できます。

フィールド・レベル・キーワード DSPATR(PC) は、レコードが書き込まれた後のフィールドの最初の位置にカーソルを配置します。ただし、OVERLAY キーワードをレコード・レベルで指定する場合、後続の書き込み操作後にカーソル位置は失われる可能性があります。

注: 出力操作前にキーボードがロック解除される場合、カーソルは配置されません。

CSRLOC と DSPATR(PC) の両方のキーワードを指定する場合、カーソルは CSRLOC キーワードによって配置されます。複数のフィールドに DSPATR(PC) キーワードを指定する場合、DSPATR(PC) キーワードが指定されている最初のフィールドに配置されます。

CSRLOC と DSPATR(PC) の両方のキーワードを指定しない場合、ディスプレイ上の最初の入力可能フィールドに配置されます。入力可能フィールドがない場合、カーソルはディスプレイの左上隅に配置されます。ただし、入力可能フィールドを含むレコードに CSRLOC と DSPATR(PC) の両方のキーワードを指定しない場合、レコードが中断状態になってから復元されると、カーソル位置は失われる可能性があります。例えば、レコードの表示後に F1 (ヘルプ) キーが押されると、カーソル位置は失われる可能性があります。

他の DDS 機能は、書き込み操作に影響を与える可能性があります。ユーザー定義のデータ・ストリーム (USRDFN キーワード) に対する書き込み操作の場合、実行される機能はユーザー提供コントロールによって決まります。

アプリケーションへのカーソル位置の返送

RTNCSRLOC (カーソル位置戻し) キーワードを使用して、入力時のカーソル位置を判別できます。

このキーワードは、以下の 2 通りの形式で指定できます。

- カーソルが現在置かれているレコードおよびフィールドの名前を戻す。オプションで、フィールド内の相対カーソル位置が入る 3 番目のパラメーターを指定できます。
- ディスプレイに対して相対的なカーソルの行および列の位置を戻す。オプションで、2 つの追加パラメーターで、以下のいずれかを戻すことができます。
 - アクティブ・ウィンドウに対して相対的なカーソルの行および列の位置 (存在する場合)
 - 2 イベント・マウス・ボタン呼び出しの開始時のカーソルの位置

これらの形式のパラメーターについては、iSeries Information Center の **DDS** トピックで説明されています。

アプリケーションへのサブファイル内のカーソル位置の返送

入力時に、SFLCSRNRN (サブファイル・カーソル相対レコード番号) キーワードを使用して、サブファイル内のカーソル位置を判別できます。カーソルのある相対レコード番号が、キーワード上でパラメーターとして指定されている隠しフィールドに戻されます。このフィールドは、長さが 5 で小数点以下の桁数がゼロの符号付き数値 (位置 35 が S) として、レコード様式内で定義する必要があります。また、これは隠しフィールド (位置 38 が H) でなければなりません。

アプリケーションへのサブファイルのモードの返送

SFLMODE (サブファイル・モード) キーワードを使用して、入力時にサブファイルが継続モードと切り捨てモードのどちらになっていたかを判別できます。モード・パラメーターは必須です。SFLMODE キーワードはサブファイル制御レコードでのみ有効で、SFLCTL キーワードを指定する必要があります。

モード・パラメーターに指定されるフィールドは、1 文字 (位置 35 が A) の隠しフィールド (位置 38 が H) としてレコード様式に定義されます。フィールドは、サブファイルが継続される場合は 0 の値、サブファイルが切り捨てられる場合は 1 の値で戻されます。SFLCTL (サブファイル制御) レコードに SFLDROP (サブファイル切り捨て) または SFLFOLD (サブファイル継続) を指定していない場合、値は常に 0 として戻されます。

出力/入力フィールドの初期化

装置サポートは、現在ディスプレイ上にあるレコードの入力可能フィールドから読み取られたすべてのデータを保管域に保管します。この保管域内の出力/入力フィールドは、出力操作時に更新されます。

出力操作では、以下のようになります。

- 入力専用フィールドは、ゼロ (数字フィールド)、ブランク (文字フィールド)、またはディスプレイ・ファイルのデフォルト値 (DFT キーワード) に初期化されます。
- 出力/入力フィールド、隠しフィールド、およびプログラム-システム間フィールドは、出力バッファの内容に初期化されます。この出力操作がレコード初期設定機能 (INZRCD キーワード) による場合は、使用可能な出力バッファはありません。出力/入力フィールドおよび隠しフィールドは、入力専用フィールドと同じ様に初期化されます。出力/入力フィールドは、入力および出力が可能です。
- 出力専用フィールドは、サブファイル・レコードの一部になっていない場合、入力バッファの一部ではありません。この場合、出力専用フィールドは、出力/入力フィールドであるかのように保管されます。
- このレコードのすべての応答標識はオフに設定されます。

注: 入力可能フィールドの場合、PUTRETAIN または ERASEINP キーワードが有効な場合、フィールドの保管域は未変更のままです。

書き込み操作中は、入力バッファも出力バッファも変更されません。

ディスプレイに対する入力の送信勧誘

送信勧誘操作は、入力要求をディスプレイ装置に送信し、入力の到着を待機せずにプログラムに戻るために使用されます。これにより、プログラムは 1 つ以上のディスプレイ装置から入力を要求し、ディスプレイ装置が応答するのを待機せずに処理を継続します。プログラムで入力を処理する準備が整うと、送信勧誘された装置からの読み取り操作を実行することにより、どの送信勧誘されたディスプレイ装置からでもデータを受信できます。

送信勧誘操作は、INVITE DDS キーワードが有効になっているレコード様式を使用して書き込み操作を実行することにより行われます。書き込み操作を実行する方法および標識を使用して INVITE DDS キーワードを制御する方法については、該当する高水準言語の資料を参照してください。

ディスプレイ装置が送信勧誘されると、送信勧誘された装置からの読み取り操作、および特定のディスプレイ装置に送信される読み取り (待機) 操作が、ディスプレイ装置からデータを受信するための有効な操作になります。送信勧誘されたディスプレイ装置に対する送信勧誘の取り消しも有効な操作です。

ディスプレイ装置を複数装置ディスプレイ・ファイル内の入出力操作に使用するには、その前にディスプレイ装置をファイルに対して獲得する必要があります。プログラムは、ファイルに対して現在獲得されている任意のディスプレイ装置に対して送信勧誘を送信できます。

DFRWRT(*YES) を指定して複数装置ディスプレイ・ファイルを作成した場合は、INVITE DDS キーワードをオンにした出力操作により、ディスプレイ装置が送信勧誘される前に、延期されていた出力が画面に表示されます。

ディスプレイ装置を送信勧誘しても送信するデータがない場合は、INVITE DDS キーワードをオンに設定し、出力可能フィールドを含まないレコード様式で出力操作を実行します。

複数ディスプレイ装置ディスプレイ・ファイルは、ILE RPG、ILE COBOL、ILE C/C++、および CL でサポートされます。

CL プログラムからの入力の送信勧誘

送信勧誘操作は、以下の CL コマンドを介して、CL プログラムに対して直接実行できます。

- *RCVF* および *SNDRCVF* コマンドで *WAIT(*NO)* を指定:

*WAIT(*NO)* を指定すると、入出力操作および実行中のプログラムのオーバーラップ、複数のディスプレイからの入力の要求、および入力の受信 (入力を使用可能になったとき) が可能です。これにより、送信勧誘操作と同等のサポートが提供されます。

非待機オプションを指定した読み取り操作時に、システムは要求をディスプレイに送信してプログラムに戻ります。ただし、要求されたレコードは、制御が戻る時は使用できません。この操作の目的は、プログラムが他の作業を実行している間に、ディスプレイ装置が入力データを送信できるようにすることです。

レコードを検索するには、*WAIT* コマンドを実行します。*WAIT* コマンドは、送信勧誘された装置からの読み取り操作を実行します。プログラムは、ディスプレイ装置でデータが使用可能になるまで、または *WAITRCD* 時間が経過するまで待機します。その後、ディスプレイ装置名および入力データがユーザ

一・プログラムに渡されます。複数の非待機読み取り操作（それぞれ別々のディスプレイに対するもの）が実行され、複数の操作が完了すると、WAIT コマンドは完了した最初の非待機読み取り操作だけを処理します。

WAIT コマンドは、他のそれぞれの非待機読み取り操作を処理するために実行できます。操作は完了順に処理されます。

INVITE キーワードを含むレコードがディスプレイに送信されると、操作は非待機オプションを指定した書き込み・読み取り操作として処理されます。INVITE DDS キーワードは、書き込み・読み取り操作時に無視されます。

非待機オプションを指定した書き込み・読み取り操作は、書き込み操作に続けて非待機読み取り操作を実行するのと同じです。

- **ENDRCV コマンド:**

ENDRCV コマンドは、WAIT(*NO) オプションを指定して実行された入力の要求を終了するために使用されます。ENDRCV コマンドは、ディスプレイ装置でデータが使用可能であるとしても、入力要求を終了します。ENDRCV 操作の実行時に、ディスプレイがデータを送信中の場合、そのデータは失われます。ディスプレイ装置が送信勧誘されていない場合、アプリケーション・プログラムにエラーが送信されます。

ディスプレイからの送信勧誘された入力の読み取り

送信勧誘された装置からの読み取り操作は、送信勧誘されたいずれかのディスプレイ装置からのデータを待機および受信する方法を提供します。ディスプレイ装置を送信勧誘し、送信勧誘されたディスプレイ装置から読み取る方法は、アプリケーションがユーザーの応答を待機する時間を制御する必要がある場合に役立ちます。送信勧誘された装置からの読み取り操作が実行されると、プログラムは、CRTDSPF、CHGDSPF、または OVRDSPF コマンドの WAITRCD キーワードで指定した時間間隔待機します。待機が終了する方法には、以下の方法があります。

- 送信勧誘されたディスプレイ装置でデータが使用可能になる。ディスプレイ装置名、操作の結果、入力データがプログラムに渡される。データが受信されるとディスプレイ装置は送信勧誘されなくなるため、送信勧誘された装置からの読み取り操作がディスプレイ装置からさらにデータを受信する必要がある場合は、送信勧誘操作によってもう一度送信勧誘する必要があります。
- 送信勧誘された装置なし信号。ファイルに関連付けられたディスプレイ装置のうち、送信勧誘された状態になっているものが 1 つもないことを示します。これがプログラムにどのように通知されるかについては、該当する高水準言語の資料を参照してください。
- ジョブ終了制御信号。プログラムが実行されているジョブが、ジョブ終了 (ENDJOB)、システム終了 (ENDSYS)、システム電源遮断 (PWRDWN SYS)、またはサブシステム終了 (ENDSBS) コマンドの制御オプションによって終了することを示します。これがプログラムにどのように通知されるかについては、該当する高水準言語の資料を参照してください。これは、複数装置ディスプレイ・ファイルがどれほど多く使用されていても、プロセスで 1 回だけ発生します。すべての送信勧誘されたディスプレイ装置は、送信勧誘されたままです。
- データが使用可能な送信勧誘された装置なし信号。これは、ファイルに関連付けられたディスプレイ装置でデータが使用可能になっていない場合に、WAITRCD 時間が *IMMED になっていて、上記のいずれの条件にも当てはまらない場合に発生します。送信勧誘されたディスプレイ装置は、送信勧誘されたままです。これがプログラムにどのように通知されるかについては、該当する高水準言語の資料を参照してください。
- 送信勧誘された装置からのデータの待機中にタイムアウト信号。これは、WAITRCD 値が有限の秒数で、その期間内にデータが使用可能にならず、上記のいずれの条件にも当てはまらない場合に発生しま

す。これがプログラムにどのように通知されるかについては、該当する高水準言語の資料を参照してください。送信勧誘されたディスプレイ装置は、送信勧誘されたままです。

同様に、ILE COBOL も、WAITRCD(*IMMED) が指定されているかのように送信勧誘された装置からの読み取り操作を実行する方法を提供します。NODATA 句および送信勧誘された装置からの読み取り操作に対する影響については、ILE COBOL の資料を参照してください。

送信勧誘された装置からの読み取りについての理解

送信勧誘されたいずれかのディスプレイ装置からの入力を処理する準備がプログラムで整うと、プログラムは送信勧誘された装置からの読み取り操作を実行できます。この操作は、送信勧誘されたいずれかのディスプレイ装置から入力が到着するのを指定した時間待機します。制限時間はディスプレイ・ファイルの作成時に指定でき、後で変更またはオーバーライドできます。送信勧誘されたディスプレイ装置が制限時間内に応答しない場合、プログラムは制限時間が満了したため処理を継続できること伝える指示を受け取ります。送信勧誘されたディスプレイ装置が制限時間内に応答する場合、プログラムはどのディスプレイ装置が応答したのか、またデータを処理するために使用されたレコード様式を判別できます。他の送信勧誘されたディスプレイ装置は送信勧誘されたままで、データを送信可能です。応答送信勧誘ディスプレイ装置を別の送信勧誘操作でもう一度送信勧誘することができます。

特定のディスプレイ装置に対して読み取り操作を実行することもできます。この操作は、指定したディスプレイ装置がデータで応答するまで完了しません。読み取り操作を行うためにディスプレイ装置を送信勧誘する必要はありませんが、送信勧誘する場合、プログラムは入力を待機し、ディスプレイ装置は送信勧誘されなくなります。

- 送信勧誘された装置からの読み取り操作は、現在送信勧誘されているディスプレイ装置からのデータだけを受け入れます。
- ディスプレイ・ファイルに対して獲得された複数のディスプレイ装置で送信勧誘が未解決になっている場合、送信勧誘された装置からの読み取り操作は、送信勧誘されたいずれかのディスプレイ装置から次に使用可能なレコードを戻します。送信勧誘された装置からの読み取り操作の前に複数のディスプレイ装置からレコードが受信される場合、他のレコードは、後続の送信勧誘された装置からの読み取り操作または特定のディスプレイ装置に対する後続の読み取り (待機) 操作作用に保持されます。
- ディスプレイ装置が応答し、送信勧誘された装置からの読み取り操作で入力が受信されると、そのディスプレイ装置は送信勧誘されなくなります。別の送信勧誘操作によってもう一度送信勧誘することは可能ですが、これは入力可能フィールドを持つディスプレイ上のすべてのレコード様式が読み取られるまで行うべきではありません。
- 送信勧誘された装置からの読み取り操作でレコード様式を指定することはできません。ディスプレイから戻されるレコード様式は、ディスプレイ装置に書き込まれる最後のレコード様式と同じです。
- WAITRCD パラメーターに関連付けられているタイミング機能は、システムがヘルプ・キーを処理中の場合、待機の終了を強制しません。以下の場合、送信勧誘された装置からの読み取り機能は、ユーザーがヘルプ情報を閉じるまで終了しません。
 - システムが、DDS の H 指定で定義されているディスプレイ・ファイルのヘルプを表示している。
 - システムが、ディスプレイ装置がジョブの要求側ディスプレイ装置であり、ディスプレイ・ファイルが MAXDEV(1) を指定している場合にメッセージのヘルプを表示している。

CRTDSPF または CHGDSPF コマンドの MAXDEV パラメーターに 1 より大きい値を指定することによって、WAITRCD 時間の満了時にメッセージ・ヘルプの終了を強制できます。

CL プログラムからの送信勧誘された装置からの読み取り: 送信勧誘された装置からの読み取り操作は、以下の CL コマンドを介して、CL プログラムに対して直接実行できます。レコードを検索するには、WAIT コマンドを実行します。WAIT コマンドは、送信勧誘された装置からの読み取り操作を実行しま

す。プログラムは、ディスプレイ装置でデータが使用可能になるまで、または WAITRCD 時間が経過するまで待機します。その後、ディスプレイ装置名および入力データがユーザー・プログラムに渡されます。複数の非待機読み取り操作 (それぞれ別々のディスプレイに対するもの) が実行され、複数の操作が完了すると、WAIT コマンドは完了した最初の非待機読み取り操作だけを処理します。

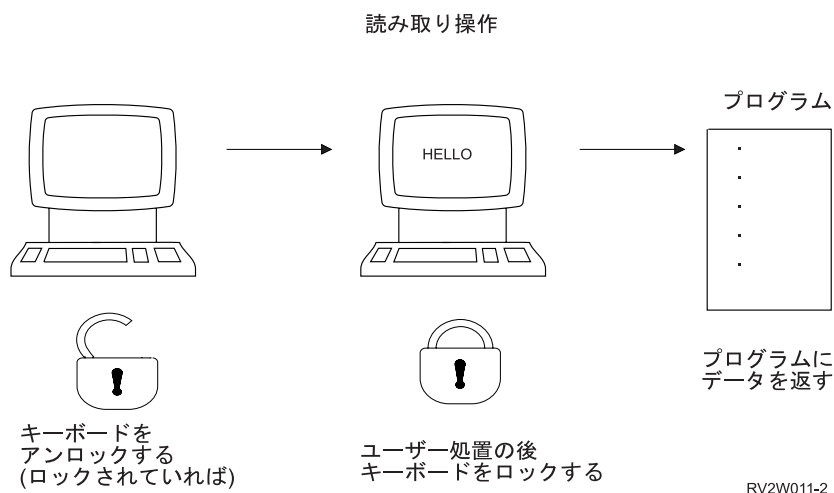
WAIT コマンドは、他のそれぞれの非待機読み取り操作を処理するために実行できます。操作は完了順に処理されます。

INVITE キーワードを含むレコードがディスプレイに送信されると、操作は非待機オプションを指定した書き込み・読み取り操作として処理されます。INVITE DDS キーワードは、書き込み・読み取り操作時に無視されます。

非待機オプションを指定した書き込み・読み取り操作は、書き込み操作に続けて非待機読み取り操作を実行するのと同じです。

ディスプレイからの入力の読み取り

読み取り操作は、レコードをシステムからプログラムに渡します。ディスプレイ・ファイルのレコード様式には、システムがレコードを処理するのに必要な情報が入っています。ユーザーは、データをシステムに渡すために Enter キーまたは機能キーを押すなどの必要なアクションを実行する必要があります。読み取り操作が実行されると、以下ようになります。



入力操作では、以下ようになります (この順序で)。

1. 入力専用操作の場合、このレコードのすべての応答標識がオフに設定され、読み取り操作が実行されます。
2. ディスプレイから受信される文字フィールドが右揃えまたは左揃えされ、必要に応じて空白が埋め込まれるか切り捨てられます。デフォルトは左寄せですが、これは AUTO または CHECK キーワードを使用してオーバーライドできます。
3. ディスプレイから受信される数字フィールドで、以下ようになります。
 - a. フィールドが負の場合、1 の位のゾーン部分が D に設定されます (82 ページの『負の数値入力データの処理』を参照)。
 - b. 数字以外のすべての文字は除去され、数字は圧縮されます。
 - c. 符号付き数字フィールドは右寄せされ、数字専用フィールドは小数部で位置合わせされます。
 - d. 必要に応じてフィールドはゼロで埋め込まれるか切り捨てられます。

- e. フィールド妥当性検査が実行されます。

ディスプレイから受信されるすべてのフィールドは、選択されたレコードの一部であるかどうかにかかわらず、この方法で処理されることはありません。

フィールド妥当性検査エラーが検出されると、エラーを訂正できるようメッセージがユーザーに送信されます。このプロセスは、エラーがなくなるまで繰り返されます。その後、要求されたレコードの保管域が入力バッファにコピーされます。

注: レコード様式名を指定していない読み取り操作の入力データを処理するために、ディスプレイ装置サポートは、少なくとも以下のいずれかを含むディスプレイに書き込まれた最後のレコードを使用します。

- 入力専用フィールド
- 出力/入力フィールド
- 隠しフィールド

このような様式がディスプレイ上にない場合は、ディスプレイ装置サポートは、この種のフィールド (例えば、有効なコマンド・キーを指定する出力専用レコード) を含まないディスプレイに書き込まれた最後の様式を使用します。このようなレコードがディスプレイ上に存在する場合、プログラムにエラー・メッセージが戻されます。

レコードがディスプレイに書き込まれていなくても、INZRCD キーワードを指定すると、プログラムがレコードを読み取ることができます。システムは、アプリケーション・プログラムが出力操作を実行するのと同じ方法でこれを実行しますが、以下の例外があります。

- 出力専用フィールドの場合、使用可能なユーザー・データがないため、フィールドはブランクに初期化されます。フィールドが編集されても、編集内容は無視されます。BLKFOLD キーワードを指定しても無視されます。
- 出力/入力フィールドの場合、使用可能なユーザー・データがないため、フィールドはブランクに初期化されます。フィールドが編集されても、編集内容は無視されます。フィールドには、ブランクとして表示されるヌル文字 (16 進数ゼロ) が実際に含まれています。
- 定数または入力専用フィールドの場合、データは通常は出力バッファからのものではないため、フィールドは、プログラムが書き込み操作を使用してフィールドを表示する場合と同じように表示されません。
- 隠しフィールドの場合、フィールドは、読み取り操作時に文字フィールドの場合はブランク (16 進数 40)、数字フィールドの場合はゼロ (16 進数 F0) として戻されます。
- メッセージの場合、メッセージ・データがないため、フィールドは無視されます。
- LOGOUT キーワードは無視されます。
- レコードがまだディスプレイ上にないため、ERRMSG および ERRMSGID キーワードは無視されます。
- SFLMSG および SFLMSGID キーワードは無視されます。

他のすべてのフィールドまたはキーワードは、出力操作時に選択されたかのように処理されます。

プログラムによるデータ処理中のキーボードのロック解除

UNLOCK キーワードを指定すると、前に入力された入力データをプログラムが処理している間にデータを入力フィールドに入力できるようキーボードをロック解除できます。

通常、キーボードがロック解除されるまで、入力フィールドは消去されません。UNLOCK キーワードを指定してあり、GETRETAIN キーワードを指定していない場合にのみ、読み取り操作時に、入力フィールドはキーボードのロック解除後に消去されます。

5250 ディスプレイ装置の場合、UNLOCK キーワードが有効になった状態で読み取り操作を実行すると、以下の結果になります。

1. 5250 ディスプレイ装置が、フィールドに対してハードウェア妥当性検査を実行します。エラーが検出されない場合、以下のようになります。
 - a. GETRETAIN キーワードを指定しないで UNLOCK キーワードを指定している場合、または UNLOCK(*ERASE) キーワードを指定している場合、変更されたすべての入力可能フィールドが消去されます。
 - b. GETRETAIN キーワードを指定して UNLOCK キーワードを指定している場合、または UNLOCK(*MDTOFF) キーワードを指定している場合、すべての変更データ・タグ (MDT) がリセットされます。
 - c. UNLOCK(*ERASE *MDTOFF) キーワードを指定している場合、変更されたすべての入力可能フィールドが消去され、MDT がリセットされます。
 - d. カーソルは、ユーザーが次のレコードを入力できるフィールドに位置変更されます。
 - e. キーボードがロック解除されます。
2. システム妥当性検査の対象は、ディスプレイ上のすべてのレコードのすべてのフィールドです。エラーが検出されると、通常のエラー再試行が実行されます。エラー・メッセージが表示されるときに、ユーザーは次のレコードに入力中である可能性があります。

注: エラー・メッセージは、データが消去されたためにディスプレイ上にないデータに言及している場合があります。

3. 制御がプログラムに戻ります。

注:

1. アプリケーション・プログラムが入力エラーを検出してディスプレイにエラー・メッセージを送信する場合、メッセージは上書きされた入力に言及している場合があります。
2. CHANGE キーワードを指定する場合で、GETRETAIN キーワードを指定しないか UNLOCK(*ERASE) キーワードを指定して UNLOCK キーワードを指定する場合、関連した応答標識が次の入力レコードでオンに設定されます。
3. サブファイルが画面上にある間に、UNLOCK キーワード (GETRETAIN キーワードは指定しない) または UNLOCK(*ERASE) キーワードを指定した読み取り操作をレコードで使用すると、ユーザーがサブファイル・レコードにデータを入力しなかった場合でも、サブファイルに対する後続の次変更読み取り操作時にサブファイル・レコードがプログラムに戻されることがあります。UNLOCK キーワード (GETRETAIN キーワードは指定しない) または UNLOCK(*ERASE) キーワードではなく、UNLOCK(*ERASE *MDTOFF) キーワードを使用することをお勧めします。これらのキーワードのいずれかを使用する必要がある場合は、すべてブランクの未変更のフィールドがプログラムに戻される可能性があること考慮して、高水準言語プログラムがブランクを比較するようにしてください。

入力データの保持

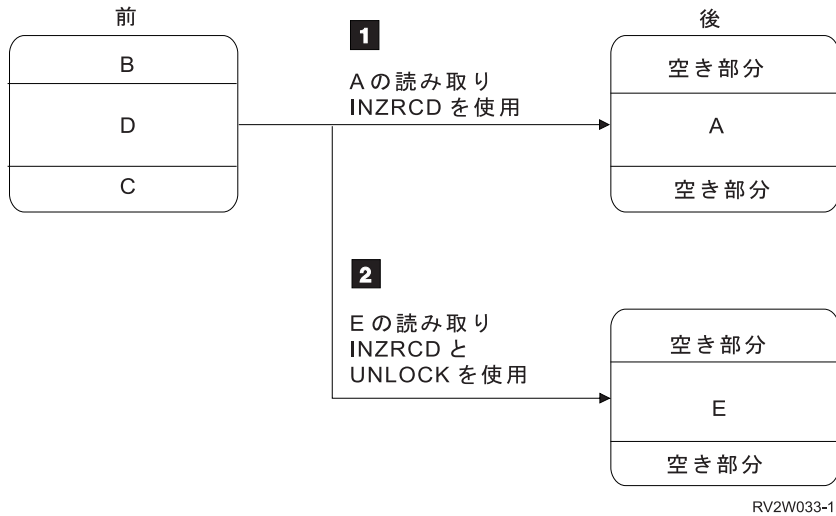
GETRETAIN キーワードを指定すると、ユーザーが Enter キーを押した後もディスプレイ上の入力データを保持できます。GETRETAIN キーワードは、常に UNLOCK キーワードと一緒に使用する必要があります。

データ変更時の標識の設定

CHANGE キーワードを指定すると、データが入力フィールドに入力される時、または出力/入力フィールドでデータが変更される時に応答標識を設定できます。

レコードの初期化とキーボードのロック解除の図

以下の図は、入力操作に対する INZRCD および UNLOCK キーワードの影響を示しています。



注: レコード様式 A、D、および E は、同じ行を占有します。

- 1 レコード様式 B、D、および C は、レコード様式 A に OVERLAY キーワードを指定しない場合は消去されます。レコード様式 A は、定数および初期化された入力フィールドと一緒に表示されます。キーボードがロック解除されます。キーボードは、ユーザーが取得操作を実行した後でロックされます。
- 2 レコード様式 B、D、および C は、レコード様式 E に OVERLAY キーワードを指定しない場合は消去されます。レコード様式 E は、定数および初期化された入力フィールドと一緒に表示されます。キーボードがロック解除されます。ユーザーが読み取り操作を実行した後で、入力フィールドの内容が消去され、キーボードがもう一度ロック解除されます。

注: UNLOCK キーワードを指定してある場合でも、フィールド妥当性検査 (指定してある場合) およびコマンド・キー妥当性検査は実行されます。したがって、エラー・メッセージがディスプレイに表示されるときに、ユーザーは次のレコードに入力できます。

妥当性検査機能の指定

ユーザーが入力したデータの妥当性検査では、以下の 2 つの方法を使用できます。

- データがアプリケーション・プログラムに渡される前にシステムでデータを検査する。
- すべての入力データをアプリケーション・プログラムに渡して、アプリケーション・プログラムでデータの妥当性検査を実行する。

いずれの場合も、エラーが検出されると、エラーを訂正できるよう、ユーザーにエラーを通知するメッセージが表示されます。エラーを検出するために 2 番目の方法を選択する場合、プログラムがどのようにエラー・メッセージを表示できるかについては、247 ページの『独自のメッセージの作成および表示』を参照してください。以下で、最初の方法 (データがプログラムに渡される前にシステムがエラーを検出する場合) の詳細について説明します。

DDS で指定できる妥当性検査機能は、以下のとおりです。

- 少なくとも 1 文字を入力する必要があるフィールドの検出 (CHECK(ME) キーワード)。空白は有効な文字です。これを入力必須 といいます。

- すべての桁に文字が含まれている必要があるフィールドの検出 (CHECK(MF) キーワード)。ブランクは有効な文字です。これを全桁入力必須 といいます。
- 文字、数字、または符号付数値データが必須である間違っただデータ・タイプの検出。
- フィールド用に指定した範囲内にはないデータの検出 (RANGE キーワード)。
- 入力されたデータと指定された定数値の比較の実行 (COMP キーワード)。
- 有効な項目の特定のリストに入力されたデータの比較 (VALUES キーワード)。
- 文字フィールドに有効なフィールドまたはレコード名が入力されたかどうかの検出 (CHECK(VN) キーワード)。
- 文字フィールドに有効なオブジェクト名が入力されたかどうかの検出 (CHECK(VNE) キーワード)。
- モジュラス 10 または 11 チェック・ディジット検査の実行 (CHECK(M10) または CHECK(M11) キーワード)。(指定できるのは 1 つだけです。)
- 入力が行われていないかのようにブランク・キー入力を処理できるようにする (CHECK(AB) キーワード)。フィールドにサブファイル・キーワード SFLROLVAL または SFLRCDNBR も指定されている場合、CHECK(AB) (ブランク使用可能) は無視されます。
- スペース、正符号、または負符号が数字フィールドの数字に埋め込まれているかどうかの検出。また、数字フィールドの数字の前に正符号または負符号が付いているかどうかの検出。これを検出するには、数字妥当性検査 (VALNUM) キーワードを使用します。

妥当性検査キーワード CHECK(M10 M11 VN VNE)、COMP、RANGE、および VALUES に加えて ERRSFL キーワードを使用すると、キーワードに関連付けられた複数のエラー・メッセージを同時に表示できます。

妥当性検査用に RANGE、COMP/CMP、VALUES、CHECK(VN)、CHECK(VNE)、CHECK(M10)、または CHECK(M11) キーワードを指定して、これらの妥当性検査機能のいずれかでエラーが検出されると、以下のようになります。

1. キーボードがロックされます。
2. エラーのあるすべてのフィールドが反転表示されます。エラーのあるフィールドに下線 (UL) 表示属性と強調表示属性 (HI) の両方があっても、この属性の組み合わせは DSPATR(ND) と同じ影響があるため、イメージは反転表示されません。
3. カーソルが、エラーのある最初のフィールドの先頭に配置されます。
4. エラーのある最初のフィールドのシステム提供エラー・メッセージが、エラー行に表示されます。

または

CHKMSGID キーワードを使用してフィールドの独自のエラー・メッセージを提供する場合で、これがエラーのある最初のフィールドの場合、独自のエラー・メッセージがエラー行に表示されます。

コントローラーに自己検査機能をインストールしてある場合 (「5250 Functions Reference」を参照)、コントローラーは CHECK(M10F) および CHECK(M11F) キーワードの妥当性検査を実行します。エラーは、Enter キーまたはコマンド・アテンション・キーを押すときではなく、入力フィールドからカーソルを移動しようとするときに検出されます。システム提供メッセージまたはユーザー指定メッセージではなく、オペレーター・エラー・コード 00115 が、ディスプレイの左下隅に表示されます。USRDSMGT キーワードも指定する場合、CHECK(M10) および CHECK(M11) は、CHECK(M10F) および CHECK(M11F) として機能します。

RANGE、COMP、VALUES、CHECK(VN)、または CHECK(VNE) キーワードをフィールドに指定する場合で、そのフィールドにデータを入力する場合、ユーザーがエラー後にフィールドの復元を試行するかどうか

にかかわらず、フィールドは変更されたことを示します。ブランク (文字フィールドの場合) またはゼロ (数字フィールドの場合) が妥当性検査機能で不合格になる場合は、CHECK(AB) キーワードを使用してください。これにより、妥当性検査機能がうまく機能します。

サブファイルの一部であるレコードの妥当性検査を指定すると、ディスプレイからサブファイルに配置される前にレコード内の各フィールドが妥当性検査されます。エラーのあるすべてのフィールドを訂正するまで、レコードをロールできません。

フィールドがユーザーによって変更される場合、または DSPATR(MDT) を使用して変更データ・タグ (MDT) がオンに設定される場合、システムはフィールドに対して妥当性検査のみを実行します。

注:

1. ユーザーが複写キーを押すと、フィールドの妥当性検査は無視されます。DUP キーワードを指定すると、ユーザーが複写キーを使用できるようになります。
2. COMP、VALUES、または RANGE キーワードが指定されている数字フィールドの値は、フィールドに指定した小数部の桁に基づいて桁合わせされ、必要に応じてゼロが埋め込まれます。フィールドの小数部の桁を入力しなかった場合、小数点は値の一番右の数字の右側にあると見なされます。例えば、長さが 5 で小数点以下の桁数が 2 の数字フィールドの場合、値 1.2 は 001.20 と解釈され、値 100 は 100.00 と解釈されます。
3. 入力フィールドの妥当性検査に RANGE キーワードを使用して入力フィールドにブランクを入力すると、入力フィールドの値は範囲の要件を満たしていない場合があります。ブランクは、数字フィールドの場合はゼロに変換され、文字フィールドの場合はブランクとして渡されます。フィールド・レベル・キーワード BLANKS を使用して、どんな場合にフィールドをすべてブランクとして表示するかを決定します。ユーザーがブランクを入力すると、BLANKS キーワードの応答標識はオンに設定されます。

入力可能フィールドの数の制限についての理解

リモート 5250 ディスプレイ装置 (リモート・コントローラーを介して接続されるディスプレイ装置) の場合、コントローラー・モードに応じて 1 つのディスプレイに 126 または 256 の入力フィールドを指定できます。(5294 コントローラーは 126 の入力フィールドをサポートし、5394 コントローラーは 256 の入力フィールドをサポートします。)さらに、DSPATR(OID) または DSPATR(SP) を指定する場合、この最大数はこれらのキーワードの 3 つのインスタンスを使用するたびに 1 ずつ減ります。インスタンスが 3 つより少ない場合でも、1 減ります。

ローカル 5250 ディスプレイ装置 (ローカル・ディスプレイ装置コントローラーを介して接続されるディスプレイ装置) の場合、256 の入力フィールドを指定できます。また、DSPATR(OID) または DSPATR(SP) を指定する場合、この最大数はこれらのキーワードの 3 つのインスタンスを使用するたびに 1 ずつ減ります。さらに、ローカル 5250 ディスプレイ装置で磁気ストライプ読取装置を使用する場合もフィールドの最大数が減ります。フィールドの最大数は、以下のように計算されます。

$$256 - \left\lceil \frac{3+B}{6} + \frac{A}{3} \right\rceil \text{ 整数に切り上げ}$$

RSLH131-2

A はディスプレイ上の DSPATR(OID) および DSPATR(SP) フィールドの数、B は予想される最長の磁気ストライプ入力 (最大許容データ文字数は 125) です。DSPATR(OID) として指定されていない磁気ストライプ・データを入力フィールドに入力できます。

上記のいずれかの場合に入力フィールドの最大数を超えると、使用プログラムにメッセージ CPF5192 が発行されます。

フィールドおよびレコード様式の数プログラムが実行されるまで分からないため、ディスプレイ・ファイルの作成中はフィールド最大数診断は行われません。

サブファイル・レコードが表示されると、ディスプレイに送信される入力可能フィールドの実際数は、レコードに定義されている数と表示されているサブファイル・レコードの数を乗算した数です。

リモート接続された 3270 ディスプレイの場合、126 の入力フィールドに制限されます。

プロトコル変換装置を介して接続される ASCII ディスプレイの場合、制限は接続先のコントローラーと同じです。

負の数値入力データの処理

数値入力データの負の符号は、以下の 3 つの形式で表示できます。

- - (マイナス) キーを使用して符号を入力する場合は、16 進数 60
- フィールド・マイナス・キーを使用して符号を入力する場合は、16 進数 D
- D ゾーンで英数字として符号を入力する場合は、16 進数 Dn

16 進数 60 は、最下位数字の右側にある場合は負符号として扱われます。

16 進数 D ゾーンは、最下位数字の場合は負符号として扱われます。さらに、数値部分と等しい値の有効数字として扱われます。

組み込みブランク (有効数字間) は、小数点位置合わせの前にゼロに変更されます。

システムがディスプレイから入力を読み取る方法についての理解

読み取り操作が実行されると、システムはディスプレイ上のすべてのレコードを読み取ります。ただし、各読み取り操作でプログラムに渡されるレコードは 1 つだけです。さらなる読み取り操作に備えて、システムは他のすべてのレコードを保管します。

各読み取り操作がディスプレイ上の別のレコードを参照する場合、ユーザーのアクションは不要です。ただし、各読み取り操作がディスプレイ上の別のレコードではなく同じレコードを参照する場合で、RTNDTA キーワードを指定しない場合、入力される各レコードはプログラムに 1 回だけ渡されるため、次の読み取り操作を開始するために、ユーザーは Enter キーまたは CFnn キーを押すなどのアクションを実行する必要があります。RTNDTA キーワードを指定する場合、レコードの前の読み取り操作時にプログラムに戻されたのと同じ入力バッファがもう一度戻されるため、ユーザーのアクションは不要です。

システムは、ディスプレイ上でアクティブなレコードの入力可能フィールドの内容を保管します。この保管されたデータはユーザー・プログラムに渡され、以下のように変更できます。

- 書き込み操作時にデータを定数で初期化する。フィールドは、DFT キーワードに指定する値で初期化できます。
- データを直接入力するか、ライト・ペンを使用してデータを選択する。(フィールドの MDT をオンに設定して、ユーザー入力をシミュレートできます。)
- 書き込み操作時にプログラムからデータを入力する。これは、出力/入力フィールド (およびサブファイルの出力専用フィールド) に適用されます。
- PUTRETAIN キーワードを指定していない限り、同じレコードの出力操作時にブランク (文字フィールド) またはゼロ (数字フィールド) でデータを初期化する。これは、入力専用フィールドに適用されません。

出力書き込みおよび入力読み取りの同時実行

書き込み・読み取り操作は、CL プログラムの SNDRCVF コマンドなどの高水準言語ステートメントを使用して、同じレコード様式に対して書き込み操作と読み取り操作を組み合わせて実行することです。これは、書き込み操作の直後に読み取り操作を指定したかのように動作します。

一部の高水準言語には、ディスプレイへの情報の書き込みとユーザー応答の読み取りを 1 つのステートメントで実行する書き込み・読み取り操作があります。例えば、ILE RPG には、様式実行 (EXFMT) 操作があります。この種の操作は、新しい情報をディスプレイに提供すると同時に、ユーザーに情報を要求する必要がある場合に役立ちます。同じレコード様式に対する書き込み操作の直後に読み取り操作を実行することによっても、書き込み・読み取り操作の組み合わせをサポートしない言語でこの操作をシミュレートすることができます。

この操作を実行すると、以下のようになります。

1. プログラムがシステム・ディスプレイ・サポートを呼び出し、ディスプレイに表示するデータと、そのデータの書き込みおよび読み取り時に使用するレコード様式を提供します。
2. システムがレコード様式内で検出される情報とデータを結合し、ディスプレイに送信するデータ・ストリームを構成します。
3. データ・ストリームがディスプレイに送信され、キーボードがロック解除されます。
4. ユーザーが入力可能なフィールドにデータを入力し、Enter キーまたは他の機能キーを押します。
5. データがディスプレイからシステムに送信されます。システムは、そのデータをデコードし、アプリケーション・プログラムが知る必要のある情報だけを抽出し、そのデータをアプリケーション・プログラムに戻します。

処理するレコード様式が 1 つだけの場合、このような書き込み・読み取り作業は最も一般的な作業です。この操作の書き込み部分で、ユーザーに対して表示するデータを提供します。読み取り部分では、ユーザーが入力または変更したデータを受け取ります。

待機されなかった入力の取り消し

送信勧誘取り消し操作は、送信勧誘操作で前に送信勧誘されたディスプレイ装置に対して実行された入力要求を取り消すために使用します。入力要求を取り消すには、送信勧誘されたディスプレイ装置に対して書き込み操作を実行します。以下のいずれかのようにになります。

- 送信勧誘操作からの入力要求にユーザーが応答する前に書き込み要求が受信される場合、入力要求は取り消され、書き込み操作で指定するレコード様式がディスプレイ装置に送信されます。レコード様式で DDS キーワード INVITE のオプション標識をオンに設定している場合、ディスプレイ装置はもう一度送信勧誘されます。
- 送信勧誘操作からの入力要求にユーザーが応答した後で書き込み要求が受信される場合、入力要求は取り消されず、書き込み操作は失敗します。使用可能なデータを受信するには、送信勧誘された装置からの読み取り操作または読み取り (待機) 操作を実行する必要があります。

ディスプレイ装置を解放すると、ディスプレイ装置に対する入力要求も暗黙的に取り消されます。ディスプレイ装置に使用可能なデータがあっても、そのデータは失われます。

入出力操作中のキーボードのロックおよびカーソルの位置決め

書き込み、書き込み・読み取り、または読み取り操作が実行されるときに、キーボードがどうなるかを以下に示します。

操作	キーボード
書き込み	キーボードはデフォルトでロック解除されます。LOCK キーワードが指定されている場合、キーボードはロック解除されません。
書き込み・読み取り 読み取り	キーボードがロック解除されます。 キーボードは、ディスプレイ装置ユーザー・アクションの前にロック解除されます (ロックされている場合)。ユーザー・アクション後にキーボードはデフォルトでロックされません。UNLOCK キーワードが指定されている場合、キーボードはロック解除されたままです。

キーボードがロック解除されるたびに、カーソルは位置変更されます。ある場合、読み取り操作の間に書き込み操作を多数実行すると、カーソル移動が不安定になる可能性があります。最後の書き込み操作前にユーザーが入力を開始すると、キーボードのロック解除時にカーソルが位置変更されるため、ユーザーは混乱する可能性があります。LOCK キーワードを使用することによって、この状態が生じないようにすることができます。最後の書き込み操作以外のすべての書き込み操作に LOCK キーワードを使用することによって、キーボードは最後の書き込み操作までロックされたままになります。これにより、不安定なカーソル移動を回避できますが、ユーザーはデータの入力を開始できません。

通常、有効なコマンド・キーを押すなどのユーザー・アクションによってキーボードはロックされます。

次の入力操作時にシステムがキーボードをロック解除しないようにするには、ロック状況の保持 (RETLCKSTS) キーワードを指定します。このキーワードを指定すると、入力操作が開始されてデータがキーボードからすでに伝送されている場合に、データが失われないようにすることができます。

注: RETLCKSTS キーワードは、キーボードがすでにロック解除されている場合にのみ使用してください。

DSPATR(PC)、CSRLOC、または SFLRCDNBR(CURSOR) キーワードでカーソルを配置するには、キーボードがロックされている必要があります。出力操作時にキーボードがロックされるのは、以下の条件が満たされるときだけです。また、ディスプレイ装置がカーソルを配置する場合も、以下の条件が満たされている必要があります。(通常、出力操作では、LOCK キーワードが指定されていない限り、操作が終了する前にキーボードをロック解除するため、以下の条件は一時的にキーボードをロックするに過ぎません。)

- 入力フィールドが消去される (ERASEINP キーワード)。
- 変更データ・タグがリセットされる (MDTOFF キーワード)。
- 入力可能フィールドがディスプレイに書き込まれる。
- ディスプレイ全体が消去される (OVERLAY キーワードを指定しない書き込み操作)。
- 以下の結果として 5250 様式バッファがリセットされる。
 - 入力可能フィールドのあるレコード様式がオーバーレイまたは消去される。
 - カーソル位置指定のあるレコード様式がオーバーレイまたは消去される。
 - 書き込み中のレコードに PROTECT キーワードが指定されている。

カーソル位置決めキーワードを有効にするには、キーボードをロック状態からロック解除状態にする必要があります。つまり、書き込み操作前にキーボードがロック解除される場合、カーソル位置決めキーワードは書き込み操作時に即時に有効になりません。ただし、例外が 1 つあります。キーボードが出力操作中に一時的にロックされる場合、出力操作の終了時にキーボードがロック解除される場合にカーソル位置決めキーワードが有効になります。

さらに、書き込み操作時に上記のいずれかの条件が発生する場合は、同じ操作または次の操作 (最後の書き込み操作) によってユーザー・アクション前にキーボードがロック解除される必要があります。

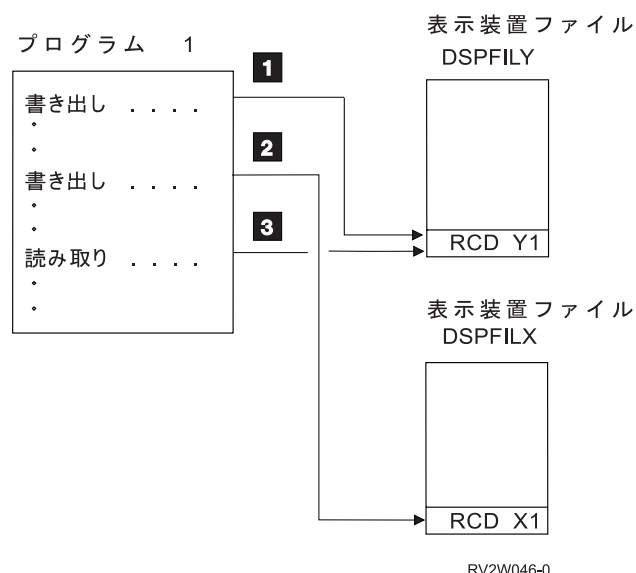
サブファイルに対する書き込み操作では、入力または出力がディスプレイ装置に送信されないため、キーボードはロック解除されません。

前に表示した情報の保管

別のディスプレイ・ファイルがすでにディスプレイ装置を使用している場合であっても、ディスプレイ・ファイルはそのディスプレイ装置に対して開くことができます。2 番目のディスプレイ・ファイルに対して入出力操作を実行すると、最初のディスプレイ・ファイルは中断状態になります。

ディスプレイ・ファイルが中断状態になると、ディスプレイ・ファイル作成 (CRTDSPF) またはディスプレイ・ファイル変更 (CHGDSPF) コマンドの RSTDSP パラメーターに *YES を指定する場合、ディスプレイ上の情報はシステムによって自動的に保管されます。そのディスプレイ・ファイルに対して後で入出力操作が実行されるときに、ディスプレイ・ファイルの内容を復元することが可能です。RSTDSP パラメーターに *NO を指定する場合、アプリケーション・プログラムはディスプレイを再書き込みして再表示する必要があります。

RSTDSP パラメーターを指定すると、以下の例に示されているとおり、プログラム呼び出し、キーボード入力、およびファイル入出力処理をオーバーラップできます。



- 1** プログラム 1 が、ディスプレイ・ファイル DSPFILY 内のレコード様式 RCD Y1 に対して書き込み操作を実行し、これによってディスプレイ・ファイル DSPFILY が活動化されます。
- 2** プログラム 1 が、ディスプレイ・ファイル DSPFILX 内のレコード様式 RCD X1 に対して書き込み操作を実行し、これによってディスプレイ・ファイル DSPFILY が中断状態にされ、ディスプレイ・ファイル DSPFILX が活動化されます。ディスプレイ・ファイル DSPFILY に RSTDSP(*YES) を指定する場合、ディスプレイ装置に表示されるデータは保存されます。代わりに RSTDSP(*NO) を指定する場合、データは失われ、プログラムは RCD Y1 に情報を書き込んで再表示する必要があります。
- 3** プログラム 1 が、レコード様式 RCD Y1 に対して読み取り操作を実行し、これによってディスプレイ・ファイル DSPFILX が中断状態にされ、ディスプレイ・ファイル DSPFILY が復元されます。DSPFILY に RSTDSP(*YES) パラメーターを指定する場合、DSPFILY が中断状態になるときにディスプレイ装置に表示されているデータを復元できます。

ディスプレイ・ファイルに RSTDSP(*YES) を指定する場合で、別のディスプレイ・ファイルに対する操作のためにそのディスプレイ・ファイルを中断状態にしたり復元したりすると、ディスプレイによっては画面上で短く明滅する場合があります。ディスプレイ・ファイルのレコードがディスプレイ上にあり、2 番目のディスプレイ・ファイルに対して入出力操作が行われる場合、最初のファイルは中断状態になり、画面の内容が保存されます。最初のディスプレイ・ファイルに戻ると、ディスプレイ・ファイルおよび画面の内容が復元されます。ディスプレイ・ファイル内の別のレコード様式に対して書き込み操作が行われる場合、復元されたディスプレイは出力操作が完了する前に短く明滅します。最初のファイルに戻る場合に、ディスプレイの内容をプログラムから完全に再書き込みする場合は、RSTDSP(*NO) を使用してください。

以下の状態では、RSTDSP パラメーターに *YES を指定する必要があります。

- 以下のキーワードが有効になっているレコードを書き込んでいる場合。
 - CLRL
 - OVERLAY
 - PUTOVR
 - PUTRETAIN
 - ERRMSG
 - ERRMSGID

ディスプレイ上のレコードは、これらのキーワードが適用されるレコードであることを確認する必要があります。ディスプレイ・ファイルが中断状態になる場合、これらのキーワードを使用するレコード様式に対する書き込み操作が有効になるよう、データを画面に復元する必要があります。

- 書き込み操作に干渉することなく、ディスプレイ上のレコード様式に対する複数の読み取り操作を実行する場合。読み取られたデータの処理中にプログラムを呼び出す必要がある場合で、そのプログラムが独自のディスプレイを提供する場合、プログラムで実行される後続の読み取り操作は、ディスプレイを正しく復元します。

データを保管してもう一度表示すると、システムおよびデータ伝送オーバーヘッドが大きくなります。1920 文字 5250 ディスプレイ装置の場合、ディスプレイ・データが保管され再表示されるたびに約 3000 文字が伝送されます。このオーバーヘッドを回避するには、以下を実行するようアプリケーション・プログラムを作成してください。

- ディスプレイ・ファイルに SHARE(*YES) を指定することによって、アプリケーション内のプログラムがディスプレイ・ファイルの同じコピーを共用するようにする。
- アプリケーション内のプログラムがディスプレイに書き込むたびに、完全なディスプレイ再書き込みを実行する。完全なディスプレイ再書き込みは、レコードが画面に書き込まれる場合、および OVERLAY キーワードが使用されていないか暗黙指定されていない場合に実行されます。

注: 完全なディスプレイ再書き込みが実行されず、新しい入力フィールド (現在表示されているフィールド上の画面上の位置を占有している) がディスプレイに送信される場合、プログラムはメッセージ (CPF5192) を受信します。このようになるのは、5250 ディスプレイ装置で、ディスプレイに送信される新しい入力フィールドが、現在ディスプレイ上にある入力フィールドより下の位置に表示される必要があるためです。通常の操作では、5250 要件を満たすためにデータ管理機能がフィールド処理を実行します。87 ページの『5250 ディスプレイ装置上のレコード様式の問題の回避』を参照してください。

SHARE(*YES) パラメーターが指定されたディスプレイ・ファイルを使用するプログラムが、同じディスプレイ・ファイルを使用する別のプログラムを呼び出すと、両方のプログラムがファイルを開いているとして

もディスプレイ・ファイルは中断状態になりません。ディスプレイ・ファイルが共用されていない場合、システムはプログラムごとに別々のディスプレイ・ファイルのコピーを保守し、ディスプレイ・ファイルを個別に中断状態にして復元します。

システム・プログラムはファイル共用を指定しないため、プログラムがディスプレイ・ファイルを含んでいて、ディスプレイを提供するシステム機能呼び出す場合は、CRTDSPF または CHGDSPF コマンドで RSTDSP(*YES) を指定する必要があります。ただし、アプリケーションの通常のパスを中断するシステム機能 (システム要求メニューまたは中断メッセージの表示など) は、RSTDSP(*YES) が指定されていない場合でもディスプレイを復元します。

中断状態のファイルに対してクローズ操作が発行された後で、保管されたディスプレイをもう一度表示するには、保管されたディスプレイ・データ内のレコード様式に KEEP キーワードを指定します。

コマンド・キーに対する入出力操作の影響についての理解

読み取り操作および書き込み操作は、機能キーの機能に影響を与える場合もあれば、影響を与えない場合もあります。

- ディスプレイに入力または出力が送信されない書き込みまたは書き込み・読み取り操作は、どのキーが有効であるかに影響を与えません。このような操作の例として、サブファイル・レコード様式に対する書き込み操作または更新操作があります。
- ERRMSG または ERRMSGID キーワードを選択することによって書き込みまたは書き込み・読み取り操作がメッセージを表示する場合、その出力で有効になっているコマンド・キーは有効です。したがって、エラーが発生する場合に、さまざまなコマンド・キーのセットが有効になるよう指定できます。
- 表示されているサブファイル・レコード様式が 1 つだけで、サブファイル制御レコード様式で SFLDROP キーワードに CAnn または CFnn キーが指定されている場合、そのキーはサブファイルがディスプレイ上にある限り有効のままです。さらに、SFLENTER キーワードに指定されるキーは、別の書き込みまたは書き込み・読み取り操作が実行されるまで有効のままです。次の出力操作時に、そのレコードの指定が適用されます。
- 2 つのサブファイル・レコード様式が表示されている場合で、その両方で SFLDROP キーワードが指定されている場合は、最後の SFLDROP キーワードだけが使用されます。同時に存在できるドロップ・キーは 1 つだけです。

5250 ディスプレイ装置上のレコード様式の問題の回避

5250 ディスプレイ装置の特性のため、特定のレコード様式位置決めおよび操作の組み合わせにより、望ましくない結果が生じる可能性があります。以下の例は、望ましくない結果をもたらす可能性のある組み合わせを示し、このような結果を回避するための方法を説明しています。

作成されるディスプレイとレコード様式の DDS:

Enter all information regarding the subscriber:

Last name: _____ First name: _____ MI: ____
Street: _____ Apt: _____
City: _____ State: ____ ZIP: _____

DETAIL
レコード
様式

CA1-Display state table, CA2-Display subscription table

RSLH166-0

Alabama AL Alaska AS Arkansas AK Arizona AK
California CA Delaware DE DST Columbia DC Florida DC
Y-More state names, N-No more state names

Enter all information regarding the subscriber:

Last name: Doe First name: John MI: E
Street: 112 Elm Apt: 3A
City: Anytown State: ____ ZIP: _____

STATES
レコード
様式

DETAIL
レコード
様式

CA1-Display state table, CA2-Display subscription table

RV2W047-0


```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A           R RCDA
A           FLD1           1 I 2 4DSPATR(PC)
A* STATES record format follows
A           R STATES                   OVERLAY
A           SNAME1        12 0 1 2
A           SCODE1         2 0 1 15
A           SNAME2        12 0 1 18
A           SCODE2         2 0 1 31
A           SNAME3        12 0 1 34
A           SCODE3         2 0 1 47
A           SNAME4        12 0 1 50
A           SCODE4         2 0 1 63
A           .             .
A           .             .
A           .             .
A           SNAME8        12 0 2 50
A           SCODE8         2 0 2 63
A                   3 2'Y-More state names, N-No more +
A                   state names'
A           MORESNAM      1 I 3 48VALUES('Y' 'N')
A* DETAIL record format follows
A           R DETAIL                   OVERLAY CA01(11) CA02(12)
A                   4 2'Enter all information regarding +
A                   the subscriber:'
A                   5 1'Last name:' DSPATR(HI)
A           NAMEL         20 I 5 12
A                   5 33'First name:' DSPATR(HI)
A           NAMEF         13 I 5 45
A                   5 59'MI:' DSPATR(HI)
A           MI            1 I 5 63
A                   6 1'Street:' DSPATR(HI)
A           STREET       45 I 6 9
A                   6 55'Apt:' DSPATR(HI)
A           APT           4 I 6 60
A                   7 1'City:' DSPATR(HI)
A           CITY         15 I 7 7
A                   7 23'State:' DSPATR(HI)
A           SCODE         2 I 7 30
A                   7 33'Zip:' DSPATR(HI)
A           ZIP1          5 I 7 38
A                   7 44'-'
A           ZIP2          4 I 7 46
A           .             .
A           .             .
A           .             .
A                   10 1'CA1-Display state table, +
A                   CA2-Display subscription table'

```

図 20. レコード様式の問題を示すサンプル DDS

DETAIL レコード様式が画面上にあり、ユーザーがサブスクライバーのデータを入力するとします。ユーザーは入力する状態の状態コードが分からないため、ユーザーは CA01 キーを押します。

CA01 は CAnn キーとして定義されているため、CA01 キーが押されてもデータはシステムに伝送されません。ただし、データは画面に残ります。応答標識 11 がオンに設定されているため、プログラムは CAnn キーが押されたことを検出します。次に、プログラムは STATES レコード様式を表示します。

STATES レコード様式は、物理的に DETAIL レコード様式の上にあるため、システムは DETAIL レコード様式内の入力フィールドのフィールド様式を再送する必要があります。(STATES レコード様式に含まれ

ていたのが出力専用フィールドだけであり、RCDA レコード様式が置換されていた場合にも、システムはフィールド様式を再送します。この場合、特定のカーソル位置を指定するレコード様式 (RCDA) は除去されるため、フィールド様式が再送されます。)

システムが DETAIL レコード様式のフィールド様式を再送するため、以下の問題が発生します。

- DETAIL レコード様式内のすべての入力フィールドの変更データ・タグ (MDT) が失われます。プログラムが DETAIL レコード様式に対して次の読み取りを実行すると (例えば、ユーザーが Enter キーを押す場合)、ユーザーが CA01 キーを押す前に入力されていたフィールドはプログラムに戻されません。データが画面に残っているとしても、プログラムは入力されたそのデータを取得できません。

この問題を回避するには、以下のようにします。

- CAnn キーを使用しないようにします。
- CAnn キーを使用する必要がある場合、CAnn キーを含む様式を画面に書き込まないようにし、入力可能フィールドが両方の様式に含まれている場合は、最初の様式の上に物理的に配置される別の様式を書き込むようにします。
- カーソル位置指定が含まれている様式を除去する様式を画面に書き込まないようにします。
- 定数フィールドの強調表示属性 (姓は例外) が失われます。システムは、出力専用フィールドのフィールド様式を再送しません。ただし、出力フィールドの先頭属性文字が入力可能フィールドの終了属性文字と同じ位置になるよう、出力専用フィールドが入力可能フィールドの直後にある場合は、出力フィールドの属性は通常に戻ります。

この問題を回避するには、以下のようにします。

- 特別な表示属性を指定する出力専用フィールドを入力可能フィールドの直後に指定しないでください。
- 特別な表示属性を指定する出力専用フィールドを入力可能フィールドの直後に指定する必要がある場合、その様式を画面に書き込まないようにし、入力可能フィールドが両方の様式に含まれている場合は、最初の様式の上に物理的に配置される別の様式を書き込むようにします。
- カーソル位置指定が含まれている様式を除去する様式を画面に書き込まないようにします。

注: システムは、サブファイルが全ページから部分ページ、部分ページから全ページ、または部分ページから部分ページにロールされるときに、入力可能フィールドの属性を再送する必要があります。フィールド属性の再送時に、上記の 2 つの問題も生じる可能性があります。

獲得されたディスプレイ装置の入出力操作からの解放

解放操作により、ディスプレイ装置は、ファイルを介したさらなる入出力操作に対して不適格になります。この操作は、複数のディスプレイ・ファイル・アプリケーションで使用されます。あるいは、プログラムでエラー回復を実行している場合に使用されます。解放中のディスプレイ装置が送信勧誘される場合、送信勧誘は終了します。ディスプレイ装置に使用可能なデータがあっても、そのデータは失われます。解放操作は、ファイルに対して現在獲得されているディスプレイ装置に対してのみ実行できます。

解放操作は、獲得、入出力、および解放操作のエラーから回復する場合にも使用できます。ディスプレイ装置を解放した後は、別の獲得操作でもう一度獲得しなければ、そのディスプレイ装置に対して入出力操作を実行することはできません。ディスプレイ装置を解放してからもう一度ディスプレイ装置を獲得することによってエラーから回復するようプログラムが作成されている場合は、WAITFILE キーワードに *IMMED 以外の値を指定する必要があります。これは、システムがディスプレイ装置の割り振り記述をジョブからサブシステム、サブシステムからジョブに転送するのに短い時間がかかるためです。

ディスプレイ・ファイルのクローズ

クローズ操作によって、ディスプレイ・ファイルは、プログラムとシステムとのさらなる入出力操作に対して不適格になります。クローズ操作を開始する方法については、該当する高水準言語の資料を参照してください。

ディスプレイ・ファイルが共有されていない場合、クローズ操作は、ファイルに対して獲得されたすべてのディスプレイ装置を暗黙的に解放し、オープン操作または獲得操作で割り振られたファイル・リソースを割り振り解除します。

クローズ操作が成功すると、ファイルに対して有効な操作はオープン操作だけになります。クローズ操作が失敗すると、プログラムは 2 回目のクローズ操作を実行します。

表示操作の高水準言語へのマッピング

オペレーティング・システムでサポートされる入出力要求および同等の高水準言語操作を以下に示します。

表 12. オペレーティング・システムでサポートされるディスプレイ・ファイル操作および同等の高水準言語コマンド

操作	BASIC ステートメント	ILE C/C++ 関数	CL コマンド	ILE COBOL ステートメント	ILE RPG 操作
オープン 獲得 解放 属性取得 書き込み	OPEN WRITE	fopen、 _Ropen _Racquire _Rrelease _Rdevatr	 SNDF	OPEN ACQUIRE DROP ACCEPT WRITE	OPEN ACQ REL POST WRITE、 出力仕様
読み取り(待機)	READ	fread、 _Rformat、 _Rpgmdev、 _Rwrite	RCVF WAIT(*YES)	READ	1 次または 2 次 ファイル入力、 READ
読み取り	READ		RCVF WAIT (*NO)	READ	
読み取り取り消し 待機 送信勧誘		fwrite ¹ 、 _Rformat、 _Rpgmdev、 _Rwrite ¹	ENDRCV WAIT SNDF ¹	WRITE ¹	WRITE ¹
送信勧誘された装 置からの読み取り 送信勧誘取り消し		_Rreadindv		READ	READ
書き込み・読み取 り(待機)		fwrite、 _Rformat、 _Rpgmdev、 _Rwrite _Rwriterd、 _Rformat、 _Rpgmdev	SNDF	WRITE	WRITE
書き込み・読み取 り(待機なし)			SNDF WAIT(*YES)		EXFMT
			SNDF WAIT(*NO)		

表 12. オペレーティング・システムでサポートされるディスプレイ・ファイル操作および同等の高水準言語コマンド (続き)

操作	BASIC ステートメント		ILE C/C++ 関数	CL コマンド	ILE COBOL ステートメント		ILE RPG 操作
クローズ	CLOSE、END	fclose、_Rclose		RETURN、 RCLRSC	CLOSE、 CANCEL、 STOP RUN		CLOSE、 RETRN

1. これは、INVITE DDS キーワードが選択されたレコード様式の書き込み操作です。

ディスプレイ・ファイルに対する入出力操作中にエラーが発生する場合、エラーを診断し、必要なエラー回復アクションを判別するために、ファイル従属入出力フィードバック域内のメジャー/マイナー戻りコード・フィールドを使用することができます。

同一ジョブ内でのディスプレイ・ファイルの共用

CRTDSPF、CHGDSPF、および OVRDSPF コマンドに SHARE パラメーターを指定することにより、複数のプログラムが同じデータ・パスまたはディスプレイ装置を共用するよう指定できます。SHARE パラメーターを使用すると、複数のプログラムが、ファイル状況、位置、およびストレージ域を共用できるようになり、ジョブが必要とする主ストレージの量を減らすことにより、またファイルのオープン/クローズにかかる時間を短縮することによってパフォーマンスを改善することができます。

SHARE(*YES) パラメーターを使用すると、同一のジョブを実行している 2 つ以上のプログラム間でオープン・データ・パス (ODP) を共用できます。**オープン・データ・パス**とは、ファイルのすべての入出力操作が実行されるパスのことです。これは、プログラムをファイルに接続します。このように指定していない場合は、ファイルを開くたびに新しいオープン・データ・パスが作成されます。ファイルが複数回開かれる場合で、同一ジョブ内でオープン・データ・パスがファイルに対してまだアクティブの場合、現在のファイル・オープンでファイルのアクティブ ODP が使用可能であること、また新しいオープン・データ・パスを作成する必要がないことを指定できます。これによって、最初のオープン後にファイルを開くのに必要な時間が短縮され、ジョブで必要な主ストレージの量が減ります。オープン・データ・パスを共用するには、最初のオープン、および同じファイルのその後のオープンに SHARE(*YES) を指定する必要があります。他のファイルに SHARE(*YES) を指定するかどうかは、アプリケーションによって異なります。

注: ほとんどの高水準言語プログラムは、ファイルが共用されているかどうかに関係なくオープンまたはクローズ操作を処理します。ファイルの共用は、高水準言語プログラムでは指定しません。SHARE パラメーターを介して、同一ジョブ内でファイルを共用することを指示します。SHARE パラメーターは、ファイルの作成、変更、およびオーバーライドの各コマンドでのみ指定できます。詳しくは、該当する言語の資料を参照してください。

ジョブ内で共用されるファイルのオープン操作についての理解

SHARE(*YES) を指定することによって同一ジョブ内で共用されるファイルを開く場合は、以下の項目について考慮してください。

- ジョブ内で共用ファイルを初めて開くときに、ファイルの後続のオープンに必要なすべてのオープン・オプションが指定されていることを確認する必要があります。共用ファイルの後続のオープンに指定するオープン・オプションが、共用ファイルを最初に開いたときに指定されていたオプションと一致しない場合、エラー・メッセージがプログラムに送信されます。(これは、非互換オプションを除去するようプログラムを変更することによって訂正できます。)

例えば、ジョブ内の FILE1 を開く最初のプログラムが PGMA で、PGMA で必要なのはファイルの読み取りだけであるとします。ただし、PGMA は、同じ共用ファイルのレコードを削除する PGMB を呼び出します。PGMB が共用ファイルのレコードを削除するため、PGMA は、PGMA もレコードを削除するようにファイルを開く必要があります。これは、高水準言語で正しい指定を使用することによって行います。(一部の高水準言語でこれを行うには、決して実行されないファイル操作ステートメントを使用しなければならない場合があります。)詳細については、該当する言語の資料を参照してください。

- プログラムまたはオーバーライド・コマンドでライブラリー名を指定しなかった場合 (*LIBL が使用される)、システムは、*LIBL が指定された同一の共用ファイルが最後に開かれたときからライブラリー・リストは変更されていないと見なします。ライブラリー・リストが変更されている場合、正しいファイルが開かれるようオーバーライド・コマンドにライブラリー名を指定する必要があります。
- 共用ファイルを最初に開いたときに指定されていたオーバーライドおよびプログラム仕様が処理されます。後続のオプションで指定されるオーバーライドおよびプログラム仕様で、オーバーライド・コマンドの SHARE または LVLCHK で指定されるファイル名または値を変更するオーバーライドおよびプログラム仕様以外のものは無視されます。

ジョブ内で共用されるファイルの入出力操作についての理解

システムは、ファイルを共用するすべてのプログラムに同じ入出力域を使用し、どのプログラムが操作を実行するかにかかわらず、操作が順次に行われるようにします。例えば、プログラム A がファイルからレコードを順次に読み取っている場合で、プログラム B を呼び出す直前にレコード 1 を読み取り、プログラム B もファイルを順次に読み取る場合、プログラム B は最初の読み取り操作でレコード 2 を読み取ります。その後プログラム B が終了し、プログラム A が次のレコードを読み取る場合、プログラム A はレコード 3 を受け取ります。ファイルが共用されていなかった場合、プログラム A はレコード 1 およびレコード 2 を読み取り、プログラム B はレコード 1 を読み取ります。

ディスプレイ・ファイルの場合、ディスプレイ装置は最後の入出力操作のときと同じ状態のままです。

表示ファイルおよび ICF ファイルの場合、ファイルを開く最初のプログラム以外のプログラムは、さらに多くのディスプレイまたはプログラム・ディスプレイ装置を獲得するか、オープン・データ・パスに対してすでに獲得されているディスプレイまたはプログラム・ディスプレイ装置を解放する場合があります。ファイルを共用するすべてのプログラムは、新しく獲得されたディスプレイ装置にアクセスできますが、解放されたディスプレイ装置にはアクセスできません。

ジョブ内で共用されるファイルのクローズ操作についての理解

プログラムが共用ファイルを閉じるときに行われる処理は、オープン・データ・パスを現在共用している他のプログラムがあるかどうかによって異なります。他のプログラムがある場合、実行される主な機能は、クローズを要求しているプログラムをファイルから切り離すことです。ファイルを共用する他のすべてのプログラムは、ODP に接続されたままであるため、入出力操作を実行できます。

ファイルを終了するプログラムが、ファイルを共用する最後のプログラムの場合、クローズ操作は、共用オプションを指定してファイルを開かなかった場合に実行するすべての機能を実行します。これには、ファイルに割り当てられたリソースの解放およびオープン・データ・パスの破棄が含まれます。

この最後のクローズ操作で提供される機能は、特定の実行時エラーから回復するのに必要な機能です。アプリケーションがこのようなエラーから回復するよう作成されていて、アプリケーションが共用ファイルを使用する場合、エラー発生時にファイルに接続されているすべてのプログラムはファイルを閉じる必要があります。つまり、プログラム・スタック内の前のプログラムに戻って、これらの各プログラムでファイルを閉じなければならない場合があります。

第 4 章 サブファイルを使用したレコードのグループの表示

サブファイルは同じレコード様式を持つレコードのグループで、ディスプレイ装置を対象として 1 つの操作で読み取りおよび書き込みが行われます。以下のサンプル画面は、サブファイルの例を示しています。

CUSTOMER NAME SEARCH			
Search code: 41401			
NUMBER	NAME	ADDRESS	CITY STATE
41401	Adam's Home Repair	121 Golden Circle	Chicago IL
41402	Jane's Radio/TV	135 Ransom Drive	St Paul MN
41403	Advanced Electronics	809 8th Street	St Paul MN
41404	Riteway Repair	443 Western Lane	New York NY
41405	Fixtures, Inc.	607 9th Avenue	Chicago IL
41406	Hall's Electric	200 Main Street	St Paul MN

プロンプト・レコード様式

サブファイル

RV2W049-0

DDS キーワードについて

この章では、DDS キーワードを使用してサブファイルを記述します。特定の DDS キーワードについて詳しくは、iSeries Information Center の『DDS』のトピックを参照してください。

サブファイルの使用法についての理解

サブファイルは、類似している複数のレコードを表示しなければならない場合に便利です。表示される複数のレコードが 1 つのディスプレイに収まるようにサブファイルを記述できます。そうしない場合、ディスプレイに表示できる行数を超えてしまいます。

以下の目的でサブファイルを使用できます。

- 表示専用。この場合、ユーザーはディスプレイ上でサブファイル・レコード (例えば、特定のオーダー番号に関するすべての行項目、または直前のサンプル画面に示したような顧客名と住所を含むレコードのグループ) を検討できます。
- 選択項目とともに表示。この場合、ユーザーは画面上の項目の 1 つに関する詳細情報を要求できます。以下の例の 1 つ目のサンプル画面では、ユーザーはレコード番号フィールドにレコード番号を入力して、特定の顧客に関するレコードを要求できます。2 つ目のサンプル画面では、ユーザーは選択番号フィールドに X を入力して、特定の顧客に関するレコードを要求できます。

```

Enter customer number: 41401
Enter record number: _____
RECORD NUMBER NAME ADDRESS CITY STATE
01 41401 Adam's Home Repair 121 Golden Circle Chicago IL
02 41402 Jane's Radio/TV 135 Ransom Drive St Paul MN
03 41403 Advanced Electronics 809 8th Street St Paul MN
04 41404 Riteway Repair 443 Western Lane New York NY
05 41405 Fixtures, Inc. 607 9th Avenue Chicago IL
06 41406 Hall's Electric 200 Main Street St Paul MN

```

```

Enter customer number: 41401
SELECT
RECORD NUMBER NAME ADDRESS CITY STATE
- 41401 Adam's Home Repair 121 Golden Circle Chicago IL
- 41402 Jane's Radio/TV 135 Ransom Drive St Paul MN
- 41403 Advanced Electronics 809 8th Street St Paul MN
- 41404 Riteway Repair 443 Western Lane New York NY
- 41405 Fixtures, Inc. 607 9th Avenue Chicago IL
- 41406 Hall's Electronic 200 Main Street St Paul MN

```

- 情報の変更。この場合、ユーザーはサブファイル中の 1 つ以上のレコードに変更を加えることができます。以下のサンプル画面では、ユーザーは「QTY」と「SHIP」の値を変更できます。

```

UPDATE SHIP QUANTITY ON ORDERS
Order: 11589 Customer number: 11111 Customer name: Al'Supply
ITEM DESCRIPTION QTY SHIP LOCATION
25764 Pliers 10 10 RST
33624 Hammer 500 250 RST
49821 Pliers 200 200 RST
26837 Wire Cutters 50 25 RST

```

- 入力専用、妥当性検査なし。この場合、ユーザーは可能な限り高速にデータを入力できます。あるいは、入力専用、妥当性検査あり。この場合、ユーザーはシステムまたはプログラムによって有効な項目に関する妥当性検査が行われたデータを入力できます。以下のサンプル画面は、入力専用のサブファイルを示しています。

```

Enter order number: XXXXX
ITEM NUMBER QUANTITY
XXXXX XXX
XXXXX XXX
XXXXX XXX
XXXXX XXX
XXXXX XXX
XXXXX XXX
XXXXX XXX
_____ _____

```

- タスクの組み合わせ。この場合、ユーザーは、新しいレコードを入力とともにデータの変更を行うことなどができます。以下の例では、ユーザーは既存の名前と住所を変更するか、新しいレコードを入力できます。

```

CUSTOMER NAME SEARCH
Search code: 41401
NUMBER NAME ADDRESS CITY STATE
41401 Adam's Home Repair 121 Golden Circle Chicago IL
41402 Jane's Radio/TV 135 Ransom Drive St Paul MN
41403 Advanced Electronics 809 8th Street St Paul MN
_____
_____
_____

```

- 単一選択項目および複数項目選択リストの表示。単一選択項目リストは、基本的にスクロール可能なリストで、ユーザーはそこから 1 つの項目を選択することができます。複数項目選択リストは、基本的に

スクロール可能なりストで、ユーザーはそこから 1 つ以上の項目を選択することができます。選択リストについて詳しくは、173 ページの『選択リストの概要』を参照してください。



RV3W077-0

DDS ソースでのサブファイルの記述

DDS ソースで記述するサブファイルごとに、サブファイル・レコード様式とサブファイル制御レコード様式の 2 つのタイプのレコード様式が必要です。

- **サブファイル・レコード様式**は、サブファイルの 1 行中のフィールドを定義します。

高水準言語プログラムはサブファイル・レコード様式を使用して、サブファイルを読み取り、新しいレコードをサブファイルに書き込み、サブファイルを更新します。サブファイル・レコード様式に対する操作はサブファイルと高水準言語プログラムとの間で実行されます。サブファイル・レコード様式に対する操作が行われてもディスプレイの変更はありません。

サブファイル・レコード様式にはサブファイル (SFL) キーワードが必要です。

- **サブファイル制御レコード様式**には、ヘッディング情報が含まれており、サイズ変更、初期設定、消去などのサブファイル機能を制御します。

高水準言語プログラムはサブファイル制御レコード様式に関する操作を実行して、サブファイルのディスプレイへの書き込みや、サブファイルのディスプレイからの読み取りを行います。

サブファイル制御レコード様式には以下の DDS キーワードが必要です。

- サブファイル制御 (SFLCTL) キーワード。直前のサブファイル・レコード様式のサブファイル制御レコード様式を識別します。
- サブファイル・サイズ (SFLSIZ) キーワード。このキーワードは、サブファイルのサイズを指定します。
- サブファイル・ページ (SFLPAG) キーワード。このキーワードは、サブファイル・ページのサイズを指定します。

- サブファイル表示 (SFLDSP) キーワード。このキーワードは、サブファイル中のレコードの表示を開始する時点を指定します。

サブファイル・レコード様式の DDS は、サブファイル制御レコード様式の DDS より前になければなりません。

個々のサブファイルには、以下の 2 つのタイプのレコードがあります。

- **活動サブファイル・レコード**は、以下のようなレコードです。
 - 書き込み操作によりサブファイルに追加された。
 - サブファイル初期設定 (SFLINZ) キーワードにより、活動状態として初期設定された。
 - 有効な次に変更するサブファイル (SFLNXTCHG) キーワードを使用した書き込み操作または更新操作がレコードに対して発行された際に、変更が加えられた。
 - ユーザーによって変更が加えられた。
- **非活動サブファイル・レコード**は、以下のようなレコードです。
 - 書き込み操作によりサブファイルに追加されなかった。
 - SFLINZ キーワードとサブファイル・レコード非活動 (SFLRNA) キーワードによって非活動状態として初期設定された。

サブファイルに対して以下の機能を実行することもできます。

表 13. サブファイルに関するオプション機能

機能	DDS キーワード	追加情報
プログラム・メッセージ待ち行列からのメッセージをサブファイルに含めることができる	サブファイル・メッセージ・キー (SFLMSGKEY)、サブファイル・メッセージ・レコード (SFLMSGRCR)、およびサブファイル・プログラム・メッセージ待ち行列 (SFLPGMQ)	サブファイル・メッセージ・キーワードについて詳しくは、iSeries Information Center の『DDS』のトピックを参照してください。
新しいレコードを書き込む前にサブファイルのすべてのレコードを消去する	サブファイル消去 (SFLCLR)	ただし、サブファイル制御レコードに対して SFLDSP キーワードが有効であれば、サブファイルはディスプレイから消去されません。レコードのないサブファイルに SFLCLR キーワードを指定すると無視されます。
サブファイル制御レコードを表示する時点を制御する	サブファイル表示制御 (SFLDSPCTL) または SFLDSP	ディスプレイの内容の変更を行うキーワードは、SFLDSP キーワードと SFLDSPCTL キーワードのみです。フィールドが表示されていないにもかかわらず、入力操作を行って CFnn または CAnn キーの状況を検索する場合は、SFLDSPCTL キーワードが有効でなければなりません。

表 13. サブファイルに関するオプション機能 (続き)

機能	DDS キーワード	追加情報
サブファイルを削除して、別のサブファイルを使用できるようにするか、サブファイルを使用せずにディスプレイ・ファイルの処理を続行する	サブファイル削除 (SFLDLT)	通常は、プログラムによってサブファイルを削除しないでください。サブファイルを含むファイルを閉じると、サブファイルはシステムによって自動的に削除されます。しかし、別のプログラムによってファイルが共有され、依然として開かれている場合は、サブファイルは削除されないため、ご使用のプログラムで削除しなければなりません。すでに最大数のサブファイルを使用しており、別のサブファイルを使用する必要がある場合のみ、サブファイルを削除する必要があります。サブファイルがない場合は、SFLDLT キーワードは無視されます。
レコード番号別にサブファイルのページを表示する	サブファイル・レコード番号 (SFLRCDNBR)	SFLRCDNBR キーワードに CURSOR が指定されている場合は、このフィールドの内容によって相対レコード番号が識別されるサブファイル・レコード中にカーソルが置かれます。カーソルはサブファイル・レコード中の最初の入力可能フィールドに置かれます。入力可能フィールドがない場合は、カーソルは最初の出力専用フィールドか定数フィールドに置かれます。
ディスプレイに収まらないほど多くのレコードがある場合に、サブファイル表示域 (ページ) の右下隅に正符号 (+) を表示する	サブファイル終了 (SFLEND) または (SFLEND(*PLUS))	最終レコードが表示されると、正符号はブランクに置き換えられます。SFLEND または SFLEND(*PLUS) キーワードと共にオプション標識を指定しなければなりません。
ディスプレイに収まらないほど多くのレコードがある場合に、サブファイル表示域 (ページ) の下の行に「続く...」という語を表示する	サブファイル終了 SFLEND(*MORE)	最終レコードが表示されると、「続く...」という語は「終わり」という語に置き換えられます。SFLEND(*MORE) キーワードと共にオプション標識を指定しなければなりません。
サブファイルの隣にスクロール・バーを表示する	サブファイル終了 SFLEND(*SCRBAR)	詳細については、173 ページの『選択リストの概要』を参照してください。
現行レコード・ページの先頭のレコードの相対レコード番号に戻る	サブファイル・スクロール (SFLSCROLL)	詳細については、173 ページの『選択リストの概要』を参照してください。

表 13. サブファイルに関するオプション機能 (続き)

機能	DDS キーワード	追加情報
サブファイル中のレコードを継続するか切り捨てるコマンド・キーを使用可能にする	サブファイル継続 (SFLFOLD) またはサブファイル切り捨て (SFLDROP)	<p>SFLFOLD キーワードを指定すると、レコードの初期画面が継続されます。SFLDROP キーワードを指定すると、レコードの初期画面が自動的に切り捨てられます。それからユーザーはコマンド・キーを押して、サブファイル・レコードの切り捨てられたバージョンまたは継続されたバージョンをそれぞれ表示できます。ページ・サイズがサブファイル・サイズと等しいか、サブファイルが 1 表示行に収まる場合は、指定したキーワード (SFLFOLD または SFLDROP) は無視されます。</p> <p>SFLFOLD と SFLDROP を両方とも同じサブファイルに対して使用できません。これらのキーワードにオプション標識を使用できます。オプション標識を使用して、サブファイルが最初に表示されるモードを判別します。両方のキーワードのオプションをオンまたはオフにすると、サブファイルは最初に継続モードで表示されます。キーワードのオプションをオフにしても、引き続きコマンド・キーを使用して切り捨てられたバージョンや継続されたバージョンを表示できます。</p>
Enter キーを次ページ・キーとして使用可能にして、コマンド・キーを使用して高水準言語プログラムに戻るようになる	サブファイル実行キー (SFLENTER)	<p>SFLENTER を使用した複数のサブファイルが同時に表示される場合は、Enter キーとして有効な CAnn または CFnn キーは、最近表示されたサブファイル上の SFLENTER に指定した CAnn または CFnn キーのみです。Enter キーを押した時点のカーソル位置により、影響を受けるサブファイルが決まります。</p>

表 13. サブファイルに関するオプション機能 (続き)

機能	DDS キーワード	追加情報
活動状態のレコードのないサブファイル、そのサブファイルが活動状態であっても初期設定する	SFLINZ およびサブファイル・レコード非活動 (SFLRNA)	<p>以下のいずれかが生じると、レコードは活動状態になります。</p> <ul style="list-style-type: none"> 特定のレコードに関する出力操作がサブファイルに対して発行される。SFLNXTCHG キーワードを使用しないと、レコードは変更済みと見なされません。 ユーザーが表示されたレコードにデータを入力する。レコードは活動状態で変更済みと見なされます。 <p>SFLDSP キーワードが有効な場合は、レコードは表示されます。レコード中のフィールドにデフォルト値が指定された場合は、それらの値はディスプレイに組み込まれます。</p>
ディスプレイ・ファイル中のサブファイル・レコード様式のフィールド記述によってすべてのレコードを初期設定する	SFLINZ	<p>サブファイル制御レコードに対する出力操作 (SFLCTL) 時に SFLINZ キーワードが有効な場合、システムはサブファイル・レコードに対するオプション標識はすべてオフであると想定します。したがって、N が前に付いているオプション標識のみ有効になります。出力操作に対する SFLDSP が有効な場合は、サブファイル・レコードが表示されます。出力操作に対する SFLINZ キーワードが有効な場合は、デフォルト値が指定されていない入力可能フィールドの内容は以下のように処理されます。</p> <ul style="list-style-type: none"> 数字フィールドはゼロに初期設定される。 文字フィールドはブランクに初期設定される。 浮動小数点フィールドはヌルに初期設定される。
次変更読み取り操作の実行時にレコードをプログラムに戻す	次に変更するサブファイル (SFLNXTCHG)	ユーザーがレコードに変更を加えなかった場合でも、レコードが戻されます。
ページの代わりに指定されたレコードの数によってロールする	サブファイル・ロール値 (SFLROLVAL)	このフィールドには、小数点以下の桁数がゼロの符号付き数値のキーボード・シフト属性がなければなりません。この長さは最大 4 桁まで可能で、出力/入力または入力専用フィールドとして定義しなければなりません。

表 13. サブファイルに関するオプション機能 (続き)

機能	DDS キーワード	追加情報
複数のレコードが 1 行に表示される場合に、1 行の個々のレコード間のスペース数を指定する	サブファイル行 (SFLLIN)	このキーワードは、サブファイルが横方向に表示される場合に使用されません。ディスプレイ・ファイルが複数の画面サイズをサポートしており、SFLLIN キーワードがデフォルトの (1 次) 画面サイズに加えて 2 次画面サイズにも適用される場合、画面サイズの条件名を指定しなければなりません。
ご使用のプログラムがサブファイル制御レコードに対して出力操作を行う際に、ディスプレイ上のメッセージ行にメッセージを書き込む	サブファイル・メッセージ (SFLMSG) およびサブファイル・メッセージ識別コード (SFLMSGID)	メッセージ・キーワードについて詳しくは、iSeries Information Center の『DDS』のトピックを参照してください。
サブファイル中のカーソルがある場所を判別する	サブファイル・カーソル相対レコード番号 (SFLCSRRRN)	カーソルのある相対レコード番号が、キーワード上でパラメーターとして指定されている隠しフィールドに戻されます。
サブファイルのカーソル移動先を指定する	サブファイル・カーソル移動先 (SFLCSRPRG)	SFLCSRPRG キーワードにより、カーソルが、あるサブファイル・レコード中のフィールドから、次に表示されるサブファイル・レコード中の同じフィールドに移動します。詳細については、33 ページの『入力フィールドのカーソル移動先の定義』を参照してください。
サブファイルが継続モードまたは切り捨てモードのどちらであったかを判別する	サブファイル・モード (SFLMODE)	これは必須パラメーターで、サブファイル制御レコードの場合のみ有効であり、SFLCTL キーワードを指定しなければなりません。
単一選択項目リストを定義する	サブファイル単一選択項目選択リスト (SFLSNGCHC)	詳細については、173 ページの『選択リストの概要』を参照してください。
複数項目選択リストを定義する	サブファイル複数選択項目選択リスト (SFLMLTCHC)	詳細については、173 ページの『選択リストの概要』を参照してください。
選択リスト内の選択項目の可用性を制御する	サブファイル選択制御 (SFLCHCCTL)	詳細については、173 ページの『選択リストの概要』を参照してください。
次変更読み取り操作を使用して、選択リスト内の選択された項目をすべて戻す	サブファイル戻り選択済み項目 (SFLRTNSEL)	詳細については、173 ページの『選択リストの概要』を参照してください。

DDS キーワードはどの順序でも指定できます。しかし、サブファイル・レコード様式 (SFL) がサブファイル制御レコード様式 (SFLCTL) の前でなければなりません。

多くの DDS サブファイル・キーワードは、オプション標識を使用して条件付けを行えます。

1 つのディスプレイ・ファイルに存在できるレコード様式の最大数は 1024 なので、1 つのディスプレイ・ファイルに最大 512 のサブファイルを指定できます。12 以下のサブファイルを同一のディスプレイ装置に対して同時に活動状態にすることができます。1 つ以上の活動状態のサブファイルをディスプレイ装置で同時に表示できます。サブファイルには表示できるフィールドが 1 つ以上含まれていなければならない、サブ

ファイル・レコード様式とサブファイル制御レコード様式がオーバーラップすることはできません。これらのレコードがオーバーラップすると、ディスプレイ・ファイルを作成できません。

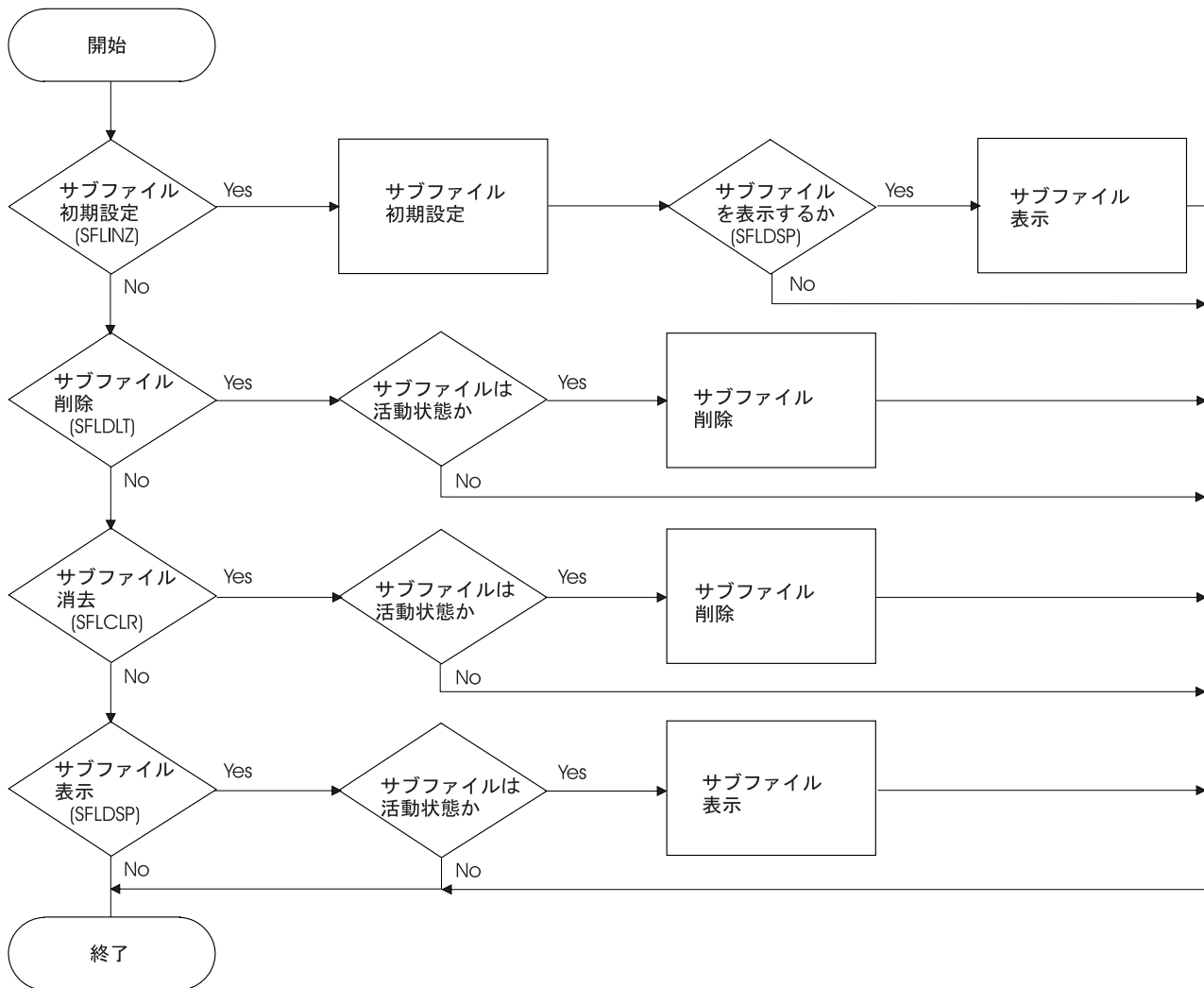
サブファイル・レコード中で名前指定されたすべてのフィールドは、入力可能でないフィールドを含めて、プログラムに戻されます。

サブファイル・レコードに関する入力データ妥当性検査が指定されている場合は、ロール機能が実行される前に妥当性検査が実行されます。データの妥当性検査が失敗すると、ロール機能は実行されません。

サブファイルに書き込まれたレコードの相対レコード番号がサブファイル・サイズと等しい場合は、システムはプログラムに CPF5003 メッセージを送信し、サブファイルがいっぱいであることを示します。(すべてのレコードが活動状態になっている必要はありません。つまり、サブファイルに書き込まれたレコードが、サブファイル中の最終レコードのみであった場合でも、このメッセージは送信されます。) サブファイル・サイズがページ・サイズと等しくない場合に、プログラムがさらにレコードをサブファイルに書き込むと、それらのレコードが追加されるたびにシステムはサブファイルを自動的に拡張します。サブファイルが拡張されたことは、プログラムには通知されません。(レコード数が 9999 を超えるとサブファイルを拡張できなくなります。) また、サブファイル・サイズがページ・サイズと等しい場合でも、サブファイルに最後に書き込まれたレコードがサブファイル表示域の使用可能な最後の行を占めなければ、サブファイルがいっぱいであることはプログラムに通知されません。

拡張されたスペースはサブファイルと接続されないため、拡張されたサブファイルの処理効率は低下します。サブファイルのサイズの指定値を大きくして拡張されないようにすることもできますが、拡張されたスペースが非常にまれにしか使用されない場合や全く使用されない場合は、スペースをむだにすることになります。

104 ページの図 21 は、サブファイルの制御に使用される DDS キーワードの一部が実行時に処理される順序を示しています。



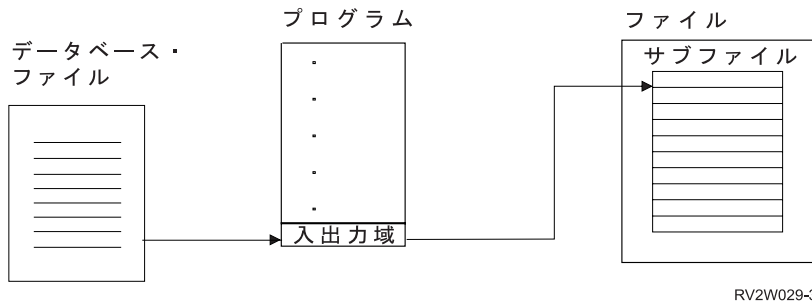
RSLH181-0

図 21. サブファイル制御用の DDS キーワードを処理する順序

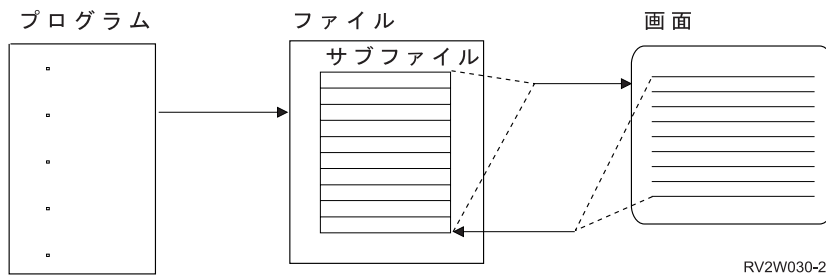
プログラムでのサブファイルの使用

サブファイルを使用するには、高水準言語プログラムで以下の基本操作を実行します。

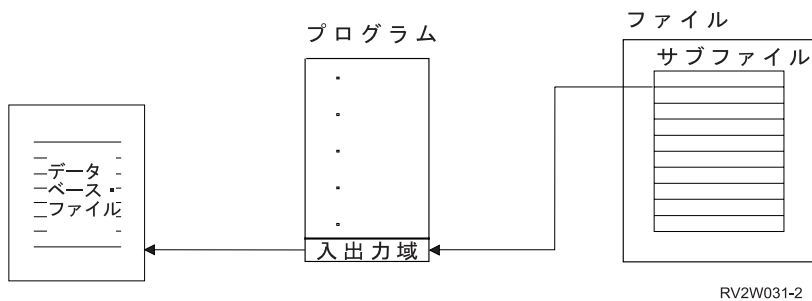
1. サブファイルを初期設定します。サブファイルを初期設定する方法の 1 つとして、データベース・ファイルからレコードを読み取ってサブファイルに書き込むという方法があります。サブファイルがいっぱいになるか、レコードがなくなるまで、一度に 1 つずつレコードをサブファイルに挿入します。



2. サブファイル制御レコード様式を使用して、1 回の出力操作でディスプレイにサブファイルを送信します。
3. ユーザーがレコードを検討するか、レコードに変更を加えるか、または新しいレコードを入力した (サブファイルの機能に応じて異なる) 後で、サブファイル制御レコード様式を読み取ります。



4. サブファイル中の個々のレコードを個別に処理します。必要に応じてデータベース・ファイルを更新したり新しいレコードをデータベース・ファイルに書き込んだりします。サブファイルの機能がレコードの更新の場合、プログラムで行う必要があるのは ILE RPG の READC 演算命令か ILE COBOL 動詞 Read Subfile Next Modified を使用した変更済みのレコードの処理だけです。



サブファイルを使用するディスプレイ・ファイルは、一度に 1 つのサブファイルの 1 つの部分のみ表示できます。表示される部分のことを、**サブファイル・ページ**と呼びます。サブファイル・ディスプレイの入力専用フィールドに入力したデータは、ファンクション・キー (ロール・キーなど) を押すと、サブファイルに入れられます。その後、このフィールドにサブファイル中の値が表示され、Enter キーを押すと起こることはアプリケーション・コードによって異なります。

注: ILE RPG の READC 演算命令の場合、データはサブファイルからプログラムに移動します。データはサブファイルから除去されず、入力専用フィールドがこの値に初期設定されていたかのように引き続きこのフィールドに表示されます。表示されないと、実際にデータがあってもまるでサブファイルが空であるかのようになってしまいます。

サブファイルが処理される (例えば、RPG の UPDAT 演算命令により) と、サブファイルからデータが除去され、入力専用フィールドはブランクになり、フィールドが空であるという現実の状態が反映されます。この処理は、READC 演算命令によりデータがプログラムに移動された後で行われる必要があります。

サブファイルに関する入出力操作の要求

呼び出し側プログラムによるサブファイル・レコード様式への入出力要求により、レコードがサブファイルに書き込まれるか、レコードがサブファイルから読み取られますが、ディスプレイへの実際の入出力は決して行われません。サブファイル・レコードをディスプレイに書き込むには、プログラムからサブファイル制御レコード様式に要求を発行しなければなりません。

サブファイルに対して行える有効な要求は、その要求がサブファイル・レコード様式かサブファイル制御レコード様式のどちらに対して行われるかによって異なります。

サブファイル・レコード様式に関する入出力操作の要求

サブファイル・レコード様式 に関する正しい入出力操作を要求すると、以下のことを行えます。

- サブファイル中の指定された場所に、(プログラムから渡された) レコードを追加する
- サブファイル中の既存の活動状態のレコードを更新する
- サブファイル中の指定された場所にある活動状態のレコードを読み取る
- 相対取得操作または次変更読み取り操作で以前に読み取られた相対レコード番号より大きい、サブファイル中の次の変更済みレコードを読み取る

サブファイル中の指定された場所にレコードを追加する

相対挿入操作により、サブファイル中の指定された場所に、(プログラムから渡された) レコードが追加されます。

この場所は、サブファイル中の有効な相対レコード番号でなければなりません。最小の相対レコード番号は常に 1 です。サブファイル・サイズがサブファイル・ページと等しい場合は、相対レコード番号の最大値はサブファイル・サイズの値になります。サブファイル・サイズがサブファイル・ページより大きい場合は、必要に応じてシステムがサブファイルを自動的に拡張するので、相対レコード番号の最大値は 9999 になります。また、相対レコード番号は、サブファイル中の既存の活動状態のレコードの番号になることはできません。サブファイル・レコードに関するフィールド選択が指定されている場合は、相対レコード番号は無視されます。

相対挿入操作によりサブファイル中の最終レコードの場所 (サブファイル・サイズの値) にレコードが追加されると、サブファイルがいっぱいの状態になります (メッセージ CPF5003)。ILE RPG および ILE COBOL の両方とも、この状態をアプリケーション・プログラムに通知する特殊なサポートがあります。該当する高水準言語の資料を参照してください。

デフォルト値が指定されていない入力可能フィールドの内容は、以下のように処理されます。

- 数字フィールドはゼロに初期設定される。
- 文字フィールドはブランクに初期設定される。
- 浮動小数点フィールドはヌルに初期設定される。

サブファイル中の活動状態のレコードを更新する

更新操作により、サブファイル中の既存の活動状態のレコードが更新されます。

更新操作の前に、取得要求 (相対取得操作または次変更読み取り操作) により、活動状態のレコードが読み取られていなければなりません。読み取りと更新の間に、更新されるサブファイルに対する他の入出力操作を実行できません。また、読み取りと更新の間に、(例えば、サブファイルのロールまたは SFLDROP 処理を使用して) 更新されるサブファイルを再表示できません。

注:

1. 高水準言語の中には、ディスプレイ・ファイル中の単一のサブファイル・レコードの読み取りと更新の間に、ディスプレイ・ファイル中の様式に入出力できないものもあります。詳細な情報については、ご使用の高水準言語の資料を参照してください。
2. サブファイル・レコードに関するフィールド選択が指定されている場合、サブファイル中にレコードが挿入された際に選択されたフィールドのみ更新できます。さまざまなフィールドを選択すると、予測できない結果になります。

サブファイル中の指定された場所で活動状態のレコードを読み取る

相対取得操作により、サブファイル中の指定された場所にある活動状態のレコードが読み取られます。

この場所は、サブファイル中の有効な相対レコード番号でなければなりません。応答標識 (ファイル・レベルで定義されているか、またはサブファイル・レコード中のフィールドで定義されている)、入力、出力、出力/入力、および隠しフィールドを含むレコード全体がプログラムに渡され、相対レコード番号が入出力フィールドバック域に入れられ、レコードが変更済みレコードとして識別されなくなります。ファイル・レベルで定義されている応答標識は、常にオフとして戻されます。BLANKS キーワードや CHANGE キーワードなどの、サブファイル・レコード中のフィールドで定義されている応答標識は、取得操作が行われた時点のフィールド中の情報に応じて、オンまたはオフとして戻されます。

相対取得操作で指定されたレコードが活動状態でない場合は、レコード番号が無効な状態になります (メッセージ CPF5020)。高水準言語によっては、この状態はレコード不在状態になります。該当する高水準言語の資料を参照してください。

注:

1. 相対取得操作と次変更読み取り操作は両方とも、入出力フィールドバック域の相対レコード番号を更新します。後続の次変更レコード読み取り要求は、この相対レコード番号より大きい順次変更されたレコードを検索します。
2. 相対取得操作と次変更読み取り操作は、サブファイル中の重複フィールドの入力データを処理しません。プログラムに戻されるレコードには、重複フィールドに関する読み取り操作の前にすでにバッファー中にあったデータが含まれます。このことが問題になる場合は、サブファイル初期設定機能を使用して、確実にすべてのサブファイル・フィールドが消去されるようにしてください。

サブファイル中の次の変更済みレコードを読み取る

次変更読み取り操作により、相対取得操作または次変更読み取り操作により以前に読み取られた相対レコード番号より大きい、サブファイル中の次の変更済みレコードが読み取られます。

最初の読み取り操作として次変更読み取り操作が使用される場合、サブファイル中の最初の変更済みレコードが読み取られます。応答標識 (ファイル・レベルで定義されているか、またはサブファイル・レコード中のフィールドで定義されている)、入力、出力、出力/入力、および隠しフィールドを含むレコード全体がプログラムに渡され、相対レコード番号がデータ管理フィールドバック域に入れられ、レコードが未変更レコードにリセットされます。ファイル・レベルで定義されている応答標識は、常にオフとして戻されます。BLANKS キーワードや CHANGE キーワードなどの、サブファイル・レコード中のフィールドで定義されている応答標識は、取得操作が行われた時点のフィールド中の情報に応じて、オンまたはオフとして戻されます。

サブファイル中に変更済みのレコードがなくなると、最後の更新済みレコードがすでに検索されていることを示すメッセージ (CPF5037) がプログラムに送信されます。この状態がプログラムに報告される方法に関する説明は、該当する高水準言語の資料を参照してください。

次更新読み取り操作によって検索されたレコードが更新された場合に、更新されたレコードに関する SFLNXTCHG キーワードが指定されていると、更新されたレコードは更新済みのレコードとして再設定されます。この処理により、ユーザーがレコードを変更したことをプログラムが確認できます。例えば、プログラムがレコード中のエラーを検出した場合、ユーザーがエラーを訂正する必要がある方が好都合です。SFLNXTCHG キーワードを使用すると、次更新読み取り操作でプログラムがこのレコードを再び読み取ることができるので、エラーが訂正されるまでレコードをリジェクトし続けることができます。次の更新済み読み取り操作でこの更新済みレコードは検索されません。サブファイル中の後続の更新済みレコードがすべて処理されるまで、次更新読み取り操作でこのレコードを再び検索することはできません。その理由は、更新済みのレコードは順次アクセスされますが、サブファイル中に更新済みレコードがなくなったことを示すメッセージがプログラムに送信された後まで、その順序は先頭から始まらないからです。このメッセージの後に行われる次更新読み取り操作では、サブファイル中の最初の変更済みレコードが取得されます。ディスプレイに対する入出力操作が発行されていないので、更新済みのレコードは、すべて SFLNXTCHG キーワードを使用して処理されたこととなります。

注:

1. 相対取得操作と次更新読み取り操作は両方とも、入出力フィールドバック域の相対レコード番号を更新します。後続の更新レコード読み取り要求は、この相対レコード番号より大きい順次更新されたレコードを検索します。
2. 相対取得操作と次更新読み取り操作は、サブファイル中の重複フィールドの入力データを処理しません。プログラムに戻されるレコードには、重複フィールドに関する読み取り操作の前にすでにバッファ一中にあったデータが含まれます。このことが問題になる場合は、サブファイル初期設定機能を使用して、確実にすべてのサブファイル・フィールドが消去されるようにしてください。

サブファイル制御レコード様式に関する入出力操作の要求

サブファイル制御レコード様式 に関する正しい入出力操作を要求すると、以下のことを行えます。

- サブファイル・レコードを表示する
- プログラムによる処理のために、ディスプレイ上のサブファイル・レコードをサブファイル中に挿入する
- サブファイル・レコードの表示と処理を同時に行う

サブファイル・レコードの表示

サブファイル制御レコード様式に対する書き込み操作を発行すると、サブファイル・レコードを表示できません。

以下の DDS キーワードを使用して、書き込み操作を制御できます。

SFLDSP サブファイルを表示します。

SFLDSPCTL サブファイル制御レコードを表示します。

SFLCLR 活動状態のレコードのサブファイルを消去します。

SFLDLT サブファイルを削除します。

SFLINZ 活動状態のレコードのあるサブファイルを初期設定します。または、SFLRNA キーワード

が指定されている場合は、非活動状態のレコードのあるサブファイルを初期設定します。サブファイルが初期設定されると、サブファイル・レコード中のオプション標識はすべてオフであると想定されます。

SFLEND 最後の使用可能なレコードが表示される際にそのことをユーザーに通知します。

SFLRCDNBR サブファイルの指定されたページを表示します。

注: これらのキーワードについては、97 ページの『DDS ソースでのサブファイルの記述』で説明されています。

処理のためにディスプレイ上のサブファイル・レコードを挿入する

プログラムによる処理のために、ディスプレイ上のサブファイル・レコードをサブファイル中に挿入するには、表示されているレコード様式に対する読み取り操作を発行しなければなりません。ディスプレイ上のサブファイル・レコードは、サブファイル中の対応するレコード位置に挿入されます。

サブファイル・レコードの表示と処理を同時に行う

書き込み・読み取り操作は、書き込み操作と読み取り操作を結合した単一の操作で、単一の書き込み操作の後で単一の読み取り操作を行うより効率的です。

高水準言語でのサブファイル入出力要求についての理解

表 14 は、システムによってサポートされている入出力要求と、同等の高水準言語操作を示しています。

表 14. システムによってサポートされるサブファイル操作および同等の HLL コマンド

操作	ILE C/C++ 関数	ILE RPG 演算命令	ILE COBOL ステートメント	BASIC ステートメント
相対挿入	_Rwrite	WRITE、出力仕様	WRITE SUBFILE	WRITE REC =
更新	_Rupdate	UPDAT	REWRITE SUBFILE	REWRITE REC =
相対取得	_Rread	CHAIN	READ SUBFILE	READ REC =
次変更読み取り	_Rreadnc	READC	READ SUBFILE NEXT MODIFIED	READ MODIFIED
書き込み	_Rwrite	WRITE	WRITE	WRITE
読み取り	_Rread	READ	READ	READ
書き込み・読み取り	_Rwriterd、_Rformat、 _Rpgmdev	EXFMT		

サブファイルの外観の制御

サブファイル中のレコードを縦または横の方向に表示できます。サブファイルを縦方向に表示すると、レコードは 1 つ以上の行として表示され、レコードごとに新しい行が開始されます (110 ページの図 22 を参照)。サブファイルを横方向に表示すると、レコード全体が 1 行になり、1 行に複数のレコードが表示されます (110 ページの図 23 を参照)。SFLLIN キーワードを使用して、表示行の個々のサブファイル・レコード間のスペース数を定義すると、サブファイルが横方向に表示されるように指定できます。110 ページの図 24 には、縦方向のサブファイルと横方向のサブファイルを同時に表示する例が示されています。

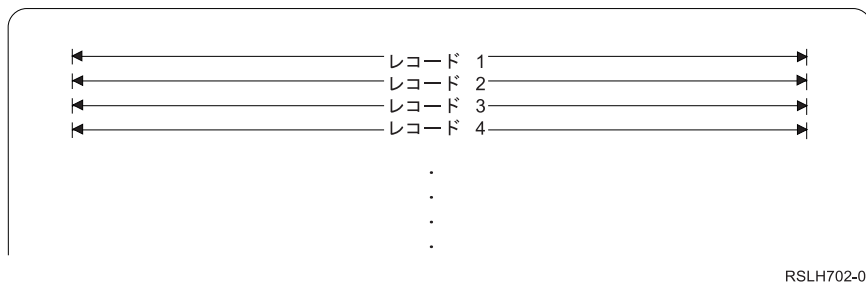


図 22. サブファイルを縦方向に表示する

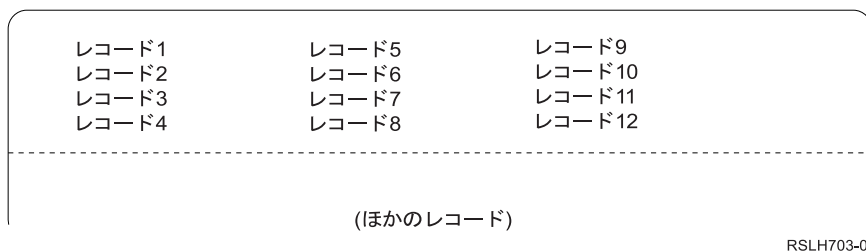


図 23. サブファイルを横方向に表示する

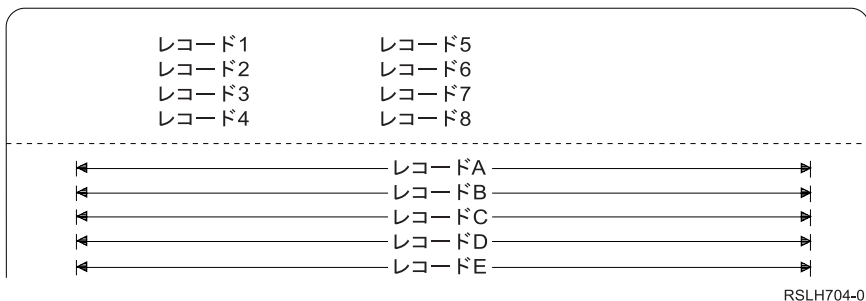


図 24. 横方向に表示するサブファイルと縦方向に表示するサブファイルを同時に表示する

あるサブファイルが、画面上のサブファイルに使用できるスペースより大きい場合は、ユーザーは表示をそのサブファイル中の 1 つのレコード・グループからロールできます。同時に表示されるレコードのグループのことをページと呼びます。サブファイルのあるディスプレイ・ファイルを作成する際には、ページ内のレコードの数を指定して (SFLPAG キーワード)、サブファイルのページのサイズを指定しなければなりません。普通、ページ・サイズはディスプレイで使用できる行数に基づきます。また、サブファイル内のレコードの数を指定して (SFLSIZ キーワード)、サブファイルのサイズも指定しなければなりません。

ページ・サイズとサブファイル・サイズを同じにすることができます。この場合、サブファイル中のすべてのレコードが 1 ページに収まります。ページ・サイズとサブファイル・サイズが等しい場合、可変長サブファイル・レコードがサポートされます。あるレコードを 1 行に収め、別のレコードを複数の表示行にすることができます。個々のレコードは、サブファイル中の使用可能な最初のレコード位置に入れられます。この位置は常に新しい行になります。さらに、ページ・サイズとサブファイル・サイズが等しい場合には、SFLDROP キーワードと SFLROLVAL キーワードはディスプレイ装置サポートによって無視されます。

ページ・サイズとサブファイル・サイズについて詳しくは、112 ページの『ページ・サイズと等しいサブファイル・サイズの指定』と、114 ページの『ページ・サイズと等しくないサブファイル・サイズの指定』を参照してください。

レコードを横方向に表示する場合には、サブファイル中の表示されるレコードの数 (SFLPAG キーワード) が調整されるので、画面上の最後の行を使用してレコードの行全体を表示できます。例えば、行の個々のレコード間のスペース数 (SFLLIN キーワード) を、1 行に 6 つのレコードが収まるように指定し、ページ・サイズ (SFLPAG キーワード) を 20 と指定した場合、20 は最も近い 6 の倍数である 24 に変更されます。サブファイル中のレコードの数 (SFLSIZ キーワード) は、同じ数ずつ増えていきます。

注: サブファイルの初期画面の場合、表示前にサブファイルに挿入されるレコードが多いほど、応答時間は遅くなります。

表示モードでサブファイルを横方向に表示する

表示モード (DSPMOD) キーワードを使用して、24x80 または 27x132 の 2 つのモード (またはディスプレイ・サイズ) のどちらかをディスプレイ装置に使用するか指定できます。

表示モードを変更すると、ディスプレイは消去されますが、データはサブファイルから消去されません。制御レコード中で DSPMOD を活動状態にするには、SFLDSP または SFLDSPCTL が有効でなければなりません。

以下の例は、DSPMOD とサブファイルの指定方法を示しています。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     DSPSIZ(*DS4 *DS3)
A           R SFLR                     SFL
A           FLD1                       8  0  1  5
A           FLD2                       7  I  1 16
A           FLD3                       7  B  1 24
A           R SFLCTLR                   SFLCTL(SFLR)
A                                     SFLDSP
A                                     SFLDSPCTL
A                                     SFLSIZ(60)
A                                     SFLPAG(12)
A                                     SFLLIN(4)
A *DS3                                 SFLIN(6)
A 02                                  SFLEND
A 10                                  DSPMOD(*DS3)
A
A
```

図 25. DSPMOD とサブファイルを併用したサンプル DDS

上記の例で、ユーザーのプログラムが標識 10 をオフにして、サブファイル制御レコード様式に対する書き込み・読み取り操作を発行する (SFLCTLR) と、サブファイルは以下のように表示されます。

- 27 x 132 (*DS4) モード。DSPMOD キーワードの標識 10 がオフであるため。
- 横方向。SFLLIN が指定されているため。SFLLIN 値はレコード間のバイト数を示します。個々のレコードの長さは 30 バイトで、個々のレコード間のスペースの長さは 4 バイトなので、横方向の 1 行に 4 つのレコードを表示でき、 $(4 \times 30) + (3 \times 4) = 132$ バイトになります。SFLPAG(12) が指定されているので、サブファイルは 3 行で表示されます。

以下の例は、*DS4 モードで表示されるサブファイルを示しています。

レコード 1	レコード 4	レコード 7	レコード 10
レコード 2	レコード 5	レコード 8	レコード 11
レコード 3	レコード 6	レコード 9	レコード 12

ユーザーが Enter キーを押すと、制御がユーザーのプログラムに戻されます。ユーザーのプログラムが標識 10 をオンにしてから、サブファイル制御レコード様式に対する別の書き込み・読み取り操作を発行する (SFLCTLR) と、サブファイルは以下のように表示されます。

- 24 x 80 (*DS3) モード。DSPMOD キーワードの標識 10 がオンなので。
- 横方向。*DS3 モードに SFLLIN が指定されているので。*DS3 モードに SFLLIN が指定されなかった場合は、サブファイルは縦に表示されます。複数の画面サイズに SFLLIN キーワードを使用する場合、2 次画面サイズごとに画面サイズ条件名が必要です。個々のレコードの長さは 30 バイトで、個々のレコード間のスペースの長さは 6 バイトなので、横方向の 1 行に 2 つのレコードを表示でき、(2 x 30) + 6 = 66 バイトになります。SFLPAG(12) が指定されているので、サブファイルは 6 行で表示されま
す。他のレコードが確実に消去されないようにするには、2 次画面サイズに SFLPAG を指定する必要が
生じることがあります。

以下の例は、*DS3 モードで表示されるサブファイルを示しています。

レコード 1	レコード 7
レコード 2	レコード 8
レコード 3	レコード 9
レコード 4	レコード 10
レコード 5	レコード 11
レコード 6	レコード 12

ページ・サイズと等しいサブファイル・サイズの指定

SFLSIZ キーワードとSFLPAG キーワードを指定して、サブファイルのサイズと一度に表示されるサブファイル・レコードの数を指定しなければなりません。表示されるサブファイル・レコードの数が 1 ページに収まる場合、またはサブファイル中に入れられるレコードの数が不明だが多い場合には、ページ・サイズと等しいサブファイル・サイズを使用することをお勧めします。通常ユーザーが先頭ページに必要な情報を見つける場合は、大規模なサブファイルを満たす多数のデータベース・レコードを検索することは効率的なりソースの使用法ではありません。

サブファイル・サイズがページ・サイズと等しい場合、システムは次ページ・キーと前ページ・キーの使用を自動的にサポートしません。ユーザーがこれらのキーを使用してサブファイルをロールできるようにする場合は、サブファイル制御レコードに ROLLUP キーワードか ROLLDOWN キーワードを指定しなければならず、ご使用のプログラムで次ページ機能または前ページ機能を処理しなければなりません。

例えば、サブファイルを使用してユーザーが長いリストを検索できるようにする場合は、サブファイル制御レコードに SFLSIZ、SFLSIZ と等しい SFLPAG、および ROLLUP を指定できます。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R SFLCTLR          SFLCTL(SFLRCD)
A          SFLSIZ(16)
A          SFLPAG(16)
A          ROLLUP(20 'Ro11 Up')
```

図 26. SFLSIZ、SFLPAG、および ROLLUP キーワードを使用したサンプル DDS

ユーザーが次ページ・キーを押すと、標識 20 がオンに設定され、制御がプログラムに戻されます。ご使用のプログラムで、以下のようになります。

- サブファイルを消去する (SFLCLR キーワードが有効なサブファイル制御レコード様式に対する書き込み操作)。
- 標識を使用して、別のレコード・ページでサブファイルを充てんする論理に制御を戻す。
- 新しいサブファイル・ページを表示する。

また、ディスプレイ装置のユーザーが、CFnn キーを押して検索の開始に戻れるようにすることもできます。ユーザーが CFnn キーを押すと、関連した標識がオンに設定され、制御がプログラムに戻されます。ご使用のプログラムで、以下のようになります。

- サブファイルを消去する。
- 標識を使用して、入力した検索コードに基づいて先頭のサブファイル・ページを構築する論理に制御を戻す。(そのためには、プログラムが元の検索コードを保持する必要があります。)

サブファイル・サイズがページ・サイズと等しい場合に **ROLLDOWN** キーワードを使用すると、プログラム中の必要なコードの行数が多くなります。なぜなら、プログラムがサブファイルおよびデータベース・ファイル中のレコードの位置を追跡しなければならないからです。

サブファイル・サイズがページ・サイズと等しい場合、サブファイル中でフィールド選択と可変長レコードを使用できます。フィールド選択を使用する場合、以下の事項について考慮してください。

- オプション標識を使用してフィールドを選択する場合、相対レコード番号は無視され、個々のレコードはサブファイル中の最初の使用可能なレコード位置に挿入される。
- レコードが更新される場合、元の出力の内容と一致しないフィールド選択は無視される。例えば、レコードがサブファイル中に挿入される際に **FIELD1** と **FIELD2** が選択されると想定します。更新時に **FIELD2** と **FIELD3** が選択されると、フィールドは元の **FIELD1** および **FIELD2** フィールドをオーバーレイし、結果は予測できなくなります。
- サブファイル・レコード上でフィールド選択を指定する場合、画面に表示できるレコードの数は、選択されたフィールドの数に応じて異なる。フィールド選択を指定する際に、**SFLPAG(value)** キーワードはサブファイル・レコードの表示に使用できる画面の行の数を指定します。その他の場合には、**SFLPAG(value)** キーワードは一度に表示できるサブファイル・レコードの数を指定します。
- **SFLFOLD**、**SFLDROP**、および **SFLROLVAL** キーワードは無視される。

可変長レコードを使用する場合、サブファイル中の個々のレコードは、ページ上の最初の使用可能な行の先頭に表示されます。可変長レコードに関してフィールド選択を使用する場合、ディスプレイ上の行数はレコードごとに異なります。したがって、実際にサブファイル中に収まるレコードの数は、サブファイルに書き込まれるレコードごとのフィールド選択に応じて異なります。以下に、可変長レコードの場合の **DDS** の例を示します。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R SFLRCD          SFL
A          ITMNR           8Y 0   6 2
A          ITMDSC          15     6 11
A          QTYOH           4 0    6 28
A          LSTPC           7 2    6 39
A          ALLOH           8Y 0   6 49
A          SLSMO           10 2    6 63
A N80     SLSYR            12 2    7 7
A N80     CSTYR            12 2    +3

```

図 27. 可変長レコードの場合のサンプル DDS

標識 80 がオンの場合は、サブファイル中の個々のレコードは 1 行に収まります。しかし、標識 80 がオフの場合は、ディスプレイ上で個々のレコードに 2 行使用されます。

もう 1 つの代表的な可変長レコードの使用例として、複数の全く異なる様式を使用して 1 つの様式を形成するというものがあります。この場合、個々のフィールドがオプション標識によって別々に条件付けされ、あるレコード様式は複数の行を使用でき、別の様式は 1 行のみ使用するようになります。

ページ・サイズと等しくないサブファイル・サイズの指定

サブファイル中に挿入できるレコードの数が有限で、その数が少ない (50 など) 場合、ページ・サイズと等しくないサブファイル・サイズを使用する必要があります。SFLSIZ キーワードは、サブファイル・サイズを指定します。システムは、SFLSIZ の指定値に基づいて、サブファイル・レコードを含めるスペースを割り振ります。通常サブファイル中にあるレコードの数と等しい値を指定する必要があります。ご使用のプログラムが SFLSIZ 値より大きい相対レコード番号のレコードをサブファイル中に挿入する場合は、システムはサブファイルを拡張してそのレコードを含めます (最大 9999 のレコード)。

サブファイル・サイズがページ・サイズと等しくない場合、次ページ・キーと前ページ・キーの使用は自動的にサポートされます。

サブファイル中にさらにレコードがあることをユーザーに通知するには、サブファイル制御レコードで SFLEND キーワードを使用します。SFLEND が有効な場合 (例えば、オプション標識がオンの場合) は、最終ページ以外のページごとに画面の右下に + (正符号) が表示されます。最終サブファイル・ページでは、+ がブランクに置き換えられます。

サブファイル・サイズがページ・サイズと等しくない場合には、SFLROLVAL キーワードを使用することにより、ユーザーが値を入力して、該当するキーを押した際にロールアップまたはロールダウンされるレコードの数を指定できるようにすることができます。SFLROLVAL キーワードを使用しないと、SFLFOLD または SFLDROP を使用しているサブファイルを除いて、SFLPAG 値によってサブファイルがロールされます。SFLFOLD または SFLDROP キーワードを使用すると、レコードが切り捨て様式で表示される際に、SFLPAG 値より多くのレコードが表示されます。切り捨てられたレコードの場合、ディスプレイは切り捨て様式で表示されるレコードの数ずつロールします。SFLROLVAL キーワードの使用時に次ページ・キーを押すと、表示されているサブファイル中の最上部のレコード番号がロール値に追加され、新しい最上部のレコード番号が決められます。この値がサブファイル中の最終レコードより大きい場合は、最終レコード・ページ全体が表示されます。ロール値がページ・サイズ値以上の場合、最終サブファイル・ページが表示されている際に次ページ・キーを押すと、エラー・メッセージが発行されます。ロール値がページ・サイズ値より小さい場合は、ロール機能が実行されます。

サブファイル・サイズがページ・サイズと等しくない場合には、可変長レコードやフィールド選択を使用できません。

複数のページがあるサブファイルを使用する際のパフォーマンスを改善する手法の 1 つに、一度にサブファイル・レコードの 1 ページのみ書き込み、その一方でオペレーティング・システムのサポートを使用してサブファイル全体をロールすることがあります。そのためには、応答標識を指定して DDS で ROLLUP キーワードを定義し、さらに SFLRCDNBR キーワードも使用する必要があります。ご使用のプログラムでは、1 つのサブファイル・ページを充てんするのに必要なレコードを書き込んでから、そのページを表示します。ユーザーが追加のレコードを参照する際には、次ページ・キーを押します。次にプログラムは別のレコード・ページをサブファイルに書き込み、レコードの相対レコード番号を 2 ページ目から SFLRCDNBR フィールドに挿入して、そのレコードを表示します。

この時点でサブファイル・レコードの 2 ページ目が表示され、ユーザーが前ページ・キーを押すと、システムによってロールダウンが処理されます。先頭ページが表示されている際にユーザーが次ページ・キーを押すと、システムはロールアップも処理します。プログラムが通知を受けるのは、ユーザーがサブファイル内の現行レコードを超えてロールアップを試行する場合だけです。次にプログラムは追加のロールアップ要求を 2 ページ目の場合と同じ方法で処理します。この手法を使用すると、ロール・キーの使用により、サブファイルに複数のページがあるかようになります。それでも、プログラムは画面に書き込む前に 1 つのサブファイル・ページを埋めるだけなので、優れた応答時間を維持できます。

サブファイル・データに関する妥当性検査

DDS の妥当性検査キーワード (CHECK、COMP/CMP、RANGE、および VALUES) に加えて、プログラム中のサブファイル・データに関する妥当性検査を実行でき、この場合ユーザーがエラーを訂正する必要があります。

例えば、受注プログラムに関するサブファイルを使用しており、品目番号フィールドを検査して有効なオーダー番号を確認すると想定します。また、オーダー数量フィールドを検査して、オーダーを満たす品目が手元に十分あるか確認するとします。そのためには、サブファイル・レコード (SFL) に SFLNXTCHG キーワードを使用して、プログラムがエラーを診断し、ユーザーがそれらのエラーを訂正する必要が生じることができるようにすることができます。以下の DDS には、SFLNXTCHG キーワードの使用例が示されています。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R ORDENTD
A          1 30'ORDER ENTRY DISPLAY'
A          3 2'Enter customer number:'
A          CUST          5      3 25
A          R SFLRCD          SFL
A 61          SFLNXTCHG
A          LINNBR        2      7 4
A          ITMNBR        5      B 7 9
A 40          DSPATR(RI PC)
A          QTYORD        4      B 7 20
A 35          DSPATR(RI PC)
A          R SFLCTLR          SFLCTL(SFLRCD)
A          SFLSIZ(5)
A          SFLPAG(5)
A 55          SFLDSP
A 50          SFLDSPCTL
A 30          SFLCLR
A 10          SFLINZ
A 40          SFLMSG('Item number not valid' 40)
A 35          SFLMSG('Qty not available' 35)
```

図 28. SFLNXTCHG キーワードを使用したサンプル DDS

プログラムがエラーを検出すると、SFLNXTCHG キーワードを条件付ける標識をオンに設定し、SFLDSP キーワードが有効なサブファイル制御レコードに対する書き込み操作を発行します。エラー状態のフィールドは反転表示され、カーソルがそのフィールドに置かれます。関連したエラー・メッセージが表示されません。続いて、ユーザーがエラーを訂正します。

SFLNXTCHG キーワードの使用時には、ユーザーがエラー状態でなかったサブファイル・フィールドに変更を加えられるようにするかどうか決定しなければなりません。ディスプレイ装置のユーザーがこれらのフィールドに変更を加えられないようにする場合は、DSPATR(PR) キーワードを指定して、これらのフィールドを保護できます。変更を加えられないようにするフィールドの場合、SFLNXTCHG キーワードが有効な場合に限り DSPATR(PR) キーワードが有効でなければなりません。ユーザーがフィールドに変更を加えられるようにするには、以下のようにします。

- これらのフィールドの検査対象の隠しフィールドを定義する。
- サブファイル上でエラーが起きた際に、元々ユーザーが入力したデータを隠しフィールドに移動する。
- 隠しフィールド中のデータを読み取ったばかりのフィールドと比較して、変更が加えられたフィールドを識別し、ユーザーが変更を加えた時点ですでに処理済みだったレコードを更新できるようにする。

サブファイルからのエラー・メッセージの表示

サブファイルを使用して、複数のエラーに関するメッセージを表示できます。サブファイル中に挿入されるメッセージは、プログラム・メッセージ待ち行列に入れられます。サブファイルに書き込まれるメッセージごとに別の行に表示され、必要に応じて切り捨てられます。個々のメッセージ行の位置 1 には属性文字が含まれ、ブランクとして表示されます。その後メッセージ・テキストが表示されます。24 x 80 表示モードの場合、76 文字が表示されます。27 x 132 表示モードの場合、128 文字が表示されます。プログラム・メッセージ待ち行列からメッセージ ID とメッセージ・データの両方を使用できるので、メッセージ・サブファイル中に挿入されるメッセージに関するメッセージ・ヘルプと置換テキストがサポートされます。SFLMSGRC D キーワードを指定する場合は、SFLPGMQ キーワードと SFLMSGKEY キーワードも指定しなければなりません。

以下に、メッセージ・サブファイルの場合の DDS の例を示します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R MSGSFL          SFL
A          SFLMSGRC D(14)
A          MSGKEY          SFLMSGKEY
A          PGMQ            SFLPGMQ
A          R MSGCTL        SFLCTL(MSGSFL)
A          SFLSIZ(8)
A          SFLPAG(8)
A 50          SFLDSP
A 55          SFLDSPCTL
A 60          SFLINZ
A          PGMQ            SFLPGMQ
A          NBR              4 0H    SFLRCDNBR(CURS O R)
A
```

図 29. メッセージ・サブファイルの場合のサンプル DDS

SFLRCDNBR(CURS O R) キーワードを使用して、SFLRCDNBR フィールドで指定されているメッセージ・サブファイル中の最初に表示される文字にカーソルを置き、次ページ・キーと前ページ・キーがメッセージ・サブファイルに適用されるようにします。

メッセージの送受信と、プログラム・メッセージ待ち行列に関する情報は、iSeries Information Center の『CL』トピックの『Messages』セクションを参照してください。制御言語プログラムおよび C/C++ プログラム用にサポートされているサブファイルの種類は、メッセージ・サブファイルのみです。

表示されているサブファイル上のカーソルの位置決め

DSPATR(PC) キーワードを使用すると、表示されているサブファイル・レコードのページごとにカーソルの位置を決めることができます。書き込み操作と更新操作を使用して、以下に関する DSPATR(PC) を制御できます。

- サブファイル・レコードの初期画面。(SFLDSP キーワードおよび SFLDSPCTL キーワードを使用している場合は、サブファイル制御レコードに対する書き込み操作または書き込み・読み取り操作。)
- ロール・キー、または継続キーか切り捨てキーを使用して表示されるサブファイル・レコード (SFLDROP キーワードまたは SFLFOLD キーワード)。

カーソルの最初の位置決め

以下の条件のうち適用される最初の条件によって、カーソルの位置が決められます。

- サブファイル制御レコード上の CSRLOC キーワード
- 表示されているレコード中の DSPATR(PC) キーワード
- サブファイル制御レコード中のフィールド内の DSPATR(PC) キーワード
- サブファイル制御レコード中の SFLRCDNBR(CURSOR) キーワード
- 何も指定していない場合は、カーソルはディスプレイ上の最初の入力可能フィールドに置かれる。

注: サブファイルを表示する出力操作の前にキーボードがアンロック状態になっている場合は、明示的なカーソルの位置決めは実行されません。

以下の順序でキーワードを使用します。

- CSRLOC キーワードを使用して、画面上の任意の位置にカーソルを置くことができる。
- DSPATR(PC) キーワードを使用して、出力操作で SFLDSP キーワードが指定された際に表示される最初のレコード中の任意のフィールドにカーソルを置くことができる。
- DSPATR(PC) キーワードを使用して、サブファイル制御レコードの任意のフィールドにカーソルを置くことができる。
- SFLRCDNBR(CURSOR) キーワードを使用して、最初に表示するページの選択に使用されるレコード番号のあるレコードの最初の入力可能フィールドにカーソルを置くことができる。入力フィールドがない場合は、カーソルはこのレコード中の最初に選択された出力フィールドに置かれる。
- DSPATR(PC) キーワードと SFLRCDNBR(CURSOR) のどちらも使用しない場合は、カーソルはディスプレイ上の最初の入力可能フィールドに置かれる。

ロール・キー使用時のカーソルの位置決め

ロール・キーを使用する際のカーソルの位置は、DSPATR(PC) キーワードを使用しているかどうかに応じて異なります。

- DSPATR(PC) キーワードを使用しない場合は、カーソルはロール・キーを押した時点の位置に置かれる。
- DSPATR(PC) キーワードを使用する場合、カーソルは、表示されていて DSPATR(PC) キーワードが有効になっているサブファイル・レコード中の最初のフィールドに置かれる。

以下の例は、両方の場合を図示し、レコードが縦方向に表示されるサブファイルに関する DDS の一部を示しています。顧客番号、名前、住所、市区町村、および都道府県が表示されます。ユーザーは顧客名、住所、市区町村、および都道府県を変更できます。顧客番号は出力フィールドのみなので変更できません。顧客番号フィールド (CUST) に関する DSPATR(PC) キーワードが指定されています。サブファイル・サイズは 21 で、ページ・サイズは 7 です。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R SUBFIL1                SFL
A          TEXT('Subfile record')
A          CUST          5          4 3DSPATR(PC)
A          NAME         20         B 4 10CHECK(LC)
A          ADDR         20         B 4 32CHECK(LC)
A          CITY         20         B 4 54CHECK(LC)
A          STATE        2          B 4 76
A          R FILCTL1                SFLCTL(SUBFIL1)
A 50          SFLDSPCTL
A 55          SFLDSP
A          SFLSIZ(21)
A          SFLPAG(7)
A 60          SFLCLR
A          TEXT('Subfile control record')
A          OVERLAY
A          PROTECT
A          CA03(98 'End of program')
A          2 2'NUMBER'
A          2 10'NAME'
A          2 32'ADDRESS'
A          2 54'CITY'
A          2 76'STATE'

```

図 30. DSPATR(PC) キーワードを使用したサンプル DDS

初期画面は以下のようになります。

カーソル

NUMBER	NAME	ADDRESS	CITY	STATE
41394	Sorensen and Walton	500 5th Avenue	New York	NY
41395	Charland, Inc.	200 Madison Avenue	New York	NY
41316	Anderson's Electric	950 2nd Avenue	Atlanta	GA
41397	Morem Motors	1300 Pine Street	Atlanta	GA
41398	Polt Electronics	240 Walters Place	Chicago	IL
41399	Clark's TV	560 3rd Street	Chicago	IL
41400	Jim's Repair	700 4th Avenue	Chicago	IL

RSLH709-0

サブファイル中の最初の 7 つのレコードが表示され、最初のレコード中の顧客番号の下にカーソルが置かれます。ユーザーはカーソルを 3 つ目のレコードに移動し、この顧客の住所を更新し、カーソルを 4 つ目のレコードの顧客番号に移動します。

カーソル

変更されたフィールド

NUMBER	NAME	ADDRESS	CITY	STATE
41394	Sorensen and Walton	500 5th Avenue	New York	NY
41395	Charland, Inc.	200 Madison Avenue	New York	NY
41316	Anderson's Electric	950 2nd Avenue	Atlanta	GA
41397	Morem Motors	1300 Pine Street	Atlanta	GA
41398	Polt Electronics	240 Walters Place	Chicago	IL
41399	Clark's TV	560 3rd Street	Chicago	IL
41400	Jim's Repair	700 4th Avenue	Chicago	IL

RSLH183-0

この時点でユーザーは、次ページ・キーを押して次の 7 つのレコードを表示します。最初のレコード中の顧客番号の下にカーソルが置かれます。

カーソル

NUMBER	NAME	ADDRESS	CITY	STATE
41401	Adam's Home Repair	121 Golden Circle	Chicago	IL
41402	Jane's Radio/TV	135 Ransom Drive	St Paul	MN
41403	Advanced Electronics	809 8th Street	St Paul	MN
41404	Riteway Repair	443 Western Lane	New York	NY
41405	Fixtures, Inc.	607 9th Avenue	Chicago	IL
41406	Hall's Electric	200 Main Street	St Paul	MN
41407	Electric House	903 East Place	Atlanta	GA

RSLH184-0

DSPATR(PC) キーワードが指定されていないので、ユーザーが次ページ・キーを押すと、4 つ目のレコードの顧客番号の下にカーソルが置かれます。

カーソル

NUMBER	NAME	ADDRESS	CITY	STATE
41401	Adam's Home Repair	121 Golden Circle	Chicago	IL
41402	Jane's Radio/TV	135 Ransom Drive	St Paul	MN
41403	Advanced Electronics	809 8th Street	St Paul	MN
41404	Riteway Repair	443 Western Lane	New York	NY
41405	Fixtures, Inc.	607 9th Avenue	Chicago	IL
41406	Hall's Electric	200 Main Street	St Paul	MN
41407	Electric House	903 East Place	Atlanta	GA

RSLH185-0

継続または切り捨てキー使用時のカーソルの位置決め

SFLFOLD または SFLDROP キーワードを使用して CFnn または CAnn キーを割り当てると、カーソルの位置決めは、117 ページの『ロール・キー使用時のカーソルの位置決め』の説明と同じ方法で処理されます。カーソルは、継続されているレコードを含む、すべての表示されているレコードの指定どおりに位置決めされます。SFLFOLD または SFLDROP キーワードを使用して CFnn または CAnn キーを割り当てると、これらのフィールドのうち表示されているフィールドのみによりカーソルが位置決めされます。レコードの継続している部分のフィールドに関するカーソル位置の指定は、無視されます。

複数のレコードが表示される場合のカーソルの位置決めとロール

複数のレコードを同時に表示している場合、どのレコードがディスプレイに最後に書き込まれたかに関係なく、カーソルの位置により、ユーザーがロール・キーを押した際のアクションが決まります。

カーソルは、画面のロール可能域 か、または画面のロール可能でない 領域に置くことができます。ロール可能域 には以下のものがあります。

- サブファイルがなく、ROLLUP/ROLLDOWN キーワードが有効なレコード
- ROLLUP/ROLLDOWN キーワードが有効なサブファイル制御レコード
- ロール可能サブファイル。これは、サブファイル・サイズがページ・サイズより大きい活動状態のサブファイルのことです。
- サブファイル・サイズがページ・サイズと等しく、サブファイル制御レコードに関する ROLLUP/ROLLDOWN キーワードが有効な活動状態のサブファイル

カーソルの位置に基づいて、ユーザーがロール・キーを押す際のアクションは以下のようになります。

- ロール可能域の、ロール可能サブファイルまたはロール可能サブファイルのサブファイル制御レコードにカーソルが置かれている場合は、サブファイルはロールされます。サブファイルがサブファイルの末尾にあり、対応する ROLLUP/ROLLDOWN キーワードが無効な場合は、サブファイルの終わりを示すメッセージがユーザーに送信されます。サブファイルの末尾で ROLLUP/ROLLDOWN キーワードが有効な場合は、制御がプログラムに戻されます。
- ロール可能域の、サブファイルがなく ROLLUP/ROLLDOWN キーワードが有効なレコード、ROLLUP/ROLLDOWN キーワードが有効なサブファイル制御レコード、またはサブファイル・サイズがページ・サイズと等しくサブファイル制御レコードに関する ROLLUP/ROLLDOWN キーワードが有効な活動状態のサブファイルにカーソルが置かれている場合は、制御がプログラムに戻されます。
- ロール可能域にカーソルが置かれていない場合は、システムは画面上の最上部のロール可能域の検出を試行し、上記のリストに示されているアクションを実行します。画面上にロール可能域がない場合は、コマンド・キーが無効であるというメッセージがユーザーに送信されます。

注: 表示スペースを占有しないレコード (フィールドがないか、隠しフィールド、プログラム-システム間フィールド、またはメッセージ・フィールドのみがあるか、または CLRL キーワードが指定されていて入力可能フィールドがないレコード様式、およびメッセージ・サブファイル) は行 0 にあると想定されます。したがって、これらのレコードは画面の最上部にあると見なされ、システムは最初にこれらのレコードのロールを試行します。

以下の例は、カーソルの位置に基づいて、2 つのレコードが表示されている場合にユーザーがロール・キーを押すと取られるアクションを図示しています。

以下の例で、対応する ROLLUP/ROLLDOWN キーワードが有効な場合は、制御はプログラムに戻されません。その理由は、サブファイルがなく、ROLLUP/ROLLDOWN キーワードが有効なレコードにカーソルが置かれているからです。

NUMBER	NAME	ADDRESS	CITY	STATE	
41394	Sorensen and Walton	500 5th Avenue	New York	NY	
41395	Charland, Inc.	200 Madison Avenue	New York	NY	
41316	Anderson's Electric	950 2nd Avenue	Atlanta	GA	
41397	Morem Motors	1300 Pine Street	Atlanta	GA	
41398	Polt Electronics	240 Walters Place	Chicago	IL	+

Enter next customer number: _____

カーソル

RV2W050-0

サブファイル
SFLSIZ>
SFLPAG

サブファイル
以外のレコード
ROLLUP/
ROLLDOWN
が有効

以下の例では、サブファイル・サイズがページ・サイズより大きい 2 つのサブファイルと、それらのサブファイルの制御レコードが表示されています。ユーザーはサブファイル制御レコードの下部にカーソルを置

いています。ディスプレイのロール可能域内のロール可能サブファイルにカーソルが置かれているので、下部のサブファイルはロールされます。

NUMBER	NAME	ADDRESS	CITY	STATE	
41401	Adam's Home Repair	121 Golden Circle	Chicago	IL	
41402	Jane's Radio/TV	135 Ransom Drive	St Paul	MN	
41403	Advanced Electronics	809 8th Street	St Paul	MN	
41404	Riteway Repair	443 Western Lane	New York	NY	
41405	Fixtures, Inc.	607 9th Avenue	Chicago	IL	+

サブファイル 1
SFLSIZ>
SFLPAG

NUMBER	ORDER NUMBER	LINE NUMBER	DESCRIPTION	QTY	Price	TOTAL PRICE
41401	35900	1	E35 Motor	10	15.00	150.00
41401	35400	2	F60 Pump	25	20.00	500.00

サブファイル 2
SFLSIZ>
SFLPAG

カーソル

RV2W051-0

以下の例では、サブファイル・サイズがページ・サイズと等しく、ROLLUP/ROLLEDOWN キーワードが有効なサブファイルが、最初にディスプレイに書き込まれます。次に、サブファイルがなく、有効な ROLLUP/ROLLEDOWN キーワードがないレコードがディスプレイに書き込まれます。サブファイル・レコードか 2 番目のレコードにカーソルが置かれている場合は、制御はプログラムに戻されます。

カーソル

NUMBER	NAME	ADDRESS	CITY	STATE	
41394	Sorensen and Walton	500 5th Avenue	New York	NY	
41395	Charland, Inc.	200 Madison Avenue	New York	NY	
41316	Anderson's Electric	950 2nd Avenue	Atlanta	GA	
41397	Morem Motors	1300 Pine Street	Atlanta	GA	
41398	Polk Electronics	240 Walters Place	Chicago	IL	+

サブファイル
SFLSIZ>
SFLPAG

Enter next customer number: _____

サブファイル
以外のレコード
ROLLUP/
ROLLEDOWN
なし

RV2W052-0

以下の例では、1 つ目のサブファイルは、サブファイル・サイズがページ・サイズより大きく、ROLLUP/ROLLEDOWN キーワードが指定されていません。2 つ目のサブファイルは、サブファイル・サイズがページ・サイズと等しく、ROLLUP/ROLLEDOWN キーワードが指定されていません。2 つ目のサブファイルにカーソルが置かれている場合は、1 つ目のサブファイルがロールされます。この位置の場合、カーソルはロール可能域にありません。したがって、システムはディスプレイ上の最上部のロール可能域を検索して、ロール機能を実行します。

NUMBER	NAME	ADDRESS	CITY	STATE
41401	Adam's Home Repair	121 Golden Circle	Chicago	IL
41402	Jane's Radio/TV	135 Ransom Drive	St Paul	MN
41403	Advanced Electronics	809 8th Street	St Paul	MN
41404	Riteway Repair	443 Western Lane	New York	NY
41405	Fixtures, Inc.	607 9th Avenue	Chicago	IL +

NUMBER	ORDER NUMBER	LINE NUMBER	DESCRIPTION	QTY	Price	TOTAL PRICE
41401	35900	1	E35 Motor	10	15.00	150.00
41401	35400	2	F60 Pump	25	20.00	500.00

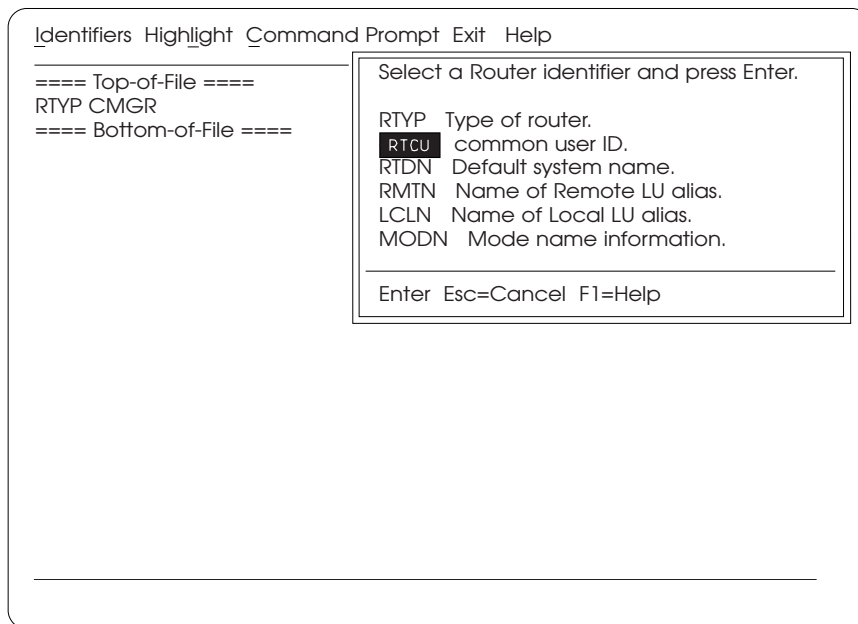
サブファイル
SFLSIZ>
SFLPAG
かつ
ROLLUP/
ROLLEDOWN
なし

サブファイル
SFLSIZ=
SFLPAG
かつ
ROLLUP/
ROLLEDOWN
なし

カーソル

RV2W053-1

以下の例では、サブファイル・サイズがページ・サイズより大きく、ROLLUP/ROLLEDOWN キーワードがないサブファイルが、最初にディスプレイに書き込まれます。次に、サブファイルがなく、ROLLUP/ROLLEDOWN キーワードが有効なレコードが、ディスプレイのサブファイルの上に書き込まれます。カーソルがサブファイル・レコード中に置かれている場合は、サブファイルがロールされます。カーソルがサブファイル・レコード中に置かれていない場合は、サブファイルがロールされず、制御がプログラムに戻ります。



RSLH054-0

サブファイル DDS およびプログラム論理の例についての理解

前述の顧客名検索サブファイルについて記述した DDS を以下に示します。DDS の後に、プログラムがこのサブファイルを処理するのに使用する論理の説明があります。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A** DISPLAY CUS220D  CUSTOMER NAME SEARCH
A                                     CF03(99 'End of Program')
A          R NAMESR                    OVERLAY
A                                     1 29'CUSTOMER NAME SEARCH'
A                                     3  2'Search code'
A          SEARCH                      5  I  3 15
A          R SUBFIL1                    SFL
A          CUST                        5    7  2
A          NAME                        20   B  7  9
A          ADDR                        20   B  7 31
A          CITY                        20   B  7 53
A          STATE                       2    B  7 75
A          R FILCTL                     SFLCTL(SUBFIL1)
A 55                                     SFLDSPCTL
A 50                                     SFLDSP
A                                     SFLSIZ(18)
A                                     SFLPAG(6)
A 50                                     SFLEND
A 60                                     SFLCLR
A                                     OVERLAY PROTECT
A          RCDNBR                      2  0H  SFLRCDNBR(CURSOR)
A 45                                     5  2'NUMBER'
A 45                                     5  9'NAME'
A 45                                     5 31'ADDRESS'
A 45                                     5 53'CITY'
A 45                                     5 75'STATE'
```

図 31. 顧客名検索サブファイルを示すサンプル DDS

ユーザー・プログラムが上記のサブファイル进行处理するのに使用する論理の例を以下に示します。書き込み・読み取り操作とは、入出力操作を結合したものです。読み取りとは入力操作のことです。書き込みとは出力操作のことです。高水準言語プログラムで実行できる操作については、該当する高水準言語の資料を参照してください。

ユーザー・プログラム

1. ファイルを開き、NAMESR レコード様式に対する書き込み・読み取り操作を発行して、検索コードに関するプロンプトを出します。

ユーザー

2. 検索コード・フィールドに郵便番号を入力します。プログラムは検索コード・フィールドをキー・フィールドとして使用し、このキー・フィールドのあるファイル中の最初のデータベース・レコードを検索します。プログラムはこのレコードをサブファイル中の最初のレコードとして使用し、サブファイルを構築します。

ユーザー・プログラム

3. データベース・ファイルからレコードを入手して (読み取り操作)、サブファイルがいっぱいになるか、サブファイルに入れるレコードがなくなるまで、一度に 1 つずつレコードをサブファイルに挿入します (SUBFIL1 への書き込み操作)。
4. すべてのレコードをサブファイルに入れたら、以下を指定してサブファイル制御レコード様式 (FILCTL) に対する書き込み・読み取り操作を発行します。
 - a. レコードが多くて 1 つのサブファイル・ページに収まらない場合、画面の右下隅に + (正符号) が表示されます。SFLEND キーワードに関する標識がオンなので、最後のサブファイル・ページが表示される際にシステムは + をブランクに置き換えます。
 - b. SFLDSP キーワードに関する標識がオンなので、最初のサブファイル・ページが表示されます。
 - c. SFLDSPCTL キーワードに関する標識がオンなので、サブファイル制御レコードが表示されます。
 - d. フィールドに関する SFLRCDNBR(CURS) キーワードを指定します。プログラムはこのサブファイルに値 1 を挿入しているため、相対レコード番号 1 を含むサブファイル・ページが最初に表示され、このレコードの最初の入力フィールドにカーソルが置かれます。(入力フィールドがない場合は、カーソルはこのレコード中の最初に選択された出力フィールドまたは定数に置かれます。)
 - e. 定数フィールド (見出し行) 標識がオンなので、サブファイル制御レコード中の定数が表示されます。
 - f. サブファイル制御レコード中で OVERLAY キーワードと PROTECT キーワードが有効なので、ディスプレイ上のプロンプト (NAMESR) が未変更のままになります。

ユーザー

5. 必要に応じて、ロール・キーを使用してさまざまなサブファイル・レコードを表示して、表示されているレコードを更新します。サブファイルに対するすべての更新を完了したら、Enter キーを押します。

ユーザー・プログラム

6. サブファイル制御レコード様式に対する書き込み・読み取り操作の入力部分を完了します。この例では、サブファイル制御レコードには入力フィールドは含まれていません。ディスプレイ装置のユーザーは、書き込み・読み取り操作の入力部分を使用して、データをサブファイルに入力できます。
7. サブファイル・レコードに対する次変更読み取り操作を発行して、ユーザーが変更を加えた最初のサブファイル・レコード进行处理します。
8. 個々の変更済みレコードを使用して、対応するデータベース・ファイル・レコードを更新します。

9. ステップ 7 で変更済みレコードなし条件が存在するようになるまで、ステップ 7 と 8 を繰り返します。この条件が検出されると、ステップ 10 が実行されます。
10. プロンプト (NAMESR) に対する書き込み・読み取り操作を発行し、プログラムが終了するか、それともデータベース・レコードの別のグループを表示するかを決めます。OVERLAY キーワードが指定されているので、サブファイルの現行の表示内容は未変更のままになります。ユーザーが CF1 キーを押すと、応答標識 99 がオンになり、プログラムはディスプレイ・ファイルを閉じて終了します。ユーザーが、別のレコードのグループが表示されるように別の検索コードを入力すると、ステップ 11 が実行されます。
11. 以下を指定して、サブファイル制御レコード (FILCTL) に対する書き込み操作を発行します。
 - a. SFLCLR キーワードに関する標識がオンなので、すべてのレコードのサブファイルが消去されます (ディスプレイは未変更)。
 - b. SFLDSP キーワードに関する標識がオフなので、ディスプレイの内容は未変更のままになります。
 - c. SFLDSPCTL キーワードに関する標識がオフなので、サブファイル制御レコードは再び表示されません。
12. ステップ 3 から 10 までを繰り返します。ステップ 11 でサブファイルが消去された (SFLCLR キーワード) ので、新しいレコードをサブファイルに挿入できます。ステップ 4 のサブファイル制御レコードに対する書き込み操作で定数フィールド標識がオフなので、ヘッディング情報はディスプレイに再び送信されません。ディスプレイにサブファイル制御レコードが残っている (OVERLAY キーワードが有効でない書き込み操作の介入が実行されていない) 限り、フィールドをディスプレイに再び送信する必要はありません。

第 5 章 ディスプレイ・ファイルを使用したウィンドウの定義

この章では、特殊な DDS ウィンドウ・キーワードを使用して、アプリケーション中でウィンドウを作成する方法について説明します。DDS ウィンドウ・キーワードは、さまざまな目的でウィンドウを作成するための、最も単純で最も柔軟な方式を提供します。例えば、サブファイルを使用してウィンドウにデータを表示したり、システムが基礎的なディスプレイを自動的に保管して復元するようにしたり、フルスクリーン上の位置の代わりにウィンドウ自体の中の位置を参照してウィンドウ内のデータを配置したりできます。

サブファイル、表示属性、妥当性検査、オプション指定などの他の DDS 機能をウィンドウで使用しなければならない場合は、DDS ウィンドウ・キーワードを使用してください。また、ウィンドウに複数の入力フィールドが含まれている場合や、ウィンドウの場所が変わることがある場合も、DDS ウィンドウ・キーワードを使用してください。

DDS を使用してヘルプ情報を定義しており、HLPRCD キーワードを使用してこの情報を表示している場合、WINDOW キーワードを使用してウィンドウにこの情報を簡単に表示できます。

ウィンドウに関して使用する用語、ウィンドウ・キーワードの機能、および以下のタスクでキーワードを使用する方法について、以下に説明します。

- ウィンドウを作成する
- ウィンドウ枠を定義する
- ウィンドウからデータを読み取る
- ウィンドウ枠と内容に変更を加える
- ウィンドウの移動や複写を行う
- 一度に 2 つのウィンドウを活動状態のように表示する
- 一連のウィンドウの中の 1 つを目立たせる
- ウィンドウを除去する
- システムの保管と復元の操作を迂回してアプリケーションのパフォーマンスを改善する

ウィンドウ・キーワードの使用例については、141 ページの『プログラミングの例』を参照してください。

アプリケーションによっては、別の方式を使用してウィンドウを作成することもできます。

- 437 ページの『第 20 章 オンライン・ヘルプ情報の定義』には、ユーザー・インターフェース管理機能 (UIM) を使用してヘルプ・ウィンドウを作成する方法について説明されています。UIM は DDS と異なる言語を使用します。しかし、UIM は多数のヘルプ・ウィンドウ機能を自動化し、既存のアプリケーションにオンライン・ヘルプ情報を追加する最も単純な方法です。
- 「*IBM WindowTool/400 PRPQ*」(SC41-0050) には、WindowTool/400 PRPQ を使用してウィンドウを作成する方法が説明されています。主にウィンドウを使用してアプリケーション・メニューを構成する場合は、このプログラムを使用することを考慮してください。
- 『第 6 章 画面用のグラフィカルな表示の作成』には、メニュー・バー、プルダウン・メニュー、選択フィールド、継続項目フィールドを作成する方法、および DDS キーワードを使用して編集マスクを使用する方法について説明されています。

ウィンドウの用語

ウィンドウとは、ディスプレイの一部をオーバーレイする情報のことです。ユーザーはウィンドウ内、およびウィンドウによってオーバーレイされていないディスプレイの部分に、情報を表示できます。しかし、活動状態なのはウィンドウのみです。ユーザーは基礎的なディスプレイを処理できません。複数のウィンドウが表示される場合、一度に 1 つのウィンドウのみ活動状態になります。

活動状態のウィンドウとは、最新の入力または出力操作の対象のウィンドウのことです。活動状態のウィンドウは、ディスプレイ上の最上位ウィンドウとして表示されます。ワークステーションのユーザーが対話できるディスプレイの部分は、活動状態のウィンドウのみです。

アプリケーションまたはシステムがウィンドウを除去するアクションを取るまで、ウィンドウはディスプレイ上に残ります。ウィンドウの除去とウィンドウのオーバーレイは別個の操作です。ウィンドウが除去されると、ディスプレイ上に存在しなくなり、ウィンドウへの書き込みやウィンドウからの読み取りは行えなくなります。あるウィンドウが別のウィンドウによって**オーバーレイ**されると、ワークステーションのユーザーには見えなくなります。しかし、引き続きこのウィンドウを使用して処理できます。

DDS ウィンドウ・キーワード

4 つの DDS キーワードを使用すると、アプリケーションでウィンドウを作成したり処理したりできます。

WINDOW (ウィンドウ)

ディスプレイ上にウィンドウを作成するか、既存のウィンドウの内容に変更を加えるか、または既存の非活動状態のウィンドウを再び活動状態にします。

WDWBORDER (ウィンドウ枠)

ウィンドウ枠の色、表示属性、および文字を指定します。

WDWTITLE (ウィンドウ・タイトル)

ウィンドウのタイトルのテキスト、色、および表示属性を指定します。タイトルはウィンドウの上部または下部の枠に組み込まれます。

注: ウィンドウの下部の枠のテキストや、上部または下部の枠のテキストの左寄せおよび右寄せをサポートしていないコントローラーもあります。

RMVWDW (ウィンドウの除去)

新しいウィンドウが表示される際、または既存のウィンドウが活動状態のウィンドウとして再表示される際に、ディスプレイからその他のウィンドウを除去します。

USRRSTDSP (ユーザー復元画面)

ウィンドウが表示されて除去される際に、i5/OS システムが基礎的なディスプレイを保管して復元しないようにします。保管および復元操作が必要ない状態もあります。これらの操作を迂回すると、アプリケーションの速度が上がります。USRRSTDSP キーワードを使用して、シリーズ中の前のウィンドウがポップアップして後のウィンドウをオーバーレイし、2 つのウィンドウが同時に活動状態になっているように見えるようにすることもできます。

個々のキーワードに関する詳細な参照情報については、iSeries Information Center の『**DDS**』のトピックを参照してください。

ウィンドウの表示内容とハードウェア構成

使用しているディスプレイ装置とワークステーション・コントローラーのタイプに応じて、ウィンドウ枠の表示は異なります。

ハードウェア構成の違いによるウィンドウの表示内容の違いについては、151 ページの『ハードウェア構成』を参照してください。

ウィンドウの作成

レコード・レベルのキーワードである `WINDOW` を使用すると、レコード様式をウィンドウ内に表示できます。一度に最大 12 のウィンドウをディスプレイ上に作成できます。

ウィンドウを作成するには、`WINDOW` キーワードを指定するレコードを書き込んでください。ディスプレイ上のウィンドウ・サイズとその場所を指定するウィンドウ定義レコードを、最初のウィンドウ・レコードとして書き込まなければなりません。ウィンドウ定義レコードは、ディスプレイ上にウィンドウ枠を挿入します。その後、同じウィンドウ定義レコードを再び書き込むか、または 1 つ以上のウィンドウ参照レコードを使用してウィンドウの指定を完成できます。

ウィンドウ定義レコード

ウィンドウ定義レコードとは、ウィンドウ・サイズおよび場所を定義する、`WINDOW` キーワードを含むレコードのことです。サイズと場所の属性には、ウィンドウ枠の左上隅の位置と、ウィンドウ内の行と列の数が含まれます。

表示するウィンドウごとに、最初のレコードとしてウィンドウ定義レコードを書き込まなければなりません。ウィンドウ定義レコードとは、実際にウィンドウを作成してディスプレイ上に表示するレコードのことです。このレコードには、標準的なレコード中にあるフィールドやデータのタイプを含めることができます。また、ウィンドウ枠を定義する `WDWBORDER` キーワードを含めることもできます。(WDWBORDER キーワードはファイル・レベルでも使用できます。)

ウィンドウ定義レコードで、ウィンドウの仕様をすべて提供できます。ウィンドウにエラー・メッセージを表示する際に行うのと同様に、同じウィンドウ定義レコードを再び書き込むと、追加の仕様を提供できません。ウィンドウ定義レコードを書き込んだ後で、1 つ以上のウィンドウ参照レコードを書き込んでも、追加の仕様を提供できます。

ウィンドウ参照レコード

ウィンドウ参照レコードは、ウィンドウに挿入する追加データを提供します。ウィンドウ参照レコードを使用すると、1 つのウィンドウに複数のレコード様式を表示できます。

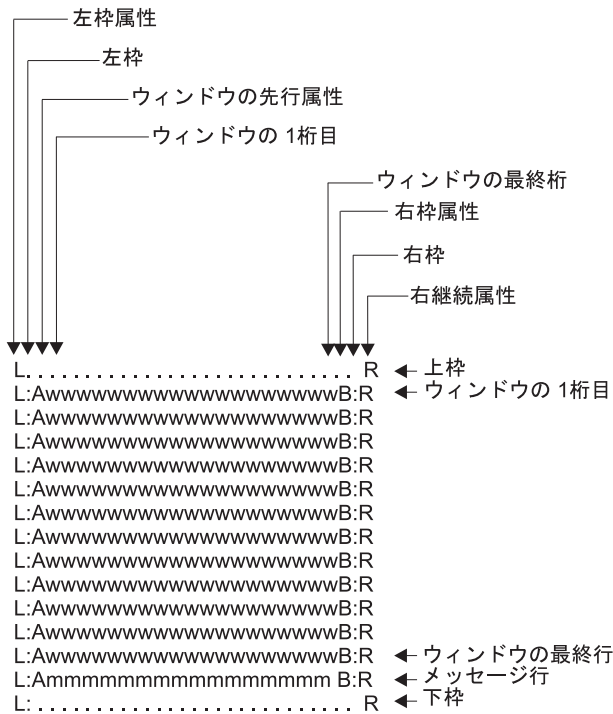
個々のウィンドウ参照レコードには、このレコードを適用するウィンドウ定義レコードの名前を指定する `WINDOW` キーワードが含まれます。ウィンドウ参照レコードを書き込む際には、参照されるウィンドウ定義レコードがディスプレイ上になければなりません。参照されるレコードがディスプレイ上にないと、ウィンドウが存在しないことを示す通知メッセージがアプリケーションに戻されます。

ウィンドウ参照レコードにはサイズと位置の属性は含まれず、活動状態の `WDWBORDER` キーワードは無視されます。

ウィンドウを完成させるのに必要な数だけウィンドウ参照レコードを使用できます。しかし、ウィンドウを表示するのに、ウィンドウ参照レコードを書き込む必要はありません。ウィンドウ定義レコードのみを使用してウィンドウを表示できます。

ウィンドウ・サイズおよび場所

以下の図は、ウィンドウ定義レコードの表示時に作成されるウィンドウの一部を示しています。



記号表:

- L 左枠属性
- : 上下枠
- R 右継続属性
- : 左右枠
- A ウィンドウの先行属性
- W ウィンドウ域
- m メッセージ行
- B 右枠属性

RBAHG501-0

このウィンドウは、深さが 13 行 (row) (行 (line) とも呼ばれる)、幅が 19 列 (位置とも呼ばれる) として指定されています。ウィンドウ枠内の使用可能域は深さ 12 行、幅 19 列です。行 13 はメッセージ用に予約されています。この行にフィールドを含めることはできません。

実際にはウィンドウはウィンドウ枠を使用するので、ディスプレイ上に追加の行と列があります。すべてのウィンドウには追加の 2 行 (上部の枠と下部の枠用に 1 つずつ) があります。また、ほとんどのウィンドウには追加の 6 列があります。

- 枠属性用に 2 つ
- 枠文字用に 2 つ
- 先行するウィンドウ属性用に 1 つ
- 右側の続きの属性用に 1 つ

ウィンドウの先頭と末尾がディスプレイ全体の最初の列と最後の列の場合、追加の列は 6 列ではなく 4 列のみになります。ウィンドウの行が DBCS フィールドをオーバーレイする場合、この行には前述の 6 列に加えて、個々の側部の DBCS シフトアウト文字とシフトイン文字用にさらに最大 4 列必要です。

ウィンドウは枠の左上隅に沿って配置され、開始行の位置は上部の枠の行になり、開始列の位置は左側の枠の列になります。ウィンドウに収まらないフィールドは、ファイルのコンパイル時に診断されます。SLNO キーワードを使用し、開始行の番号を指定している場合は、DDS コンパイラーはウィンドウに収まらないフィールドにフラグを立てます。SLNO キーワードに *VAR を指定している場合に、実行時の開始行の番号のためにレコード全体が表示されない際には、例外が発生します。

定数として左上のウィンドウ行または左上のウィンドウ位置が指定されている際には、DDS コンパイラーはウィンドウの場所の問題を診断します。動的に配置されるウィンドウがディスプレイに収まらない場合は、実行時エラーが生じます。

特殊値 *DFT を、ウィンドウ・キーワードの開始行パラメーターと開始位置パラメーターの代わりに指定できます。*DFT は、システムがウィンドウの開始行と開始位置を決めることを示します。ウィンドウはカーソル位置と比較して配置されます。*DFT の使用時には、システムは以下の規則の手順を使用して、ウィンドウを配置します。

1. ウィンドウがディスプレイ上のカーソル位置の下に収まる場合は、その場所に置かれます。上部のウィンドウ枠は、カーソルの下の 1 行目に配置されます。可能な場合、左側のウィンドウ枠はカーソルと同じ列に配置されます。可能でない場合、ウィンドウはカーソルの左側に、ディスプレイに収まるのに必要なだけ離して配置されます。
2. ウィンドウがカーソル位置の上に収まる場合は、その場所に置かれます。下部のウィンドウ枠は、カーソルの上の 1 行目に配置されます。ウィンドウは、ステップ 1 の説明どおりに横に配置されます。
3. ウィンドウがカーソル位置の右に収まる場合は、その場所に置かれます。右側のウィンドウ枠は、ディスプレイの最後から 2 列目に配置されます。可能な場合、上部のウィンドウ枠はカーソルと同じ行に配置されます。可能でない場合、ウィンドウはカーソルの上部に、ディスプレイに収まるのに必要なだけ離して配置されます。
4. ウィンドウがカーソル位置の左に収まる場合は、その場所に置かれます。右側のウィンドウ枠は、カーソルの左側の 2 列に配置されます。ウィンドウは、ステップ 3 の説明どおりに縦に配置されます。
5. 上記のどの領域にもウィンドウを配置できない場合は、ディスプレイの右下隅に置かれます。

カーソル位置

ウィンドウ内にカーソルを配置するには、CSRLOC および DSPATR(PC) キーワードを、フルスクリーン・ディスプレイの場合と同じ方法で使用します。カーソルは、ウィンドウの使用可能域の左上隅に参照と共に配置されます。

WINDOW キーワードに *RSTCSR を指定した場合、カーソルが活動状態のウィンドウの使用可能域の外側に移動されると、Print および Home コマンド機能 (CF) キーのみ活動状態になります。ワークステーションのユーザーがその他のコマンド機能 (CF) キーを押すと、アラームが鳴り、カーソルが直前の書き込み操作の位置に移動して戻ります。*RSTCSR はデフォルトです。

注: 非プログラマブル・ワークステーションに関する拡張インターフェースをサポートするコントローラーに接続しているディスプレイ装置では、*RSTCSR を指定すると、マウスを使用する場合に限りカーソルをウィンドウの外部に移動できます。

WINDOW キーワードに *NORSTCSR を指定すると、ユーザーはカーソルを活動状態のウィンドウの外側にカーソルを移動したり、コマンド機能 (CF) またはコマンド・アテンション (CA) キーを使用したりできません。

エラー・メッセージ

ディスプレイ上にウィンドウがある場合に、ウィンドウに関する WINDOW キーワードに *MSGLINが指定されていると、活動状態のウィンドウの最後の使用可能な行にエラー・メッセージが表示されます。ウィンドウ内の最後の使用可能な行は、エラー・メッセージ用に予約されています。この行にはレコードは表示されません。エラー・メッセージが行より長い場合は、収まるように切り捨てられます。ディスプレイ上にウィンドウがある場合に、ウィンドウに関する WINDOW キーワードに *NOMSGLIN を指定すると、ディスプレイの下部か、MSGLOC キーワードで定義された場所にエラー・メッセージが表示されます。

ヘルプ・キーを使用して、ウィンドウ内で表示されるエラー・メッセージに関するヘルプを使用できます。

操作エラーやキーボード・エラーを報告するメッセージ（「機能キーは使用できません。」など）が表示されると、キーボードはロックされるので、続行するためにはユーザーはエラー・リセット・キーを押さなければなりません。

以下の状況では、不一致のシフトアウトおよびシフトイン文字がディスプレイに送信されたことを示す通知メッセージがジョブ・ログに入れます。

- 基本ディスプレイに、複数の行にわたる DBCS フィールドがある。
- ウィンドウが表示され、基本ディスプレイの DBCS フィールドの一部がウィンドウのメッセージ行にある。
- ファンクション・キーを押した結果、操作エラーまたはキーボード・エラーになる。

ウィンドウに関して使用する場合、2、3 のメッセージ関連キーワードの機能に違いが生じます。

- ERRSFL キーワードは無視される。この機能は、ディスプレイ上にウィンドウがない場合のみ実行されます。
- WINDOW キーワードに *MSGLIN が指定されている場合、MSGLOC キーワードは無視される。この機能は、ディスプレイ上にウィンドウがないか、WINDOW キーワードに *NOMSGLIN が指定されている場合のみ実行されます。
- ERRMSGID、ERRMSG、SFLMSG、SFLMSGID、および DDS 妥当性検査キーワードからの結果のメッセージはウィンドウに表示されるが、キーボードはロックされない。この種のメッセージがフルスクリーン・ディスプレイに表示される場合は、キーボードはロックされます。

サブファイル

一度に最大 24 のサブファイルを活動状態にすることができます。一度に最大 12 のサブファイルを基本ディスプレイまたは単一のウィンドウに表示できます。

サブファイルがウィンドウに表示されている場合、ウィンドウがディスプレイから除去されても、サブファイルは削除されません。ディスプレイ・ファイルが閉じるか、明示的にサブファイルを削除するまで、サブファイルは活動状態のままになります。

DDS ヘルプ・レコード

ウィンドウ定義レコードが DDS ヘルプ・レコードとしてディスプレイに書き込まれると、現行のディスプレイは通常のアプリケーション・ヘルプの場合と同様に中断され、アプリケーション・ヘルプ・レコードがディスプレイにウィンドウとして書き込まれます。ウィンドウ・レコードをヘルプ・レコードとして使用する際にエラーが発生しないようにするには、以下の要件に従ってください。

- ヘルプ・レコードを含むディスプレイ・ファイル中に ASSUME キーワードを組み込む。ASSUME キーワードがない場合、ヘルプ・ウィンドウが表示されると、ディスプレイの残りの部分はブランクになります。

- ディスプレイ・ファイルの作成、変更、またはオーバーライドの際に、ディスプレイ復元 (RSTDSP) パラメーターに *YES の値を使用する。ヘルプから戻る際に、RSTDSP(*YES) は中断状態のディスプレイを復元し、ヘルプ・キーを押した時点の位置にカーソルを戻します。
- ウィンドウ定義レコードで変数の行と位置の値を使用して、オペレーティング・システムがカーソル位置に従ってヘルプ・ウィンドウを動的に配置できるようにする。システムは、ウィンドウ・キーワードに *DFT が指定されている場合と同じ規則の手順を使用して、ウィンドウを配置します。これらの規則については、130 ページの『ウィンドウ・サイズおよび場所』で説明されています。ウィンドウ定義レコード中で行の値のみまたは位置の値のみを変数として指定すると、同じ規則に従います。しかし、定数値は変更されません。

ウィンドウ枠の定義

ウィンドウ枠のシステム・デフォルトを使用するか、または WDWBORDER キーワードを使用してウィンドウ枠を定義できます。このキーワードは、色、表示属性、および文字の 3 つの枠のコンポーネントを指定します。複数の WDWBORDER キーワードを指定できます。WDWBORDER キーワードとオプション標識を併用できます。

WDWBORDER キーワードはファイル・レベルで使用する (ファイル中の個々のウィンドウ定義レコードに適用される) か、または個々のウィンドウ定義レコードで使用できます。ウィンドウ参照レコードで使用する場合は、ファイルの作成時に警告メッセージが発行されます。

以下に、ウィンドウ枠のデフォルト、システムが複数のウィンドウ枠定義を処理する方法、および WDWBORDER キーワードを使用して UIM ヘルプ・ウィンドウ枠を定義する方法について説明します。

枠のデフォルト

WDWBORDER キーワードを使用しない場合のシステム・デフォルトは以下のとおりです。

枠の要素	デフォルト
色	カラー・ディスプレイ上では青。カラーではないディスプレイ装置上では、この属性は無視されます。
表示属性	通常 (つまり、強調表示や反転表示などの属性なし)
上部および下部の枠の文字	ピリオド (.)
左側および右側の枠の文字	コロンの (:)
左上隅および右上隅の文字	ピリオド (.)
左下隅および右下隅の文字	コロンの (:)

注: 拡張インターフェースをサポートするコントローラーに接続されている RUMBA/400 ワークステーションおよび InfoWindow® II ディスプレイ装置は、ウィンドウ枠は実線です。詳細については、151 ページの表 15 を参照してください。

複数の枠定義

ファイル・レベルでウィンドウ定義レコード中に WDWBORDER キーワードを指定すると、ファイル・レベルとレコード・レベルのキーワードのパラメーター値は結合されます。パラメーター値が対立する場合、レコード・レベルのパラメーター値が使用されます。例えば、ファイル・レベルで以下のように指定されているとします。

```
WDWBORDER((*COLOR RED) (*DSPATR RI))
```

また、レコード・レベルで以下のように指定されているとします。

```
WINDOW(2 5 10 20) +  
WDWBORDER((*COLOR GRN) +  
          (*CHAR '.....'))
```

この場合、ウィンドウ枠は反転表示の緑色のピリオドで構成されます。

同じレベルで複数の `WDWBORDER` キーワードを指定すると、有効なキーワードのパラメーターが結合されます。同じパラメーターに別の値を指定すると、有効な最初のキーワードのパラメーター値が使用されます。枠の個々のコンポーネントの値は、ウィンドウ定義レコードが書き込まれる際に決められます。以下の階層によって、枠の値が決められます。

1. システム・デフォルトで開始します。
2. デフォルトを、ファイル・レベルの枠の指定でオーバーライドします。
3. ファイル・レベルの枠の指定を、レコード・レベルの枠の指定でオーバーライドします。

同じレベルの複数の `WDWBORDER` キーワードが有効で、キーワードに同じ `WDWBORDER` コンポーネント (色、属性、または文字) が指定されている場合、プロセスは似ています。最初のコンポーネント値が使用されます。例えば、ファイル・レベルで以下の 2 つの `WDWBORDER` キーワードが有効であるとしてします。

```
WDWBORDER((*COLOR GRN  
          *CHAR '.....'))  
WDWBORDER((*CHAR '---|||-|'))
```

また、レコード・レベルで以下のように指定されているとします。

```
WDWBORDER((*COLOR BLU))
```

この場合、枠のコンポーネントの値は以下のように決められます。

1. コンポーネントごとのデフォルトで開始します。
2. 文字のコンポーネントとカラー要素のデフォルトを、ファイル・レベルの値でオーバーライドします。文字のコンポーネントが複数回指定されているので、最初に指定されている文字のコンポーネントの値が使用されます。
3. ファイル・レベルの色の値を、レコード・レベルの値でオーバーライドします。

ウィンドウ枠は、レコード・レベルの枠の色、最初のファイル・レベルの枠の文字、およびデフォルトの枠の表示属性で構成されます。

単一の `WDWBORDER` キーワードで 3 つの枠のコンポーネントの一部が指定されていない場合は、指定されていないコンポーネントは有効なその他の `WDWBORDER` キーワードからの値を使用します。デフォルトは使用しません。前述の例では、このキーワードはレコード・レベルで例示されています。カラー要素のみ指定されています。しかし、ファイル・レベルで文字のコンポーネントが指定されているので、デフォルトの代わりにファイル・レベルの値が使用されます。表示属性のコンポーネントはレコード・レベルでもファイル・レベルでも指定されていないので、デフォルトが使用されます。

UIM ヘルプ・ウィンドウ枠

`WDWBORDER` キーワードを使用して、UIM パネル・グループを使用して作成されたヘルプ・ウィンドウの枠属性を指定できます。フルスクリーン・ディスプレイには `DDS` を使用し、ヘルプ・ウィンドウには UIM を使用しているとします。ヘルプ・キーを押した時点でディスプレイ上にウィンドウがある場合は、UIM ヘルプ・ウィンドウには `DDS` ウィンドウと同じ枠があります。活動状態のウィンドウがなく、ファイル・レベルまたは現在ディスプレイ上にある非ウィンドウ・レコードのレコード・レベルで

WDWBORDER キーワードが指定されている場合、システムはファイル・レベル、レコード・レベル、およびデフォルト値を結合して UIM ヘルプ・ウィンドウの枠属性および文字値を決めます。現在表示されているウィンドウがないとします。まず、WDWBORDER キーワードが指定されている非ウィンドウ・レコードがディスプレイに書き込まれます。次に、UIM ヘルプ・ウィンドウがディスプレイに書き込まれます。UIM ヘルプ・ウィンドウ枠は、非ウィンドウ・レコード上の WDWBORDER キーワードで指定されている属性を使用します。

ウィンドウ・タイトルの定義

ウィンドウ・タイトル (WDWTITLE) キーワードを使用して、ウィンドウのタイトルのテキスト、色、および表示属性を指定します。タイトルはウィンドウの上部または下部の枠に組み込まれます。タイトルの長さの最大値は、関連したウィンドウ定義キーワードで指定されているウィンドウ位置パラメーターに指定されている位置の数です。

注: ウィンドウの下部枠のテキストをサポートしていないコントローラーもあります。

WDWTITLE キーワードは、(定義様式中の) WINDOW キーワードを含むレコード上で指定しなければなりません。別のウィンドウを参照する WINDOW キーワードも指定すると、警告メッセージが発行されます。

図 32 には、ウィンドウ・タイトルの例が示されています。

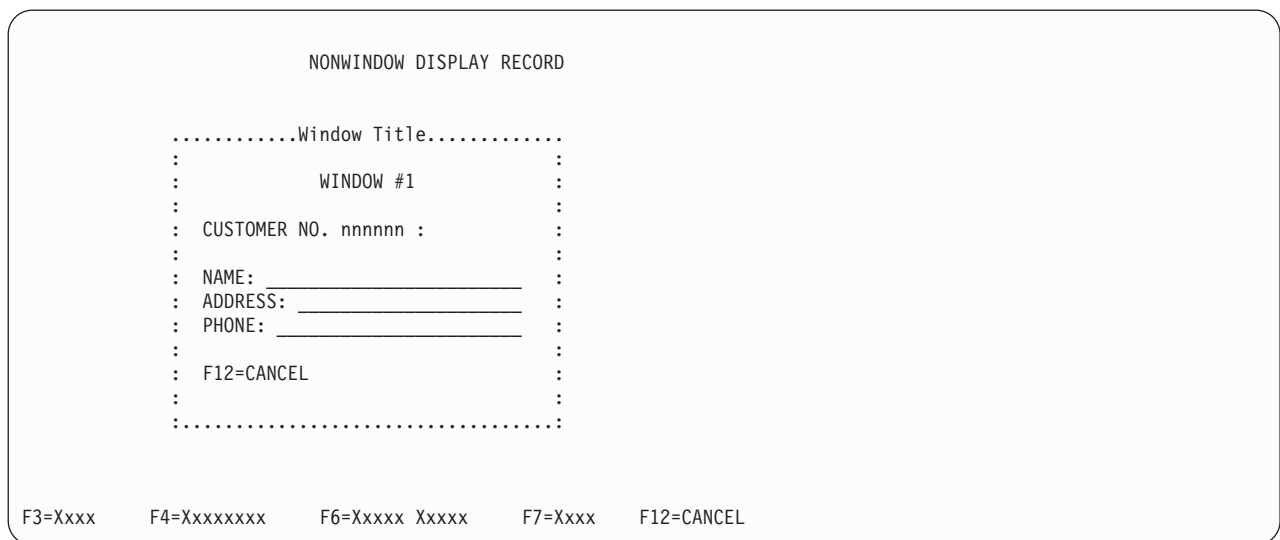


図 32. ウィンドウ・タイトルのディスプレイの例

ウィンドウ・タイトル例の DDS

```
|...+...1...+...2...+...3...+...4...+...5...+...6...+...7...
A          R RECORD1                WINDOW(6 15 18 51)
A N01     WDWTTITLE((*TEXT &TTL1) (*COLOR GRN))
A 01     WDWTTITLE((*TEXT &TTL1) (*COLOR RED))
A          FIELD1          5A B 2 2
A          FIELD2          20A B 8 5
A          TTL1            10A 0P
A*
A          R RECORD2                WINDOW(8 20 9 30)
A          WDWTTITLE((*TEXT &TTL2) +
A                      (*COLOR YLW) +
A                      (*DSPATR RI))
A          FIELD3          5A B 2 2
A          FIELD4          20A B 8 5
A          TTL2            10A 0P
```

図 33. ウィンドウ・タイトルの DDS

図 33 の RECORD1 で定義されているウィンドウがディスプレイに書き込まれる場合、タイトルは TTL1 フィールド中に含まれている任意のテキストになります。タイトルは、ウィンドウの上部の枠の中央に表示されます。標識 01 をオフに設定すると、テキストは緑色になります。標識 01 をオンに設定すると、テキストは赤色になります。

RECORD2 で定義されているウィンドウがディスプレイに書き込まれる場合、タイトルは TTL2 フィールド中に含まれている任意のテキストになります。タイトルは、ウィンドウの上部の枠の中央に表示されます。タイトルは反転表示され黄色になります。

ウィンドウからのデータの読み取り

ディスプレイ上にウィンドウがある場合、活動状態のウィンドウ (つまり、ディスプレイに最後に書き込まれたウィンドウ) だけから入力データを受け取ることができます。アプリケーションがウィンドウ・レコードを読み取る場合、そのウィンドウがディスプレイ上にあるが活動状態のウィンドウではなければ、最新の入力または出力操作の対象のウィンドウが除去されます。読み取られるレコードを含むウィンドウが復元され、活動状態のウィンドウになります。次に、レコードがディスプレイから読み取られます。

アプリケーションがウィンドウ・レコードの読み取りを試行する場合に、ウィンドウがディスプレイ上にないと、ウィンドウが存在しないことを示す通知メッセージが戻されます。アプリケーションがウィンドウ・レコードの読み取りを試行する場合に、ウィンドウはディスプレイ上にあるがレコードがディスプレイ上にないと、レコードがディスプレイ上にないことを示すメッセージが戻されます。

ウィンドウの枠と内容の変更

すでにディスプレイ上にあるウィンドウの内容に変更を加えるには、このウィンドウのウィンドウ定義レコードか、またはこのウィンドウを指定するウィンドウ参照レコードをディスプレイに書き込まなければなりません。すでにディスプレイ上にあるウィンドウの枠に変更を加えるには、そのウィンドウ定義レコードをディスプレイに書き込まなければなりません。

ウィンドウ定義レコードをディスプレイに書き込む場合に、その名前のウィンドウがディスプレイ上の同じ位置にすでにあれば、新しいウィンドウは作成されません。新しいレコードは、既存のウィンドウに対する通常書き込み操作と見なされます。新しいレコードで別の枠属性または文字を指定した場合、そのレコードが書き込まれる際に新しい属性または文字が表示されます。

ウィンドウの移動および複写

ウィンドウ定義レコードをディスプレイに書き込む場合に、その名前のウィンドウがディスプレイ上の別の位置にすでにある場合、新しいウィンドウが作成されます。指定された位置に新しいウィンドウが表示され、名前は既存のウィンドウと同じになります。ウィンドウを移動するには、書き込まれるウィンドウ定義レコード中で **RMVWDW** キーワードを使用します。新しいウィンドウが書き込まれると、この名前の既存のウィンドウと、ディスプレイ上の他のウィンドウは除去されます。その結果、ウィンドウは移動されます。**RMVWDW** キーワードを使用しない場合は、ディスプレイ上の 2 つ別々の位置に同じウィンドウが表示されます。

ウィンドウ参照レコードをディスプレイに書き込む場合に、指定されたウィンドウがディスプレイ上にあると、指定された名前の最後に作成されたウィンドウにレコードが書き込まれます。この名前の以前に作成されたウィンドウに書き込むには、以前のウィンドウの場所を指定したウィンドウ定義レコードを使用します。

一度に 2 つのウィンドウを活動状態のように表示する

正しくは一度に 1 つのウィンドウのみ活動状態にできますが、一度に 2 つのウィンドウが活動状態であるかのようにすることができます。この機能を使用して、2 つのウィンドウを横並びに表示し、ワークステーションのユーザーがそれらのウィンドウの間で相互に切り替えられるようにすることができます。

2 つのウィンドウが活動状態であるかのようにするには、次のようにします。

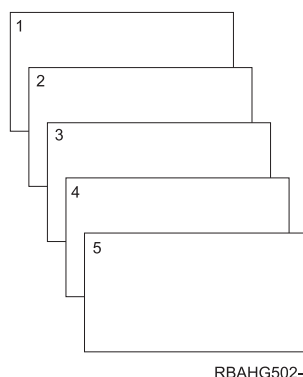
1. ワークステーションのユーザーのために切り替えアクションを実行するように機能キーをセットアップします。例えば、1 つ目のウィンドウ内のデータを使用するページに 1 つのキーを提供し、2 つ目のウィンドウ内のデータを使用するページに別のキーを提供することができます。
2. 1 つ目のウィンドウをディスプレイに書き込みます。
3. **USRRSTDSP** キーワードを使用して 2 つ目のウィンドウをディスプレイに書き込みます。**USRRSTDSP** キーワードは、2 つ目のウィンドウが表示される際に 1 つ目のウィンドウが保管されないようにします。ユーザーが 1 つ目のウィンドウに戻る際に 2 つ目のウィンドウが除去されないようにしてから、ユーザーが 2 つ目のウィンドウに戻る際に 1 つ目のウィンドウが除去されないようにします。

1 つ目のウィンドウで **USRRSTDSP** キーワードを使用することもできます。ただし、2 つ目のウィンドウでこのキーワードを使用すると、1 つ目のウィンドウと後続のすべてのウィンドウが保管されなくなるので、1 つ目のウィンドウにはこのキーワードは必要ありません。

4. ワークステーションのユーザーが切り替えアクションを実行するキーを押すと個々のウィンドウが再ビルドされるよう準備します。2 つのウィンドウが表示されたら、ユーザーが個々のウィンドウに移動する際に必ずそのウィンドウを再ビルドしなければなりません。**USRRSTDSP** キーワードは両方のウィンドウの保管と復元が行われられないようにするので、切り替えられるたびに再ビルドされなければなりません。その結果、以前にディスプレイ上にあったウィンドウはシステムにとって不明になります。

USRRSTDSP キーワードの詳細については、139 ページの『アプリケーションのパフォーマンスの改善』を参照してください。

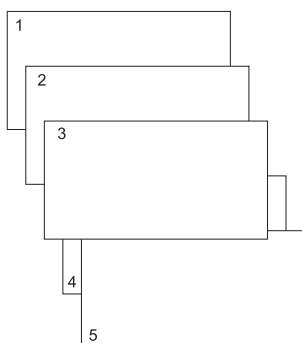
一連のウィンドウの中の 1 つを目立たせる



RBAHG502-0

図のようなウィンドウのシリーズを表示するとします。

ここで、ウィンドウ 3 に戻り、それが以下のディスプレイのように、一連のウィンドウの最上部に飛び出



RBAHG503-0

して見えるようにするとします。

ウィンドウ 4 および 5 をディスプレイから除去せずにウィンドウ 3 を目立たせるには、以下のステップを実行します。

1. ウィンドウ 4 とそれ以前のウィンドウ (つまりウィンドウ 1 から 3 まで) に **USRRSTDSP** キーワードを指定します。USRRSTDSP キーワードは、直前のウィンドウと後続のウィンドウが保管されないようにします。
2. ウィンドウ 4 を最初に表示する前と同様の概観になるようにウィンドウ 3 を再ビルドします。USRRSTDSP を使用したので、ウィンドウ 4 がディスプレイに追加された時点やその後にディスプレイは保管されませんでした。つまり、どのウィンドウもディスプレイから除去されないため、ウィンドウ 3 は一連のウィンドウの最上部に飛び出すように表示されます。

USRRSTDSP キーワードの詳細については、139 ページの『アプリケーションのパフォーマンスの改善』を参照してください。

ウィンドウの除去

ウィンドウが除去されると、ディスプレイ上に存在しなくなり、ウィンドウへの書き込みやウィンドウからの読み取りは行えなくなります。ウィンドウ・キーワードによりウィンドウを除去する方法は数種類あります。使用する方式は、除去するウィンドウと、次に実行する操作に応じて異なります。以下にさまざまな方式が説明されています。

すべてのウィンドウの除去

以下の方法の 1 つで、ディスプレイ上のすべてのウィンドウを除去します。

1. 非ウィンドウ・レコードに書き込みます。こうすると、既存のウィンドウがすべて除去され、新しいウィンドウは表示されません。
2. RMVWDW キーワードを指定するウィンドウ・レコードに書き込みます。RMVWDW キーワードを使用すると、指定されたウィンドウが表示される際に、ディスプレイ上の他のすべてのウィンドウが除去されます。他のウィンドウがない場合は、RMVWDW キーワードは無視され、エラーは戻されません。

新しい方のウィンドウの除去

活動状態にしたいウィンドウをオーバーレイしている、それより新しいウィンドウを除去するには、活動状態にするウィンドウに対する読み取りまたは書き込みを行います。ディスプレイ上にウィンドウ 3 があるが、活動状態のウィンドウでないとして。ウィンドウ 4、5、および 6 はウィンドウ 3 より新しい書き込み操作の対象でした。したがって、これらのウィンドウはウィンドウ 3 をオーバーレイしています。ウィンドウ 4、5、および 6 を除去し、ウィンドウ 3 を見える状態にして、活動状態のウィンドウにするには、ウィンドウ 3 に対する読み取りまたは書き込みを行います。

アプリケーションのパフォーマンスの改善

USRRSTDSP キーワードを使用して、オペレーティング・システムがディスプレイの保管と復元を行えないようにすることにより、アプリケーションのパフォーマンスを改善できる場合があります。システムがこの操作を実行する方法、および USRRSTDSP キーワードを使用する方法と時点について、以下に説明します。

システムの保管および復元の操作

USRRSTDSP キーワードを使用しないと、オペレーティング・システムはアプリケーションに関する保管操作と復元操作を自動的に実行します。ウィンドウが表示される前に、システムは、除去されていないすべてのウィンドウを含むディスプレイを保管します。ウィンドウが除去される際に、システムはディスプレイを復元します。

ディスプレイ上に新しいウィンドウが作成される場合、そのウィンドウの書き込みの際に活動状態になっているレコードは保管され、ディスプレイ全体は背景データとして残ります。次に、新しいウィンドウが活動状態になります。保管されたレコードは、ウィンドウ・レコードまたは非ウィンドウ・レコードになります。その手順は両方とも同じです。

ウィンドウ・レコードがディスプレイに書き込まれ、既存のウィンドウが変更または再表示される場合、ターゲットより新しいウィンドウはすべて保管されずに除去され、ターゲット・ウィンドウは復元され、新しいレコードが書き込まれ、このウィンドウが活動状態になります。

非ウィンドウ・レコードが画面に書き込まれる場合、既存のウィンドウはすべて保管されずに除去され、新しいレコードが初期画面に書き込まれ、ディスプレイが活動状態になります。

応答時間

システムが保管および復元操作を実行するのに必要な時間は、通信のセットアップと、表示されるウィンドウに応じて異なります。

最初のウィンドウがディスプレイに追加される際に実行される読み取りおよび保管操作の間、応答時間は最も遅くなります。平均的なサイズと複雑さのウィンドウを想定してください。ワークステーションが平衡型によってシステムに接続している場合は、ローカル・エリア・ネットワーク (LAN) または高速通信回線の

応答時間は速くなります。ワークステーションが 2400 ボー専用回線によって接続されている場合、読み取りおよび保管操作を完了してからウィンドウを表示するまでに約 10 秒かかります。回線速度を 9600 ボーに上げると、通常は約 2.5 秒かかります。

2 つ目または 3 つ目のウィンドウを追加する前にディスプレイを保管したり、ウィンドウを除去した後にディスプレイを復元したりするなどの他の操作に要する時間はさらに短くなります。

他のウィンドウ・サイズ、端末タイプ、および回線速度に関する詳細を含めて詳しくは、営業担当員にご連絡ください。

システムの保管および復元操作の迂回

USRRSTDSP キーワードを使用して、システムの保管および復元処理を迂回し、その代わりに必要な場合に限りアプリケーションにディスプレイを再ビルドさせることができます。この手法は、システム・パフォーマンスとアプリケーションのユーザーの応答時間を改善できます。一時に 1 つのウィンドウのみ表示し、ウィンドウが別のディスプレイ・ファイル中にある場合か、ウィンドウのシリーズを表示し、ユーザーが前のウィンドウに戻らない場合か、または一度に複数のウィンドウが活動状態に見えるようにする場合には、このキーワードを使用することを考慮してください。

例えば、以下の条件で、システムが正規に 2 つの保管操作を実行するとします。

- アプリケーションは一時に 1 つのウィンドウのみ表示する。
- RSTDSP(*YES) を指定してディスプレイ・ファイルが作成される。
- ディスプレイをオーバーレイする最初のウィンドウ・レコードが、別のファイル中にある。

最初の保管操作は、ディスプレイ・ファイルが中断される際に実行されます。2 番目の保管操作は、ウィンドウを表示するために実行されます。USRRSTDSP は 2 番目の不要な保管操作を除去します。

システムの保管および復元処理を迂回するには、以下のステップを実行します。

1. 独自のプロシージャを作成して、ウィンドウが除去された後にディスプレイを再ビルドします。ユーザーが入力し、再表示しなければならぬデータを組み込んでいることを確認してください。
2. システムに保管させないようにする最初のウィンドウの次のウィンドウに、レコード・レベルの USRRSTDSP キーワードを指定します。USRRSTDSP キーワードは、システムが保管および復元操作を実行できないようにします。USRRSTDSP キーワードは、WINDOW キーワードを含むレコードのみで使用できます。ウィンドウ参照レコードでは無視されます。

USRRSTDSP キーワードを指定すると、オプション標識をオフに設定しても、初期のウィンドウなしディスプレイまたは USRRSTDSP キーワードを指定したウィンドウの 2 つ前のウィンドウに対して読み取りまたは書き込みを行うまで、有効になります。

ディスプレイ上に 6 つのウィンドウがあり、4 つ目のウィンドウに USRRSTDSP キーワードが指定されたとします。USRRSTDSP をオフにして、システムにディスプレイの保管を再開させるには、2 つ目のウィンドウに書き込まなければなりません。図のように、システムは 1 つ目と 2 つ目のウィンドウ



のみ保管しています。

RBAHG504-0

USRRSTDSP キーワードの処理と対話

USRRSTDSP キーワードは、他のキーワードやウィンドウ関連の機能と対話します。このキーワードを使用する前に、以下の点を理解している必要があります (USRRSTDSP キーワードが有効であることを前提にしています)。

- 保管されたウィンドウ (上記の例ではウィンドウ 1 または 2) にウィンドウ・レコードが書き込まれる場合、保管されたディスプレイは復元され、現行のレコードはターゲット・ウィンドウに書き込まれ、ターゲット・ウィンドウは活動状態になる。この時点で、USRRSTDSP キーワードは有効ではなくなります。
- 保管されなかったウィンドウ (上記の例ではウィンドウ 3、4、または 5) にウィンドウ定義レコードが書き込まれると、新しいウィンドウになる。以前のディスプレイ・イメージとマージされ、ディスプレイに書き込まれます。ウィンドウは除去されません。
- 保管されなかったウィンドウ (上記の例ではウィンドウ 3、4、または 5) からウィンドウ・レコードが読み取られると、エラー・メッセージがアプリケーションに戻される。
- 初期画面が保管され、RMVWDW キーワードを指定しているウィンドウ・レコードにアプリケーションが書き込む場合、既存のウィンドウは除去される。初期画面の上部に新しいウィンドウが表示されます。新しいウィンドウは活動状態になり、USRRSTDSP は有効ではなくなります。
- 初期画面が保管されず、RMVWDW キーワードを指定しているウィンドウ・レコードにアプリケーションが書き込む場合、既存のウィンドウはすべて除去される。初期画面の上部に新しいウィンドウが表示されます。新しいウィンドウは活動状態になり、USRRSTDSP は依然として有効です。
- 非ウィンドウ・レコードがディスプレイに書き込まれ、USRRSTDSP が最初のウィンドウに指定されている場合は、ウィンドウは除去されず、非ウィンドウ・レコードはウィンドウのすべてまたは一部をオーバーレイすることがある。

プログラミングの例

以下に、ウィンドウ・キーワードの基本機能について図示します。最初の例は、さまざまなウィンドウ機能の使用法を示します。この例では、1 つのディスプレイ・ファイル中に 1 つのフルスクリーン・ディスプレイと複数のウィンドウを定義します。2 番目の例では、別のディスプレイ・ファイル中に定義されたフルスクリーン・ディスプレイに関するウィンドウを作成する方法を示します。3 番目の例では、メニュー・バー・サポートをシミュレートする方法を示します。

基本的なウィンドウ機能の使用法

以下のシナリオは、ウィンドウ・キーワードの基本機能について示します。このシナリオは、以下の 3 つの節で示されます。

- フルスクリーン・ディスプレイとウィンドウの定義に使用される DDS
- フルスクリーン・ディスプレイとウィンドウの表示に使用される RPG プログラム
- 結果の検討と図示

DDS のフルスクリーン・ディスプレイとウィンドウの定義

以下の DDS は、このシナリオで使用される初期画面と 2 つのウィンドウを定義します。

```

A*-----*
A* DISPLAY FILE - DEMOFM
A*-----*
A*-----*
A* FILE LEVEL KEYWORDS
A*-----*
A
A          DSPSIZ(24 80 *DS3)
A          HELP
A*-----*
A* RECORDS USED IN DEFINING INITIAL DISPLAY
A*-----*
A      R INITIAL
A
A          CA03(03)
A          CA04(04)
A          CA06(06)
A          CA07(07)
A          CA12(12)
A          CLRL(*ALL)
A          3 28'NONWINDOW DISPLAY RECORD'
A          6 1'FLD #1:'
A      FLD48      30A B 6 9
A          6 43'FLD #2:'
A      FLD49      15A B 6 51
A          10 1'FLD #3:'
A      FLD50      30A B 10 9
A          10 43'FLD #4:'
A      FLD51      15A B 10 51
A          23 1'F3=Xxxx'
A          23 13'F4=XXXXXXXXX'
A          23 29'F6=XXXXX XXXXX'
A          23 48'F7=Xxxx'
A          23 59'F12=CANCEL'
A*

```

```

A*-----*
A* RECORDS USED IN DEFINING WINDOW1
A*-----*
A      R WINDOW1                WINDOW(7 3 11 33)
A N01                WDWBORDER((*COLOR GRN))
A 01                WDWBORDER((*COLOR RED))
A                        2 13 'WINDOW #1'
A                        4 2 'CUSTOMER NO.'
A      FLD1            6A 0 4 15
A                        4 22 ':'
A*
A      R REC2WIN1            WINDOW(WINDOW1)
A                        OVERLAY
A                        CA12(12)
A                        10 2 'F12=CANCEL'
A*
A      R REC3WIN1            WINDOW(WINDOW1)
A                        OVERLAY
A                        6 2 'NAME:'
A      FLD2            24A B 6 8
A                        7 2 'ADDRESS:'
A      FLD3            21A B 7 11
A                        8 2 'PHONE:'
A      FLD4            23A B 8 9
A*
A*-----*
A* RECORDS USED IN DEFINING WINDOW2
A*-----*
A      R WINDOW2                WINDOW(9 25 11 32)
A*
A                        2 12 'WINDOW #2'
A      FLD5            22A 0 4 6
A      FLD6            25A 0 5 4
A      FLD7            25A 0 6 4
A      FLD8            25A 0 7 4
A*
A      R REC2WIN2            WINDOW(WINDOW2)
A                        OVERLAY
A                        CA12(12)
A                        10 8 'XXXXXXX :'
A      FLD9            6A B 10 18
A*
A*-----*

```

RPG 表示プログラム

この RPG プログラムは、上記のように定義されたフルスクリーン・ディスプレイとウィンドウを表示します。ステップ 1 からステップ 5 までは、プログラムの後で説明されています。

ディスプレイ・ファイル、例

```
F*****
F*   RPG PROGRAM - WINDEMO
F*****
FDEMOFM CF E           WORKSTN
C*****
C* Step 1: Display Initial Display
C*****
C           EXFMTINITIAL
C*****
C* Step 2: Display Window #1
C*****
C           MOVE 'nnnnn'  FLD1
C           WRITEWINDOW1
C           WRITEREC2WIN1
C           EXFMTREC3WIN1
C*****
C* Step 3: Display Window #2
C*****
C           MOVEL'XXXXXX 'TEMP16 16
C           MOVE 'xxxx x x'TEMP16
C           MOVELTEMP16  FLD5
C           MOVE 'xxxxxx'  FLD5
C*
C           MOVEL'xxxxxx x'FLD6
C           MOVEL'xx xxxx 'TEMP16
C           MOVE 'xxx xxxx'TEMP16
C           MOVELTEMP16  TEMP17 17
C           MOVE ' '      TEMP17
C           MOVE TEMP17  FLD6
C*
C           MOVEL'xxxx xxx'FLD7
C           MOVEL'xxxx x x'TEMP16
C           MOVE 'x xxxx x'TEMP16
C           MOVELTEMP16  TEMP17
C           MOVE 'x'      TEMP17
C           MOVE TEMP17  FLD7
C*
C           MOVEL'xxxxxxxx 'FLD8
C           MOVEL'xxxx xxx'TEMP16
C           MOVE 'xx xxxxx'TEMP16
C           MOVELTEMP16  TEMP17
C           MOVE '.'      TEMP17
C           MOVE TEMP17  FLD8
C*
C           WRITEWINDOW2
C           EXFMTREC2WIN2

C*****
C* Step 4: Restore Window #1
C*****
C           EXFMTREC3WIN1
C*****
C* Step 5: Display Initial Display
C*****
C           READ INITIAL           91
C*****
C* End The RPG Program
C*****
C           SETON           LR
C*****
```


ステップ 1: 初期画面の表示

アプリケーションは、ウィンドウ・キーワードを使用せずに初期画面を作成します。

```

NONWINDOW DISPLAY RECORD

FLD #1: _____ FLD #2: _____

FLD #3: _____ FLD #4: _____

F3=Xxxx   F4=Xxxxxxxxx   F6=Xxxxx Xxxxx   F7=Xxxx   F12=CANCEL

```

ステップ 2: ウィンドウ 1 の表示

ユーザーはディスプレイ上にデータを入力し、Enter キーを押します。アプリケーションはウィンドウ定義レコード WINDOW1 に書き込み、このレコードはウィンドウを作成します。次にアプリケーションは、ウィンドウ参照レコード REC2WIN1 に書き込み、ウィンドウ参照レコード REC3WIN1 に対する書き込み/読み取り操作を実行して、ウィンドウに情報を追加します。

ウィンドウが表示される前に、システムは画面即時読み取り操作を実行し、ディスプレイ・イメージを入手して、基礎的なディスプレイを保管します。最初のウィンドウがディスプレイに追加される場合に限り、システムは画面即時読み取り操作を実行します。ウィンドウが作成されるたびに保管操作を実行します。

```

NONWINDOW DISPLAY RECORD

FLD #1: DATA ENTERED HERE_____ FLD #2: DATA ENTERED
:
:           WINDOW #1
F :
: CUSTOMER NO. nnnnnn :
:
: NAME: _____
: ADDRESS: _____
: PHONE: _____
:
: F12=CANCEL
:
: .....
F3=Xxxx   F4=Xxxxxxxxx   F6=Xxxxx Xxxxx   F7=Xxxx   F12=CANCEL

```

ステップ 3: ウィンドウ 2 の表示

ユーザーは、情報を入力して、Enter キーを押します。アプリケーションは、ウィンドウ定義レコード WINDOW2 に書き込みます。WINDOW2 は活動状態のウィンドウではなく、現在ディスプレイ上にありません。したがって、システムは保管済みのデータを WINDOW1 と関連付けて、基礎的なディスプレイを保

ディスプレイ・ファイル、例

管します。次に、新しいウィンドウが作成されます。アプリケーションは、ウィンドウ・レコード REC2WIN2 に対する書き込み/読み取り操作を実行して、ウィンドウに情報を追加します。

```
NONWINDOW DISPLAY RECORD

FLD #1: DATA ENTERED HERE_____  FLD #2: DATA ENTERED
:                                     :
:                                     :
:          WINDOW                     :
F :                                     : D
: CUSTOMER NO. nnnnn :          WINDOW #2           :
:                                     :
: NAME: MY NAME      :          Xxxxxxx xxxx x xxxxxxxx :
: ADDRESS: MY ADDRESS :          xxxxxx xxx xxxx xxx xxxx :
: PHONE: MY PHONE    :          xxxx xxxxxxxx x xx xxxx xx :
:                                     :          xxxxxxx xxxxxxx xxxxxx. :
: F12=CANCEL        :                                     :
:                                     :
:          Xxxxxxx : _____ :
:                                     :
:                                     :
F3=Xxxx  F4=XXXXXXXX  F6=Xxxxx Xxxxx  F7=Xxxx  F12=CANCEL
```

ステップ 4: ウィンドウ 1 の復元

ユーザーは、データを入力して、Enter キーを押します。アプリケーションは、ウィンドウ・レコード REC3WIN1 に対する書き込み/読み取り操作を実行します。WINDOW キーワードに指定されたレコード様式名 WINDOW1 は、活動状態のウィンドウではありません。しかし、ウィンドウは現在ディスプレイ上にあります。したがって、システムは WINDOW1 に関連付けられた保管済みのディスプレイを復元します。復元操作により、WINDOW1 の後に書き込まれた WINDOW2 が除去されます。次に、復元されたウィンドウに REC3WIN1 が書き込まれます。

```
NONWINDOW DISPLAY RECORD

FLD #1: DATA ENTERED HERE_____  FLD #2: DATA ENTERED
:                                     :
:          WINDOW #1                 :
F :                                     :
: CUSTOMER NO. nnnnn :                                     :
:                                     :
: NAME: MY NAME      :                                     :
: ADDRESS: MY ADDRESS :                                     :
: PHONE: MY PHONE    :                                     :
: F12=CANCEL        :                                     :
:                                     :
:                                     :
F3=Xxxx  F4=XXXXXXXX  F6=Xxxxx Xxxxx  F7=Xxxx  F12=CANCEL
```

ステップ 5: 初期画面の表示

ユーザーは、Enter キーを押します。アプリケーションが初期画面に対する読み取り操作を実行すると、最後のウィンドウがこのディスプレイから自動的に除去されます。

```

NONWINDOW DISPLAY RECORD

FLD #1: DATA ENTERED HERE_____ FLD #2: DATA ENTERED

FLD #3: DATA ENTERED HERE_____ FLD #4: DATA ENTERED

F3=Xxxx F4=XXXXXXXX F6=Xxxxx Xxxxx F7=Xxxx F12=CANCEL

```

別のディスプレイ・ファイルでのウィンドウの定義

以下に、フルスクリーン・ディスプレイおよびウィンドウに関する DDS コードと、これらのコードを使用するのに必要な手順のステップについて説明します。前述の例と対照的な点として、この例は基礎的なディスプレイ用のファイルとは別のディスプレイ・ファイルにウィンドウ・レコードを保持します。この手法を使用すると、項目に関するウィンドウ (ヘルプなど) を、既存のアプリケーションに関するディスプレイ・ファイル・コードを再書き込みせずに、それらのアプリケーションに追加できます。

この例では、アプリケーションは RSTDSP(*NO) を使用して、ファイルの中断時に保管操作を行ってはいらないことを示します。ウィンドウを表示すると保管操作も実行されるので、RSTDSP(*NO) を使用すると 2 つの保管操作が実行されなくなります。ウィンドウを除去すると、ウィンドウ操作の前に表示されていた画面が復元されるので、アプリケーションはウィンドウ処理の後にディスプレイを再ビルドする必要はありません。USRRSTDSP キーワードについては、例の末尾にある注記を参照してください。

DDS のフルスクリーン・ディスプレイとウィンドウの定義

```

A*-----*
A* DISPLAY FILE DISPLAY1 (RSTDSP=NO DFRWRT=*YES)
A*-----*
A      R REC1
A              2 21'FIRST RECORD IN FILE'
A              4 17'Current Customer #:'
A      FIELD1      6A B 4 38
A*
A      R REC2
A              OVERLAY CA03(03)
A              6 21'SECOND RECORD IN FILE'
A              8 17'Current Customer #:'
A      FIELD2      6A B 8 39DSPATR(HI)
A              24 02'CA03=EXIT'
A*-----*

```

RPG プログラム・ソース

```

FDISPLAY1CF E
C          WRITEREC1
C          WRITEREC2
C      RETRY      TAG
C          READ REC2          90
C 03          GOTO END

```

ディスプレイ・ファイル、例

```
C          CALL 'WINPGM'
C          GOTO RETRY
C      END  TAG
C          SETON          LR

A*-----*
A* DISPLAY FILE DISPLAY2 (RSTDSP=NO)
A*-----*
A      R WINDOW1          WINDOW(7 4 11 25)
A N01          WDWBORDER((*COLOR GRN))
A 01          WDWBORDER((*COLOR RED))
A          2 9'Window #1'
A *
A      R REC2WIN1        WINDOW(WINDOW1)
A          CA12(12) OVERLAY
A          4 1'Customer No. nnnnnn:'
A          6 1'Name:'
A      FIELD3      19A B 6 7
A          7 1'Address:'
A      FIELD4      16A B 7 10
A          8 1'Phone:'
A      FIELD5      18A B 8 8
A          10 1'F12=Cancel'
A*-----*
A* Dummy record to remove window from display before returning
A*-----*
A      R RMVWDW          CLRL(*NO) OVERLAY FRCDTA
A*-----*
A* No I/O will ever be done to this record. This record prevents the
A* display from clearing.
A*-----*
A      R DUMMY          ASSUME
A          11 1' '
A*-----*
```

WINPGM の RPG プログラム・ソース

```
FDISPLAY1CF E
C          WRITWINDOW1
C          EXFMTREC3WIN1
C          WRITERMVWDW
C          RETRN
C          SETON          LR
```

ステップ 1: 初期画面の表示

アプリケーションはディスプレイ・ファイル DISPLAY1 を開き、レコード REC1 に対する書き込み操作を実行し、REC2 に対する書き込み/読み取り操作を実行します。

```

FIRST RECORD IN FILE
Current Customer #: _____
SECOND RECORD IN FILE
Previous Customer #: _____

```

ステップ 2: ウィンドウの表示

ユーザーはウィンドウを表示する必要があることを示すデータを入力します。アプリケーションは、ウィンドウ・ディスプレイ・ファイル DISPLAY2 を開きます。この操作は、ステップ 1 で画面を表示したプログラムなど、別のプログラムから行えます。レコード DUMMY 上の ASSUME キーワードは、フルスクリーン・ディスプレイが消去されないようにし、システムはディスプレイ・ファイル DISPLAY2 に中断状態としてマークを付けます。レコード DUMMY に対する入出力操作を実行する必要は決して生じません。このレコードがファイル中に存在する必要のみあります。

アプリケーションはレコード WINDOW1 に対する書き込み操作を実行します。ディスプレイ・ファイル DISPLAY1 は中断されます。RSTDSP(*NO) の設定値のために、保管は実行されません。ディスプレイ・ファイル DISPLAY2 は復元されます。RSTDSP(*NO) の設定値のために、復元データは送信されません。

アプリケーションはレコード REC2WIN1 に対する書き込み操作を実行します。

```

FIRST RECORD IN FILE
Current Customer #: xxxxxx
SECOND RECORD IN FILE
.....:
:      Window #1      : #: xxxxxx
:                    :
: Customer No. nnnnnn:
:                    :
: Name: _____   :
: Address: _____ :
: Phone: _____   :
:                    :
: F12=Cancel         :
:                    :
:                    :
:                    :
.....:

```

ステップ 3: 初期画面に戻る

ユーザーは、アプリケーションが初期ディスプレイ・ファイル DISPLAY1 に戻る必要があることを示すデータを入力します。アプリケーションはレコード RMVWDW に対する書き込み操作を実行し、その結果システムはすべてのウィンドウをディスプレイから除去します。アプリケーションは DISPLAY2 ディスプレイ・ファイルを閉じず、USRRSTDSP キーワードは指定されていないので、ウィンドウを除去すると初期画面が復元されます。アプリケーションは初期画面を再ビルドする必要はありません。RMVWDW レコードに FRCDTA キーワードが含まれている必要があります。含まれていない場合は、ディスプレイ・ファイルの作成時に DFRWRT(*NO) を指定する必要があります。

アプリケーションはレコード REC1 に対する書き込み操作を実行します。ディスプレイ・ファイル DISPLAY2 は中断されます。RSTDSP(*NO) の設定値のために、保管操作は実行されません。ディスプレイ・ファイル DISPLAY1 は復元されます。RSTDSP(*NO) の設定値のために、データは送信されません。

アプリケーションは、レコード REC2 に対する書き込み/読み取り操作を実行します。

```
FIRST RECORD IN FILE
Current Customer #: _____
SECOND RECORD IN FILE
Previous Customer #: _____
```

この例に関する追加の注記:

1. ステップ 3 でアプリケーションが DISPLAY2 を閉じる場合、アプリケーションは REC1 に対する書き込み操作を実行してから REC2 に対する書き込み/読み取り操作を実行して、初期画面を再ビルドしなければなりません。DISPLAY1 に RSTDSP(*YES) を指定すると、この操作を行わずに済みます。
2. USRRSTDSP を DISPLAY2 中のウィンドウ・レコードに追加できます。しかし、追加する場合は、ユーザーは DISPLAY1 に RSTDSP(*YES) を指定するか、またはステップ 3 で初期画面を再ビルドしなければなりません。しかし、依然としてステップ 3 でアプリケーションは RMVWDW を書き込まなければならず、書き込まないと、ウィンドウに対する次の書き込み操作でウィンドウの枠は適切に表示されません。
3. 以下の条件が両方とも真である場合、KEEP キーワードをウィンドウ様式に追加する必要があります。
 - ステップ 3 でアプリケーションが DISPLAY2 を閉じる。
 - DISPLAY2 を開いて再びウィンドウを表示する前に、ファイル DISPLAY1 に対する入出力が行われない。

第 6 章 画面用のグラフィカルな表示の作成

グラフィカルな表示は、DDS 機能の実行時の外観における変更点です。点線で囲まれたウィンドウの代わりに、はっきりとしたウィンドウ枠があります。オプション欄に数値をタイプ入力して選択する代わりに、マウスまたは略号を使用できます。略号とは、選択項目のテキストの中の下線が付いた文字で、それを入力して選択項目を選択することができます。オプション番号を表示する代わりに、ラジオ・ボタンまたはチェック・ボックスを表示することができます。ラジオ・ボタンとは、グラフィカル・ディスプレイ装置上の単一項目選択フィールド内で選択項目の前に置かれる円のことで、チェック・ボックスとは、グラフィカル・ディスプレイ装置上の複数項目選択フィールド内で選択項目の前に置かれる正方形の箱のことで、ラジオ・ボタンやチェック・ボックスをクリックして選択を行うことができます。

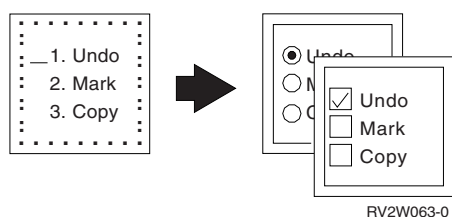


図 34. ラジオ・ボタンとチェック・ボックス

より新しい外観に加えて、拡張機能にはメニュー・バーとプルダウン・メニューが含まれます。長い DDS コーディングを使用してメニュー・バーをシミュレートする代わりに、DDS キーワードを使用することができます。

グラフィカルな表示に影響を与える要素

この章で説明されているグラフィカル機能は、ご利用のハードウェア構成と、拡張表示 (ENHDSP) パラメーターで指定した値により異なって表示されます。

ハードウェア構成

表 15 および 153 ページの表 16は、さまざまな構成で各グラフィカル機能がどのように表示されるかを示しています。表の中の文字 A から F は構成を示しています。これらの文字はこの章全体で使用されています。

表 15. ハードウェア構成 A、B、および C でサポートされる機能

	ハードウェア構成		
	A	B	C
機能	グラフィカル・ユーザー・インターフェース (GUI) プログラマブル・ワークステーション ¹	拡張インターフェースをサポートする制御装置 ³ に接続された InfoWindow II デイスプレィ装置 ²	拡張インターフェースをサポートする制御装置 ³ に接続された 3477 ディスプレィ装置
ウィンドウ ⁹	GUI ⁴ ウィンドウおよびパフォーマンスの向上の可能性	文字ベースの GUI ⁵ およびパフォーマンスの向上の可能性	文字ウィンドウおよびパフォーマンスの向上の可能性

表 15. ハードウェア構成 A、B、および C でサポートされる機能 (続き)

機能	ハードウェア構成		
	A	B	C
	グラフィカル・ユーザー・インターフェース (GUI) プログラマブル・ワークステーション ¹	拡張インターフェースをサポートする制御装置 ³ に接続された InfoWindow II デ스플레이装置 ²	拡張インターフェースをサポートする制御装置 ³ に接続された 3477 ディ스플레이装置
選択フィールドおよびメニュー・バー	GUI ⁴	文字ベースの GUI ⁵	略号、バー選択カーソル
選択リスト	バー選択カーソル。複数選択リストではチェック・ボックスの可能性。単一選択リストではラジオ・ボタンの可能性。	バー選択カーソル。複数選択リストではチェック・ボックスの可能性。単一選択リストではラジオ・ボタンの可能性。	バー選択カーソル。リストの左方に入力フィールド。
継続入力フィールド	1 つのフィールド	1 つのフィールド	1 つのフィールド
マスク編集	あり	あり	あり
強調表示	あり	あり	あり
カーソル進行	あり	あり	あり
語のスピル	あり ⁸	あり	あり
単純ホット・スポット	あり	あり	なし
スクロール・バー	GUI ⁴ スクロール・バー	文字ベースの GUI ⁵ スクロール・バー	*MORE、*PLUS、または文字スクロール・バー ⁶
押しボタン	あり	あり	あり
自動選択	あり	あり	あり
自動入力	あり	あり	あり
プログラマブル・マウス・ボタン	あり ⁸	あり	なし
グリッド・ライン ⁷	なし	なし	なし

注:

- たとえば、RUMBA/400 (Microsoft® Windows® および OS/2®) および AIX® AS/400 CONNECTION PROGRAM/6000 リリース 2。
- InfoWindow II ディ스플레이装置: 3486、3487、3488。
- 平衡型制御装置: 5494 リリース 1.1、および機構 6050、2661、9146、および 9148。
- GUI には、実線のウィンドウ枠、選択可能な背景色、ポインター装置の使用 (マウスなど)、略号選択、バー選択カーソル、ラジオ・ボタン (単一項目選択フィールドの場合)、およびチェック・ボックス (複数項目選択フィールドの場合) が含まれます。
- 文字ベースの GUI では構成体が文字を使用して作成され、背景色は選択可能ではないという点を除いて、文字ベースの GUI は GUI に似ています。
- ポインティング・デバイスがないディスプレイ装置に表示されるスクロール・バーは、表示専用です。
- グリッド・ラインは DBCS ディ스플레이装置のみでサポートされます。グリッド・ラインに必要な特定のハードウェアについては、228 ページの『グリッド・ライン構造におけるハードウェア要件』を参照してください。
- RUMBA/400 は現在この機能をサポートしていません。
- RUMBA/400 は現在ウィンドウ・フッターをサポートしていません。

表 16. ハードウェア構成 D、E、および F でサポートされる機能

機能	ハードウェア構成		
	D	E	F
	拡張インターフェースをサポートする制御装置 ¹ に接続された 5250 ディスプレイ装置	拡張インターフェースをサポートする ASCII 制御装置 ² に接続された ASCII ディスプレイ装置	拡張インターフェースをサポートしない制御装置 ³ に接続された任意のディスプレイ装置
ウィンドウ ⁷	文字ウィンドウおよびパフォーマンスの向上の可能性	文字ウィンドウおよびパフォーマンスの向上の可能性	文字ウィンドウ
選択フィールドおよびメニュー・バー	バー選択カーソル	バー選択カーソル	入力フィールド主導型
選択リスト	バー選択カーソル。リストの左方に入力フィールド。	バー選択カーソル。リストの左方に入力フィールド。	リストの左方に入力フィールド。
継続入力フィールド	1 つのフィールド	1 つのフィールド	複数のフィールド
マスク編集	あり	あり	無視
強調表示	あり	あり	無視
カーソル進行	あり	あり	無視
語スビル ⁶	あり	あり	無視
単純ホット・スポット	なし	なし	なし
スクロール・バー	*MORE、*PLUS、または文字スクロール・バー ⁴	*MORE、*PLUS、または文字スクロール・バー ⁴	*MORE、*PLUS、または文字スクロール・バー ⁴
押しボタン	あり	あり	あり
自動選択	あり	あり	なし
自動入力	あり	あり	あり
プログラマブル・マウス・ボタン	なし	なし	なし
グリッド・ライン ⁵	なし	なし	なし

注:

1. 平衡型制御装置: 5494 リリース 1.1、および機構 6050、2661、9146、および 9148。
2. 拡張インターフェースをサポートする ASCII 制御装置: 機構 6041、6141、2637、9145、9147。
3. 例: 5294 および 5394 制御装置または機構 2638、6040、および 6140 に接続された 5250 ディスプレイ装置。接続された 5250 ディスプレイ装置がある制御装置をエミュレートするプログラマブル・ワークステーションの一部 (iSeries Access for Windows など)。
4. ポインティング・デバイスがないディスプレイ装置に表示されるスクロール・バーは、表示専用です。
5. グリッド・ラインは DBCS ディスプレイ装置のみでサポートされます。グリッド・ラインに必要な特定のハードウェアについては、228 ページの『グリッド・ライン構造におけるハードウェア要件』を参照してください。
6. RUMBA/400 は現在この機能をサポートしていません。
7. RUMBA/400 は現在ウィンドウ・フッターをサポートしていません。

拡張表示パラメーター

- | 拡張表示 (ENHDSP) パラメーターは、CRTDSPF および CHGDSPF コマンドで使用することができます。
- | このパラメーターを使用して、ディスプレイ装置に表示されているデータがディスプレイ装置で使用可能な
- | 拡張機能を使用するかどうかを指定します。

通常、DDS ウィンドウおよび CUA[®] グラフィカル・アイテムは、ディスプレイ装置で使用可能な拡張機能ならどれでも使用してレンダリングされます。たとえば、ウィンドウ枠およびメニュー・バー区切り記号は、グラフィカル・ディスプレイ装置にグラフィカルに表示されます。

ディスプレイ装置の機能にかかわらず、ENHDSP(*NO) を使用すると、ディスプレイ・ファイルで定義されたすべてのレコードを文字ベースのモードで表示させることができます。ENHDSP(*NO) を指定すると、特定のディスプレイ装置で使用可能な拡張機能はどれも使用されません。すなわち、レコードは、153 ページの表 16 の構成 F のディスプレイ装置のときのように表示されます。

ENHDSP のデフォルト値は *YES です。ディスプレイ装置のすべての拡張機能は自動的に使用されます。ENHDSP(*YES) を指定し、デフォルトのウィンドウ枠とメニュー・バー区切り記号を使用する場合、ウィンドウ枠とメニュー・バー区切り記号は実線で表示されます。ディスプレイ装置が、非プログラマブル・ワークステーション用の拡張インターフェースをサポートしない制御装置に接続されている場合、ENHDSP(*YES) は無視されます。ディスプレイ・ファイル内のレコードは、文字ベースのモードで (ENHDSP(*NO) が指定されている場合のように) そのディスプレイ装置に表示されます。

ENHDSP(*YES) が指定されたファイルのレコードと ENHDSP(*NO) が指定されたファイルのレコードを同じ画面に書き込む。 表示されているレコード (複数の場合もあり) が ENHDSP(*YES) が指定されたファイルからのレコードである場合、ENHDSP(*NO) が指定されたファイルのレコードの最初の書き込み操作 (ウィンドウなど) により、すべてのメニュー・バー、プルダウン・メニュー、および画面上の他のウィンドウがグラフィカルから文字ベースに変更されます。ENHDSP(*YES) が指定されたファイルのレコードは ENHDSP(*NO) モードの表示に切り替わります。

表示されているレコードが ENHDSP(*NO) が指定されたファイルからのレコードである場合、ENHDSP(*YES) が指定されたファイルのレコードの書き込み操作 (ウィンドウなど) で、メニュー・バー、プルダウン・メニュー、または画面上の他のウィンドウの表示は変更されません。

注:

1. UIM ヘルプで使用されるシステム・ファイルは、出荷時には ENHDSP(*NO) です。ENHDSP(*YES) が指定されたファイルが含まれる UIM ヘルプを使用する場合、画面はグラフィカルから文字ベースに変更されます。
2. 拡張インターフェースをサポートするプログラマブル・ワークステーションの中には、ウィンドウ枠キーワードおよびメニュー・バー区切り記号キーワードを無視するものがあります。
3. 枠が 1 桁目、80 桁目 (画面サイズ 24 x 80 の場合)、または 132 桁目 (画面サイズ 27 x 132 の場合) にあるようなディスプレイ装置にウィンドウが書き込まれる場合、常にウィンドウは、ENHDSP(*NO) が指定されている場合のように表示されます。

DDS キーワード

この章のタスクでは DDS キーワードを参照しますが、それらのすべての詳細を説明しているわけではありません。それぞれのキーワードについての詳細は、iSeries Information Center の **DDS** のトピックを参照してください。

CCSID (コード化文字セット識別コード)

「G」タイプ・フィールドが DBCS グラフィカル・データの代わりに UCS-2 レベル 1 データをサポートするように指定します。

CHCACCEL (アクセラレーター・テキスト選択)

プルダウン・レコード内の単一項目選択フィールドのアクセラレーター・キーに対してテキストを指定します。

CHCAVAIL (選択可能項目のカラー/表示属性)

メニュー・バーまたは選択フィールドでの選択可能項目の表示時に使用されるカラーまたは表示属性を指定します。

CHCCTL (選択項目制御)

フィールドの選択項目の選択可能性を制御します。

CHCSLT (選択された項目のカラー/表示属性)

メニュー・バー内の選択された項目の表示時に使用されるカラーまたは表示属性を指定します。

CHCUNAVAIL (選択不可能な項目のカラー/表示属性)

選択フィールド内の選択不可能な項目の表示時に使用されるカラーまたは表示属性を指定します。

CHOICE (選択フィールド選択)

選択フィールド選択項目を定義します。

CNTFLD (継続入力フィールド)

フィールドを継続入力フィールドとして定義します。継続入力フィールドは、フィールド・データの入力および編集時にワークステーション制御装置によって単一の入力フィールドとして扱われる、関連付けられた入力フィールドのセットです。

EDTMSK (編集マスク)

EDTCDE または EDTWRD キーワードを持つフィールドについての編集マスクを定義します。

ENTFLDATR (入力フィールド属性)

カーソルがフィールド内に置かれると指定された属性に変わるフィールドの先行属性を定義します。フィールド・レベルとレコード・レベルの両方で定義されると、フィールド・レベルの仕様がフィールドについて使用されます。

FLDCSRPRG (カーソル移動先フィールド)

カーソルがこのフィールドから出るときに移動する先のフィールドを定義します。

GRDATR (グリッド・ライン属性)

ファイルまたはレコード内のグリッド・ライン構造のカラーおよび線のタイプ属性を定義します。

GRDBOX (グリッド・ボックス)

ボックスの形状、位置、および属性を定義します。

GRDCLR (グリッド消去)

すべてのグリッド構造が消去される画面の長方形領域を定義します。

GRDLIN (グリッド・ライン)

グリッド・ラインの形状、位置、および属性を定義します。

GRDRCD (グリッド・レコード)

このレコードがグリッド構造を定義するように指定します。このキーワードがあるレコードには他の表示フィールドを入れることは許可されません。

HLPID (ヘルプ識別コード)

フィールドのヘルプ内の固定情報についての識別コードを指定します。

HTML (ハイパーテキスト・マークアップ言語)

データ・ストリームが i5/OS 5250 ワークステーション・ゲートウェイ・ディスプレイに送られた場合、データ・ストリームと一緒に HTML タグが送られるように指定します。これらの HTML タグは HTML ブラウザーで処理されます。これを使用すると、ワールド・ワイド・ウェブを介してインターネットで、使用するアプリケーションを更新できるようになります。

MLTCHCFLD (複数項目選択フィールド)

フィールドを複数項目選択フィールドとして定義します。複数項目選択フィールドとは、ユーザーが複数の選択項目を選択できる、特定数の選択項目が入ったフィールドのことです。

MNUBAR (メニュー・バー)

メニュー・バーを定義します。メニュー・バーは選択項目の水平なリストで、区切り行が次に置かれます。

MNUBARCHC (メニュー・バー選択項目)

メニュー・バー・フィールドの選択項目を定義します。メニュー・バー選択項目は、アプリケーション・ユーザーが選択できる、関連するアクションのグループを表します。

MNUBARDSP (メニュー・バー表示)

メニュー・バーを表示します。

MNUBARSEP (メニュー・バー区切り記号)

メニュー・バー区切り行を形成するのに使用されるカラー、表示属性、または文字を指定します。

MNUBARSW (メニュー・バー切り替えキー)

CA キーをメニュー・バー切り替えキーに割り当てます。

MNUCNL (メニュー取り消しキー)

CA キーをメニュー・バーまたはプルダウン・メニューのための取り消しキーに割り当てます。

MOUBTN (プログラマブル・マウス・ボタン)

アテンション標識 (AID) をさまざまなポインター装置イベントに関連付けられるようにします。

PSHBTNCHC (押しボタン選択項目)

押しボタン・フィールド内の押しボタンを定義します。

PSHBTNFLD (押しボタン・フィールド)

フィールドを押しボタン・フィールドとして定義します。押しボタン・フィールドは、固定数の押しボタンがあるフィールドです。押しボタンとは、ユーザーが押しボタンを選択したときに開始されるアクションを表すテキストまたはグラフィック (あるいはその両方) のラベルが付いたボタンです。

PULLDOWN (プルダウン・メニュー)

レコードをメニュー・バーにおけるプルダウン・メニューとして定義します。

SFLCSRPRG (サブファイル・カーソル移動先)

カーソルがフィールドから出るときに、次のサブファイル・レコード内の同じ入力フィールドに移動させます。

SFLCHCCTL (サブファイル選択制御)

選択リストの選択項目の選択可能性を制御します。

SFLEND (サブファイル終了)

サブファイルの右下の位置に正符号 (+) またはテキスト (「続く...」または「終わり」) を表示します。これは、スクロール・バーも表示できます。

SFLMLTCHC (サブファイル複数選択項目選択リスト)

サブファイルを複数項目選択リストとして定義します。複数項目選択リストは、基本的にスクロール可能なリストで、ユーザーはそこから 1 つ以上の項目を選択することができます。

SFLRCDNBR (サブファイル・レコード番号)

このフィールドにある相対レコード番号を持つレコードの入っているサブファイルのページを表示します。

SFLRTNSEL (サブファイル戻り選択済み項目)

選択リストにおけるすべての選択項目を `get-next-changed` 命令を使用して戻します。

SFLSCROLL (サブファイル・スクロール)

制御がアプリケーションに渡される時にサブファイルの最上部にあるサブファイル・レコードの相対レコード番号を戻します。

SFLSIZ (サブファイル・サイズ)

サブファイル内のレコード数を指定します。

SFLSNGCHC (サブファイル単一選択項目選択リスト)

サブファイルを単一項目選択リストとして定義します。単一項目選択リストは、基本的にスクロール可能なリストで、ユーザーはそこから 1 つの項目を選択することができます。

SNGCHCFLD (単一選択項目フィールド)

フィールドを単一項目選択フィールドとして定義します。単一項目選択フィールドとは、ユーザーが 1 つの選択項目を選択できる、特定数の選択項目が入ったフィールドのことです。

メニュー・バーの作成

図 35 には、画面の最上部に表示される水平の選択項目のリストであるメニュー・バーの例が示されています。オプションのメニュー・バー区切り記号は、リストの下に表示されます。メニュー・バーの選択項目から選択するとき、プルダウン・メニューが表示されます。プルダウン・メニューとは、メニュー・バー選択項目に関連したアクションのグループのことです。

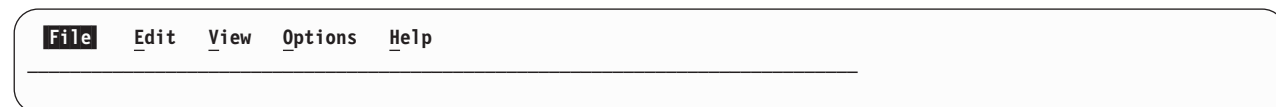


図 35. メニュー・バーの例

メニュー・バー選択項目の定義

メニュー・バーは、MNUBAR キーワードおよび 1 つのメニュー・バー・フィールドを持つ、特殊なタイプのレコードです。メニュー・バー・フィールドは、1 つ以上の MNUBARHC キーワードが含まれる数字フィールドです。MNUBARHC キーワードは、各選択項目と関連したメニュー・バー選択項目とプルダウン・メニューを定義します。メニュー・バーは常時、第 1 行目に表示されます。メニュー・バー・レコードには、メニュー・バー・フィールド以外の表示可能なフィールドを含めることはできません。

メニュー・バーが占める行数は、選択項目の数、すべてのメニュー・バー選択項目のための選択テキストの最大長、およびメニュー・バー区切り記号が指定されているかどうかにより決定されます。メニュー・バー区切り記号を指定した場合 (デフォルト)、メニュー・バーが占める行数 (メニュー・バー区切り記号を含む) は、12 行以下でなければなりません。メニュー・バー区切り記号を指定しない場合、メニュー・バーが占める行数は、11 行以下でなければなりません。メニュー・バー選択項目の範囲を 12 行目まで拡張することはできません。ただし、12 行目を別のレコードのために使用することはできます。

メニュー・バーが占める最大行数の下に収まりきらないほど大きなプルダウン・レコードを定義する場合、ファイルは作成されません。

メニュー・バー区切り記号の抑制

デフォルトでは、メニュー・バー区切り記号が表示されます。メニュー・バー区切り記号を抑制するには、MNUBAR(*NOSEPARATOR) を指定します。メニュー・バー区切り記号を抑制した場合、メニュー・バー区切り記号 (MNUBARSEP) キーワードを指定できません。

メニュー・バー区切り記号の定義

メニュー・バー区切り記号用のシステム・デフォルトを使用することもできますし、MNUBARSEP キーワードを使用することもできます。MNUBARSEP を使用すると、メニュー・バー区切り記号のカラーと表示属性、さらに区切り記号を構成する文字を指定できます。メニュー・バー区切り記号のデフォルトの表示は、151 ページの表 15 の構成 A および B のディスプレイ装置では実線です。メニュー・バー区切り記号を構成するデフォルトの文字は、151 ページの表 15 の構成 C、および 153 ページの表 16 の構成 D、E、および F のディスプレイ装置では下線 () です。MNUBARSEP キーワードの使用例については、図 36 を参照してください。

注: メニュー・バー区切り記号はプルダウン・メニューの上部枠として使用されます。そのカラーは、プルダウン・メニューが表示されるときに変更されません。画面が一貫性を持って表示されるように、メニュー・バー区切り記号とプルダウン・メニュー枠の両方に同じカラー属性および表示属性を使用してください。

図 36 は、メニュー・バー用の DDS の例を示しています。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R MNUBAR          MNUBAR
A          MNUFLD          2Y 0B 1 2
A          MNUBARHC(1 PULLFILE +
A          '>File          ')
A 02      MNUBARHC(2 PULLEDIT +
A          &EDITTXT)
A          MNUBARHC(3 PULLVIEW +
A          '>View          ')
A 04      MNUBARHC(4 PULLOPT +
A          '>Options          ' +
A          &RTNFLD)
A          MNUBARHC(5 PULLHELP +
A          '>Help          ')
A          MNUBARSEP((*COLOR WHT))
A          EDITTXT          20A P
A          RTNFLD          2Y 0H
A          .
A          .
A          .
```

図 36. メニュー・バー用の DDS: フィールド EDITTXT にはテキスト >Edit が入っていると想定します。

MNUBARHC キーワードのオプション標識を指定することにより、どのメニュー・バー選択項目を表示するかを制御できます。オプション標識は、メニュー・バー選択項目を表示するか (オプションはオン) それとも表示しないか (オプションはオフ) を指定するために、アプリケーション・プログラムにより使用されます。オプション標識の DDS は、158 ページの図 36 に示されています。アプリケーションは、オプション標識をオンまたはオフに指定した後、メニュー・バー・レコードを (MNUBARDSP を有効にせずに) 書き込んでオプション標識をシステムに送ります。メニュー・バー選択項目のオプションがオフになっている場合、選択項目のリストは圧縮されます。しかし、メニュー・バー・レコードが占める行数は圧縮されません (レコードは可変長にはならないため)。メニュー・バーが占める行数は、すべてのメニュー・バー選択項目が表示されるときに必要な行数に、メニュー・バー区切り記号用の 1 行を加えた行数になります。オ

プシヨンの設定により、選択項目のリストがより少ない行数で表示されるように圧縮される場合、区切り行は、選択項目の最後の行の次の行に表示されます。メニュー・バー区切り記号と画面上の次のレコードの間には空白行があります。

メニュー・バー内の各選択項目のために表示されるテキストは、名前付きプログラム・システム間フィールド、または MNUBARHC キーワードの選択項目テキスト・パラメーターに指定されたテキストのいずれかから取得されます。メニュー・バー・レコードのために必要であるとシステムにより計算される行数は、各プログラム・システム間フィールドのサイズまたは選択テキストの長さにより異なります。さらに、各選択項目の間に 3 つ分のスペースがあると想定しています。選択テキスト内の末尾空白はすべて除去されます。残った長さが計算に使用されます。プログラム・システム間フィールドでは末尾空白を予測できないため、そのフィールドの長さは現状のまま使用されます。しかし、メニュー・バー・レコードが表示される時、末尾空白はすべて除去されます。そのため、実際に占める行数は、計算された数より少ない場合があります。そのようになった場合には、画面上のメニュー・バー区切り記号と次のレコードの間には空白行が表示されます。

151 ページの表 15の構成 A および B のディスプレイ装置では、メニュー・バーは次のような外観になります。

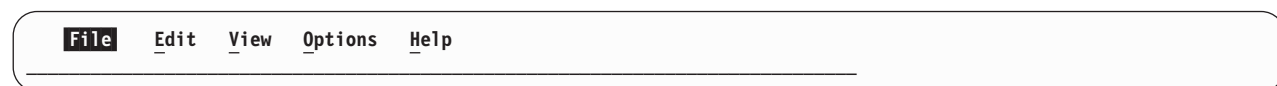


図 37. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上のメニュー・バー

151 ページの表 15の構成 C のディスプレイ装置では、メニュー・バーは次のような外観になります。

注: 図 38では、各メニュー・バー選択項目の先頭文字には下線が付きます。

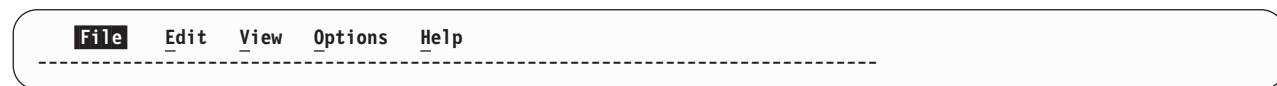


図 38. 下線機能を持つ非グラフィカル・ディスプレイ装置上のメニュー・バー

153 ページの表 16の構成 D および E のディスプレイ装置では、メニュー・バーは次のような外観になります。

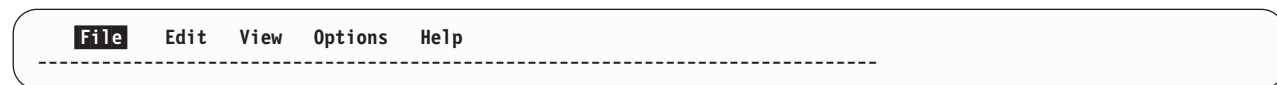


図 39. 下線機能がない非グラフィカル・ディスプレイ装置上のメニュー・バー

153 ページの表 16の構成 F のディスプレイ装置では、メニュー・バーは次のような外観になります。

図 40. 拡張インターフェースがないディスプレイ装置上のメニュー・バー

選択フィールドの概要

選択フィールドには、単一項目と複数項目の 2 つのタイプがあります。

単一項目選択フィールドおよび複数項目選択フィールドには、垂直または水平リストに表示される固定された選択項目のグループが含まれています。複数項目選択フィールドからは任意の数の選択項目を選択できます。単一項目選択フィールドからは 1 つしか選択できません。

151 ページの表 15 の構成 A および B のディスプレイ装置では、選択フィールド (垂直形式) は次のような外観になります。

```
Display Title

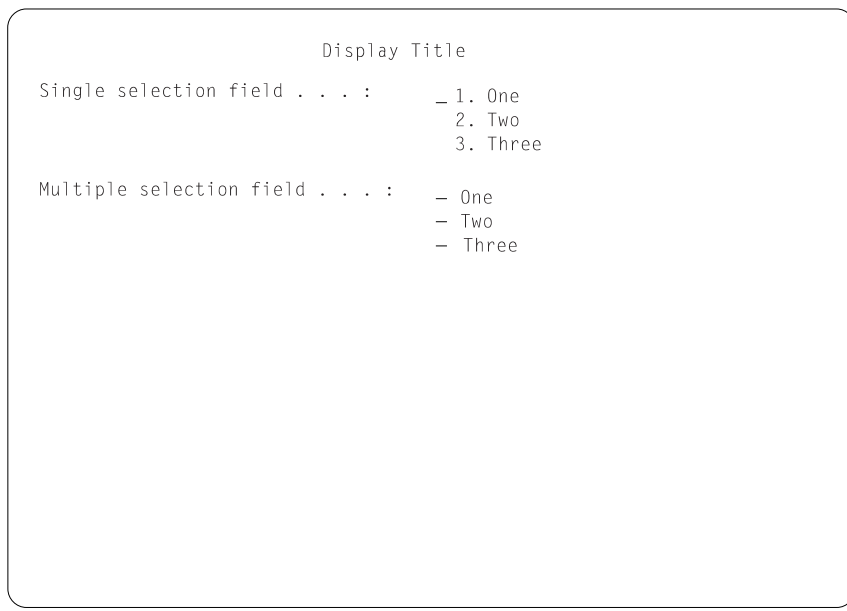
Single selection field . . . :  One
                              Two
                              Three

Multiple selection field . . . :  One
                                 Two
                                 Three
```

RV2W860-1

図 41. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上の選択フィールド

151 ページの表 15 の構成 C、および 153 ページの表 16 の構成 D、E、および F のディスプレイ装置では、選択フィールド (垂直形式) は次のような外観になります。



RV2W070-0

図 42. 非グラフィカル・ディスプレイ装置上の選択フィールド

選択フィールド用 DDS の例

162 ページの図 43は、単一項目および複数項目選択フィールド両方のための DDS の例を示しています。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R RECORD
A
A          2 30'Display Title'
A          4 5'Single selection field . . . : '
A          F1          2Y 0B 4 40SNGCHCFLD
A          CHOICE(1 '>One')
A          CHCCTL(1 &CTLONE1 MSG1111 QUSER/A)
A 01          CHOICE(2 '>Two')
A          CHCCTL(2 &CTLTW01 &MSG1 &LIB/&MSGF)
A          CHOICE(3 'T>hree')
A          CHCCTL(3 &CTLTHR1)
A          CTLONE1          1Y 0H
A          CTLTW01          1Y 0H
A          CTLTHR1          1Y 0H
A          MSG1             7A P
A          LIB              10A P
A          MSGF             10A P
A          8 5'Multiple selection field . . . : '
A          F2          2Y 0B 8 40MLTCHCFLD
A          CHOICE(1 '>One')
A          CHCCTL(1 &CTLONE2 MSG1112 QUSER/A)
A 01          CHOICE(2 '>Two')
A          CHCCTL(2 &CTLTW02 &MSG2 &LIB/&MSGF)
A          CHOICE(3 'T>hree')
A          CHCCTL(3 &CTLTHR2)
A          CTLONE2          1Y 0H
A          CTLTW02          1Y 0H
A          CTLTHR2          1Y 0H
A          MSG2             7A P

```

図 43. 単一項目および複数項目選択フィールドのための DDS

垂直単一項目選択フィールドの作成

選択項目の数および各選択項目の選択番号を定義できます。151 ページの表 15 の構成 A および B のディスプレイ装置では、選択項目の前にラジオ・ボタンが置かれます。SNGCHCFLD キーワードに *NOSLTIND が指定されていないかぎり、そのように処理されます。数字順になっていない選択項目間にはブランク行が表示されます。単一項目選択フィールドで指定する位置は、入力フィールドの位置です (文字ベースの非グラフィカル・ディスプレイの場合)。151 ページの表 15 の構成 A および B のディスプレイ装置では、フィールドで指定する位置は最初のラジオ・ボタンの位置です。

注:

1. 選択標識を抑制した場合、単一項目選択フィールドで指定する位置は、最初の選択項目の先頭文字の位置です。
2. 単一項目選択フィールドがプルダウン・メニューの中にある場合、指定する位置はプルダウン・メニュー枠によって異なります。

単一項目選択フィールドは、1 つの SNGCHCFLD キーワードおよび 1 つ以上の CHOICE キーワードが含まれる数字フィールドです。CHOICE キーワードは、単一項目選択フィールド内の選択項目を定義します。図 43 は、使用するキーワードの例を示しています。

2 桁の長さまでの選択番号を指定できます。選択項目に指定できる最大数は 99 です。出力では、フィールドに選択番号が含まれている場合、その選択項目がデフォルトの選択となります。

デフォルトでは垂直選択フィールドになります。SNGCHCFLD キーワード上で *NUMCOL または *NUMROW の値を使用すると、水平選択フィールドを作成できます。詳しくは、『水平選択フィールドの作成』を参照してください。

垂直複数項目選択フィールドの作成

複数項目選択フィールドは、以下のものが含まれる特殊な数字フィールドです。

- フィールドを複数項目選択フィールドとして識別するための MLTCHCFLD キーワード。ユーザーが選択するとき、フィールド自体に選択される選択項目の番号が入ります。
- 選択項目を定義する 1 つ以上の CHOICE キーワード。
- 選択項目ごとに隠しフィールドを定義するための CHCCTL キーワード (選択項目ごとに)。隠しフィールドは、選択項目が選択されたかどうかを示すために使用されます。CHCCTL キーワードの詳細については、184 ページの『選択項目の選択可能性の制御』を参照してください。

デフォルトでは垂直選択フィールドになります。MLTCHCFLD キーワード上で *NUMCOL または *NUMROW の値を使用すると、水平選択フィールドを作成できます。詳しくは、『水平選択フィールドの作成』を参照してください。

水平選択フィールドの作成

単一項目および複数項目選択フィールドのデフォルトの方向は垂直です。水平選択フィールドを指定するには、SNGCHCFLD および MLTCHCFLD キーワード上で *NUMCOL または *NUMROW の値を使用します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R RECORD
A          2 2'Flavor . . . '
A          F1          2Y 0B 2 16SNGCHCFLD((*NUMCOL 2))
A          CHOICE(1 'Chocolate ')
A 01          CHOICE(2 'Strawberry ')
A          CHOICE(3 'Vanilla ')
A          CHOICE(4 'Peach ')
A
```

図 44. 水平選択フィールド用 DDS の例: *NUMCOL 2 は、フィールドが 2 列に表示されることを指定します。

オプション標識 01 がオンであると想定して、この単一項目選択フィールドがどのように文字ベースの画面に表示されるかを以下に示します。

```
Flavor . . . _ 1. Chocolate 2. Strawberry
                3. Vanilla 4. Peach
```

*NUMROW 2 が指定された場合に、この単一項目選択フィールドがどのように表示されるかを以下に示します。

```
Flavor . . . _ 1. Chocolate 3. Vanilla
                2. Strawberry 4. Peach
```

選択項目が数字順になっていない場合、*NUMCOL で表示される場合のように、省略された選択項目に対する空白行または空白・スペースは残りません。

SNGCHCFLD および MLTCHCFLD キーワードで *GUTTER 値を使用することにより、選択項目と選択項目の間に表示されるスペースの数をオプションで指定できます。ガター幅は 2 以上でなければなりません。*GUTTER を指定しない場合、選択項目間のスペース数のデフォルトは 3 です。

注: ガター幅には、ガターのいずれかの側の選択項目の開始および終了属性が含まれます。

水平選択フィールドが占める領域は以下を基にして決定されます。

- 指定された選択項目の数
- 最長の選択項目の長さ
- ガターの幅
- 指定された桁の数
- 指定された最長のアクセラレーター・テキスト (プルダウン・メニュー内の水平単一項目選択フィールドの場合)

水平選択フィールドは、ファイルに対して指定された最小画面サイズ (24 x 80 または 27 x 132) の中に収まらなければなりません。レコードがウィンドウまたはプルダウン・メニューである場合、水平選択フィールドは最小ウィンドウ・サイズに収まらなければなりません。水平選択フィールドの右または左に他のフィールドを指定することが可能です。他のフィールドと同様に、水平選択フィールドにもオプション標識を指定できます。

CHOICE キーワードでオプション標識を使用することにより、どの選択項目を一度に表示するかを制御できます。垂直選択フィールドとは異なり、ある選択項目のオプションがオフになっている場合、残りの選択項目は桁移動され、スペースが埋められます。

垂直項目選択フィールドにおけるカーソル移動

SNGCHCFLD および MLTCHCFLD キーワードで *NORSTCSR を使用すると、カーソル・キーで、押されたキーの方向にあるカーソル配置可能選択項目にカーソルを移動できます。カーソルは、ヌルの選択項目およびカーソル配置不能として定義された選択項目をスキップします。上矢印キーはカーソルを 1 項目上に移動します。下矢印はカーソルを 1 項目下に移動します。カーソルが最上部の選択項目にあり、上矢印キーが押された場合、カーソルはフィールドからなくなります。同様に、カーソルが最下部の選択項目にあり、下矢印が押された場合、カーソルはフィールドからなくなります。入力可能位置にのみカーソル移動 (CSRINPONLY) キーワードが有効である場合、カーソルは、画面上の現行カーソル位置の上または下にある次のカーソル配置可能項目に移動します。カーソル左移動キーおよび右移動キーは、カーソルを左方または右方に 1 スペース分移動します。

注: 選択フィールドがプルダウン・メニュー内で定義された唯一のフィールドである場合、カーソル左移動キーおよび右移動キーは、現在のプルダウン・メニューをクローズし、左または右にある次のプルダウン・メニューをオープンします。

カーソルを選択フィールド内に保持するためには、SNGCHCFLD および MLTCHCFLD キーワードで *RSTCSR 値を使用します。カーソルがリストの最上部の選択項目にあるときにカーソル上移動キーを押すと、カーソルはリストの最後の選択項目に移動します。カーソルがリストの最下部の選択項目にあるときにカーソル下移動キーを押すと、カーソルはリストの最上部の選択項目に移動します。カーソル左移動キーを押すと、カーソルは 1 項目上に移動します。カーソルが最上部の選択項目にある場合、カーソルは最下部の選択項目に移動します。カーソル右移動キーを押すと、カーソルは 1 項目下に移動します。カーソルが最下部の選択項目にある場合、カーソルは最上部の選択項目に移動します。

注: 選択フィールドがプルダウン・メニュー内で定義された唯一のフィールドである場合、カーソル左移動キーおよび右移動キーは、現在のプルダウン・メニューをクローズし、左または右にある次のプルダウン・メニューをオープンします。

水平項目選択フィールドにおけるカーソル移動

SNGCHCFLD および MLTCHCFLD キーワードで *NORSTCSR を使用すると、カーソル・キーで、押されたキーの方向にあるカーソル配置可能選択項目にカーソルを移動できます。カーソルは、ヌルの選択項目およびカーソル配置不能として定義された選択項目をスキップします。カーソルがフィールド内のいずれかの列の最上部の選択項目にあり、上矢印キーが押された場合、カーソルはフィールドからなくなります。同様に、カーソルがフィールド内のいずれかの列の最下部の選択項目にあり、下矢印キーが押された場合、カーソルはフィールドからなくなります。入力可能位置のみカーソル移動 (CSRINPONLY) キーワードが有効である場合、カーソルは、画面上の現行カーソル位置の上または下にある次のカーソル配置可能項目に移動します。左矢印キーが押され、カーソル配置可能項目が現在の選択項目の左にある場合、カーソルはその選択項目に移動します。カーソル配置可能項目が左にない場合、カーソルはフィールドからなくなります。右矢印キーが押され、カーソル配置可能項目が現在の選択項目の右にある場合、カーソルはその選択項目に移動します。カーソル配置可能項目が右にない場合、カーソルはフィールドからなくなります。

注:

カーソルを選択フィールド内に保持するためには、SNGCHCFLD および MLTCHCFLD キーワードで *RSTCSR 値を使用します。カーソルがフィールド内のいずれかの列の最上部にあるときにカーソル上移動キーを押すと、カーソルは以下のいずれかの場所に移動します。

- 左の列にカーソル配置可能位置がある場合、カーソルはその列の最後の選択項目に移動します。
- 左の列にカーソル配置可能位置がないか、左に列がない場合、カーソルはフィールド内の右端の列の最後のカーソル配置可能選択項目に移動します。

カーソルがフィールド内のいずれかの列の最後の選択項目にあるときにカーソル下移動キーを押すと、カーソルは以下のいずれかの場所に移動します。

- 右の列にカーソル配置可能位置がある場合、カーソルはその列の最上部の選択項目に移動します。
- 右の列にカーソル配置可能位置がないか、右に列がない場合、カーソルはフィールド内の左端の列の最初のカーソル配置可能選択項目に移動します。

カーソル左移動キーが押され、カーソル配置可能選択項目が現在の選択項目の左にある場合、カーソルはその選択項目に移動します。左にカーソル配置可能選択項目がない場合、カーソルは現在の行の上の行の最初のカーソル配置可能選択項目に移動します (一番近い行の右端の選択項目)。現在の行が先頭行である場合、カーソルは最後の行の右端の選択項目に移動します。

カーソル右移動キーが押され、カーソル配置可能選択項目が現在の選択項目の右にある場合、カーソルはその選択項目に移動します。右にカーソル配置可能選択項目がない場合、カーソルは現在の行の下の行の最初のカーソル配置可能選択項目に移動します (一番近い行の左端の選択項目)。現在の行が最下部の行である場合、カーソルは最初の行の左端の選択項目に移動します。

注: 以下が当てはまる場合、カーソル左移動キーおよび右移動キーは、現在のプルダウン・メニューをクローズし、左または右にある次のプルダウン・メニューをオープンします。

- 選択フィールドが、プルダウン・メニュー内で定義された唯一のフィールドである
- 現在の選択項目の左または右にカーソル配置可能選択項目がない

選択フィールド内の選択標識の制御

選択標識とは、選択フィールドまたは選択リスト内の選択項目の前に置かれる標識のことです。それは選択項目を選択したり、選択項目が選択済みであることを示すために使用されます。選択標識の例としてラジオ・ボタンがあります。ラジオ・ボタンは単一項目選択フィールドおよび単一項目選択リスト内の選択項目の前に表示されます。デフォルトは、選択フィールドに選択標識が表示されるというものです。

SNGCHCFLD および MLTCHCFLD キーワードに *NOSLTIND パラメーターを指定することにより、選択フィールド内の選択標識を抑制することができます。

非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されていないディスプレイ装置の場合、*NOSLTIND 値は無視されます。図 45は、選択フィールド内の選択標識を抑制するための DDS の例です。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R RECORD
A          2 30'Display Title'
A          4 5'Single selection field . . . : '
A          F1          2Y 0B 4 40SNGCHCFLD(*NOSLTIND)
A          CHOICE(1 '>One')
A 01          CHOICE(2 '>Two')
A          CHOICE(3 'T>hree')
A          8 5'Multiple selection field . . . : '
A          F2          2Y 0B 8 40MLTCHCFLD(*NOSLTIND)
A          CHOICE(1 '>One')
A 01          CHOICE(2 '>Two')
A          CHOICE(3 'T>hree')
```

図 45. 選択フィールド内の選択標識を抑制するための DDS

151 ページの表 15の構成 A および B のディスプレイ装置では、選択フィールドは次のような外観になります。

```

                Display Title
Single selection field . . . : One
                               Two
                               Three

Multiple selection field . . . : One
                               Two
                               Three
```

RV3W067-0

図 46. 選択フィールド内で抑制された選択標識

単一項目選択フィールドを使用したプルダウン・メニューの作成

メニュー・バーの選択項目から選択するとき、プルダウン・メニューが表示されます。プルダウン・メニューとは、メニュー・バー選択項目に関連付けられたアクションのグループのことです。図 47は一例です。

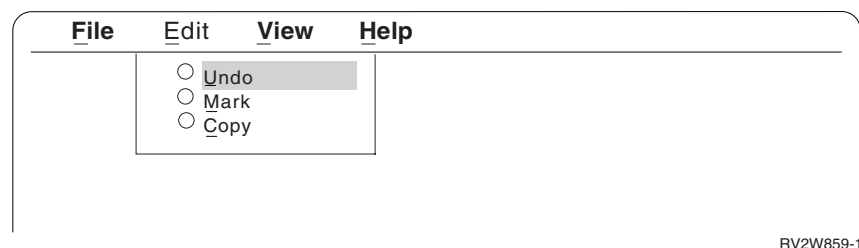


図 47. プルダウン・メニューの例

プルダウン・メニューとそれに対応するメニュー・バーを、同じファイル内で定義する必要があります。

すべてのプルダウン・メニューの最後のフィールドは常に、CHECK(FE) (フィールド終了) キーワードが指定されている場合のように作動します。これにより、プルダウン・メニューの最後のフィールドの入力データを入力した後に、カーソルをプルダウン・メニュー内に保持します。その後、最後のフィールドにカーソルを置いてフィールド・エグジット・キーを押すと、フィールドは消去され、カーソルは次のプルダウン・メニューに移動します。最後のフィールドが CHECK(FE) で作動しなかった場合、Enter キーを押した後、カーソルは自動的に次のプルダウン・メニューに移動します。

プルダウン・レコードには、ウィンドウ・レコードに含められるものをすべて含めることができます。ただし、プルダウン・メニューには単一項目選択フィールドまたは複数項目選択フィールドのみを使用してください。単一項目選択フィールドまたは複数項目選択フィールド以外のフィールドを使用する場合、カーソルはすべてのディスプレイ装置において一貫性を持って移動するわけではありません。非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されているディスプレイ装置では、以下の両方の条件を満たした場合に、カーソル右移動キーおよびカーソル左移動キーにより、次のプルダウン・メニューが表示されます。

- 表示されているプルダウン・メニューに選択フィールドが 1 つだけある。
- カーソルがその選択フィールドに置かれている。

非プログラマブル・ワークステーション用の拡張インターフェースをサポートしない制御装置に接続されているディスプレイ装置では、カーソル移動キーにより、カーソルはプルダウン・メニュー内で文字位置 1 つ分移動します。

168 ページの図 48は、プルダウン・メニュー用の DDS の例です。その後のいくつかの図は、画面のタイプごとにどのようにプルダウン・メニューが表示されるかを示しています。

注: Edit 選択項目において、レコード PULLDOWN が MNUBARCHC キーワードに指定されており、MARKTXT フィールドにはテキスト >Mark が入っていると想定します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...  
A           R PULLDOWN                PULLDOWN  
A           F1                          2Y 0B 1 1SNGCHCFLD  
A  01                                CHOICE(1 '>Undo      ')  
A                                           CHOICE(2 &MARKTXT)  
A                                           CHOICE(3 '>Copy    ')  
A                                           :  
A
```

図 48. プルダウン・メニュー用の DDS

151 ページの表 15の構成 A および B のディスプレイ装置では、プルダウン・メニューは次のような外観になります。

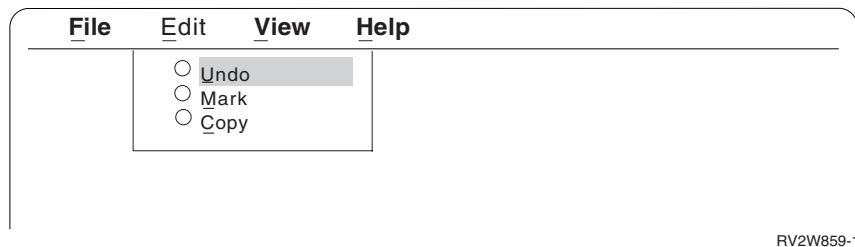


図 49. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上のプルダウン・メニュー

151 ページの表 15の構成 C のディスプレイ装置では、プルダウン・メニューは次のような外観になります。

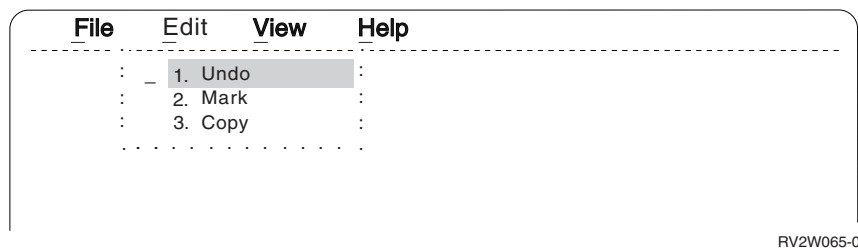


図 50. 下線機能を持つ非グラフィカル・ディスプレイ装置上のプルダウン・メニュー

153 ページの表 16の構成 D および E のディスプレイ装置では、プルダウン・メニューは次のような外観になります。

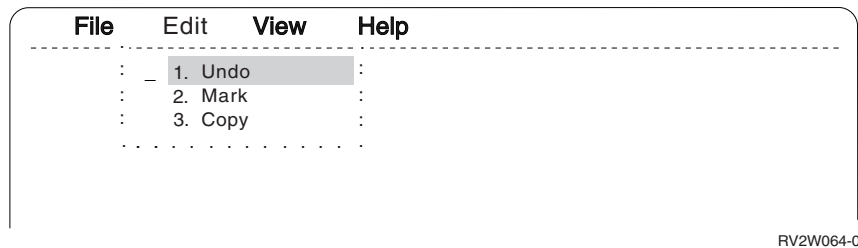
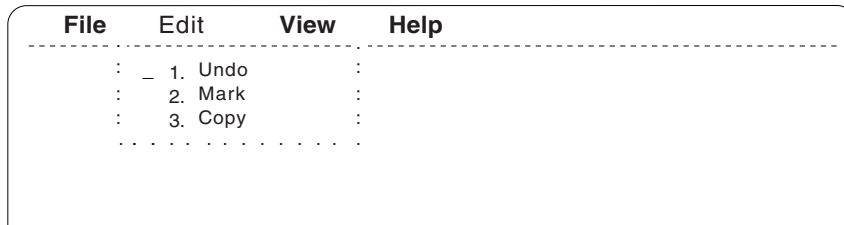


図 51. 下線機能がない非グラフィカル・ディスプレイ装置上のプルダウン・メニュー

153 ページの表 16の構成 F のディスプレイ装置では、プルダウン・メニューは次のような外観になります。



RV3W073-1

図 52. 拡張インターフェースがないディスプレイ上のプルダウン・メニュー

プルダウン・メニュー内の選択標識の制御

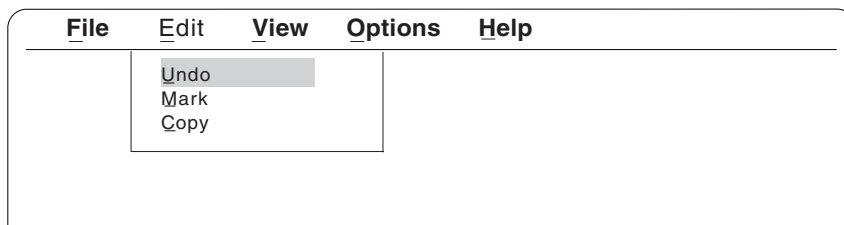
選択標識とは、選択フィールドまたは選択リスト内の選択項目の前に置かれる標識のことです。それは選択項目を選択したり、選択項目が選択済みであることを示すために使用されます。選択標識の例としてラジオ・ボタンがあります。ラジオ・ボタンは単一項目選択フィールドおよび単一項目選択リスト内の選択項目の前に表示されます。デフォルトは、選択フィールドに選択標識が表示されるというものです。

PULLDOWN キーワードに *NOSLTIND パラメーターを指定することにより、プルダウン・メニュー内の選択標識を抑制することができます。図 53は、プルダウン・メニュー内の選択標識を抑制するための DDS の例です。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R PULLDOWN          PULLDOWN(*NOSLTIND)
A          F1                   2Y 0B 1 1SNGCHCFLD
A  01                   CHOICE(1 '>Undo      ')
A                   CHOICE(2 &MARKTXT)
A                   CHOICE(3 '>Copy      ')
A
```

図 53. プルダウン・メニュー内の選択標識を抑制するための DDS

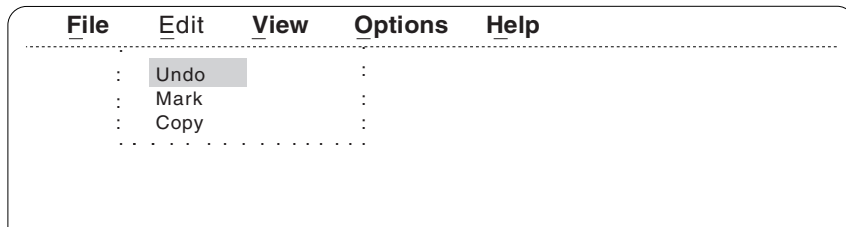
151 ページの表 15の構成 A および B のディスプレイ装置では、プルダウン・メニューは次のような外観になります。



RV2W857-1

図 54. グラフィカル・ディスプレイ装置上で抑制された選択標識

151 ページの表 15の構成 C のディスプレイ装置では、プルダウン・メニューは次のような外観になります。



RV2W066-1

図 55. 非グラフィカル・ディスプレイ装置上で抑制された選択標識

非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されていないディスプレイ装置の場合、*NOSLTIND 値は無視されます。

アクセラレーター・キーの定義

アクセラレーター・キーとは、アプリケーションで定義された機能を始動する機能キーで、プルダウン・メニュー選択項目の隣に表示されるものです。

以下のようにすることにより、プルダウン・メニュー内の単一項目選択フィールドに対してアクセラレーターを指定できます。

1. 必要な CFnn キーを指定する。
2. CHCACCEL キーワードを使用する。

CHCACCEL キーワードにアクセラレーター・テキストを指定します。P フィールドを使用してテキストを指定することができます。CHCACCEL キーワードはアクセラレーター・キー自体は定義しない点に注意してください。キーに対して CFnn キーワードを定義して、この選択項目に対するアクセラレーターとしてこのキーを認識するようにアプリケーションを設計する必要があります。さらに、CHCACCEL に指定するテキストが、定義したキーを正確に反映するようになければなりません。たとえば、CF08 をアクセラレーター・キーにしたい場合、該当する選択項目用の CHCACCEL キーワードに F8 や CF08などを指定してください。

アクセラレーター・テキストは、フィールド内の最長の選択テキストの長さより 3 スペース分後ろに表示されます。

アクセラレーター・キーは、プルダウン・メニューが表示されていない場合であっても機能するため、必要な CFnn キーをファイル・レベルで定義する必要があります。CFnn キーをレコード・レベルで定義した場合、それを使用可能にするすべてのレコードに対して、CFnn キーを指定してください。

単一項目選択フィールドは、すべてのレコードの中で定義できます。しかし、アクセラレーターはプルダウン・メニュー内の単一項目選択フィールドに対してのみ定義できます。171 ページの図 56は一例です。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R PULLEDIT                      CF04 CF06
A          PULLDOWN
A          F1                2Y 0B 1 1SNGCHCFLD
A          CHECK(ER)
A    01          CHOICE(1 '>Undo          ')
A          CHCACCEL(1 'F4')
A          CHOICE(2 &MARKTXT)
A          CHCACCEL(2 &F6)
A          CHOICE(3 '>Copy          ')
A          MARKTXT                20A P
A          F6                2 P

```

図 56. アクセラレーター・キー用の DDS

151 ページの表 15の構成 A および B のディスプレイ装置では、プルダウン・メニューは次のような外観になります。

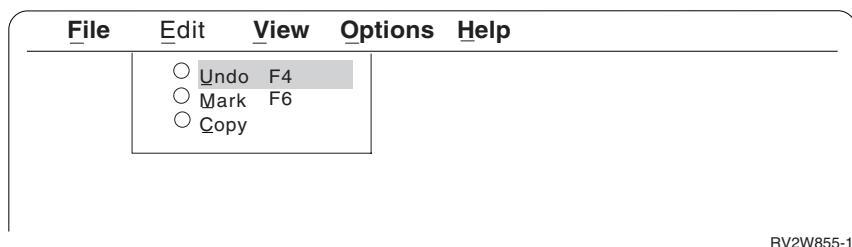


図 57. プルダウン・メニュー内のアクセラレーター

メニュー・バー切り替えキーの定義

メニュー・バー切り替えキーは、メニュー・バーとそのアプリケーション画面との間でカーソルを交互に切り替えます。

ファイル・レベルまたはレコード・レベルのいずれかで **MNUBARSW** キーワードを使用して、メニュー・バー切り替えキーを定義できます。カーソルがアプリケーション・レコードにある場合、メニュー・バー切り替えキーを押すと、カーソルがメニュー・バーの最初の選択項目に移動します。キーを再度押すと、カーソルは、メニュー・バーからアプリケーション・レコード内の直前のカーソル位置に戻ります。カーソルを、カーソル・キーを使用してアプリケーション・レコードからメニュー・バーに移動した後にメニュー・バー切り替えキーを押した場合、カーソルはアプリケーション・レコード上のその初期位置に戻ります。これは、アプリケーション・レコードにカーソル位置キーワードが指定されているのでないかぎり、最初の入力フィールドになります。プルダウン・メニューが表示されている場合、メニュー・バー切り替えキーを押すとプルダウン・メニューが取り消され、カーソルはアプリケーション・レコードに移動します。

アプリケーションまたはシステムのいずれかがメニュー・バーを表示したかにかかわらず、メニュー・バー切り替えキーはシステムが常時処理します。(詳しくは、iSeries Information Center の **DDS** トピックを参照してください。)

メニュー・バー切り替えキーを活動状態にするには、画面に書き込まれた最終レコード上でそのキーが活動状態になっていなければなりません。メニュー・バー切り替えキーを常時活動状態にするための最も簡単な方法は、**MNUBARSW** をファイル・レベルで指定する方法です。**MNUBARSW** をレコード・レベルで指定した場合、メニュー・バー切り替えキーが活動状態となるすべてのレコード上でそれを指定する必要があります。

取り消しキーの定義

MNUCNL キーワードを使用して、メニュー・バー・レコードおよびプルダウン・メニュー・レコード用の取り消しキーを定義できます。それらをファイル・レベルまたはレコード・レベルのいずれかで定義できます。

取り消しキーは、プルダウン・メニューをクローズし、カーソルをメニュー・バー上の関連した選択項目に移動します。カーソルがプルダウン・メニューにない場合でも、そのように処理されます。プルダウン・メニューが表示されておらず、カーソルがメニュー・バーの中に置かれている場合、そのキーによりメニュー・バーが取り消され、カーソルはアプリケーション・レコードに戻ります。メニュー・バー切り替えキーを使用してカーソルをメニュー・バーに移動した場合、この位置は、アプリケーション・レコード内のカーソルの直前の位置となります。カーソルがカーソル・キーでメニュー・バーに移動した場合、この位置はアプリケーション・レコード内のカーソルの初期位置となります。プルダウン・メニューが表示されておらず、カーソルがアプリケーション・レコード上に置かれている場合、そのキーは制御をアプリケーション・プログラムに戻します。この場合、MNUCNL キーワードは他のキー定義キーワードと同様に作動し、応答標識を戻す機能が組み込まれます。

メニュー・バー切り替えキーのように、取り消しキーは、画面に書き込まれた最終レコードにおいて活動状態であった場合のみ、活動状態となります。取り消しキーを常時活動状態にするための最も簡単な方法は、MNUCNL キーワードをファイル・レベルで指定する方法です。MNUCNL キーワードをレコード・レベルで使用した場合、メニュー・バー切り替えキーが活動状態となるすべてのレコード上でそれを指定する必要があります。

図 58は、MNUBARSW キーワードおよび MNUCNL キーワードを使用する方法を示しています。この例では、コマンド・アテンション・キー 10 をメニュー・バー切り替えキーとして、コマンド・アテンション・キー 12 を取り消しキーとしてセットアップします。(これらの設定はデフォルトです。)

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     MNUBARSW(CA10) MNUCNL(CA12)
A                                     MNUBAR
A      R MNUBAR
A      MNUFLD          2Y 0B 1 2
A                                     MNUBARCHC(1 PULLFILE +
A                                     '>File      ')
A 02                                     MNUBARCHC(2 PULLEDIT +
A                                     &EDITTXT)
A                                     MNUBARCHC(3 PULLVIEW +
A                                     '>View      ')
A 04                                     MNUBARCHC(4 PULLOPT +
A                                     '>Options  ')
A                                     MNUBARCHC(5 PULLHELP +
A                                     '>Help    ')
A      EDITTXT          20A P
A      .
A      .
A      .
```

図 58. メニュー・バー切り替えキーおよび取り消しキー用の DDS

カーソルがプルダウン・メニューの外にある場合の機能の制限

PULLDOWN キーワードに *NORSTCSR が指定されている場合、ユーザーはカーソルをアクティブ・ウィンドウの外に移動し、任意のコマンド機能 (CF) キーを使用することができます。*NORSTCSR はデフォルトです。

PULLDOWN キーワードに *RSTCSR が指定されており、カーソルがプルダウン・メニューの外に移動する場合、Print および Home コマンド機能 (CF) キーのみが活動状態となります。ワークステーションのユーザーが他のコマンド機能 (CF) キーを押すとアラームが鳴り、カーソルは直前の書き込み操作におけるカーソル位置に戻ります。

選択リストの概要

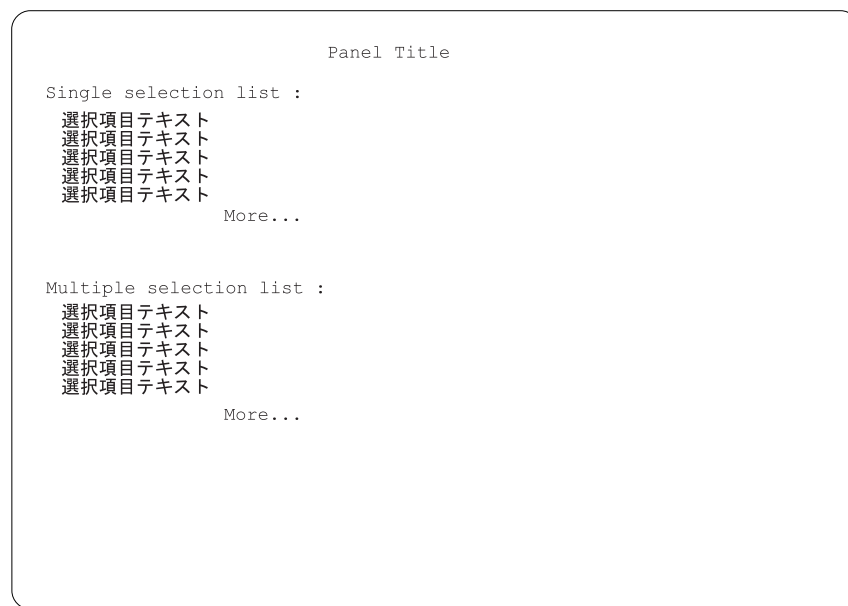
選択リストは、基本的にスクロール可能なリストで、ユーザーはそこから 1 つの項目を選択することができます。選択リストには、単一項目と複数項目の 2 つのタイプがあります。単一項目選択リストは、基本的にスクロール可能なリストで、ユーザーはそこから 1 つの項目を選択することができます。複数項目選択リストは、基本的にスクロール可能なリストで、ユーザーはそこから 1 つ以上の項目を選択することができます。

単一項目選択リストおよび複数項目選択リストには、垂直リストに表示される選択項目のグループが含まれています。これらの選択項目は、前ページ・キーおよび次ページ・キーの使用またはスクロール・バーの使用のいずれかによりスクロールできます。スクロール・バーは、特定の方向に表示可能な他の情報があり、ポインティング・デバイスまたはページ・キーを使用してその情報をビュー内に移動できることをユーザーに示す画面の一部です。スクロール・バーの詳細については、178 ページの『スクロール・バーの概要』を参照してください。

複数項目選択リストからは任意の数の選択項目を選択できます。単一項目選択リストからは 1 つしか選択できません。『選択リストの概要』は、単一項目選択リスト、複数項目選択リスト、および選択リストで使用されるスクロール・バーの例を示しています。

175 ページの図 61 の DDS は、以下の画面を表示します。

151 ページの表 15 の構成 A および B のディスプレイ装置では、選択リストは次のような外観になります。



RV3W077-0

図 59. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上の選択リスト

153 ページの表 16の構成 F のディスプレイ装置では、選択リスト (垂直形式) は次のような外観になります。

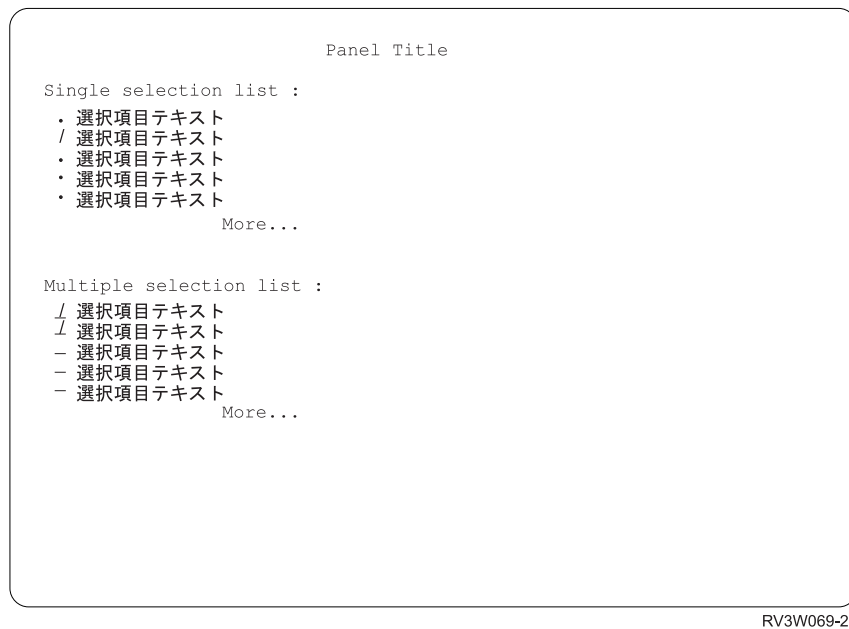


図 60. 下線機能を持つ非グラフィカル・ディスプレイ装置上の選択リスト

選択リスト用 DDS の例

175 ページの図 61の DDS は、173 ページの『選択リストの概要』で示された画面を作成します。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R SFLRCD          SFL
A          CTLFLD          1Y 0H  SFLCHCCTL
A          F1              11A 0  6 10
A          R SFLCTLRCD      SFLCTL(SFLRCD)
A          SFLSNGCHC
A          SFLPAG(5) SFLSIZ(&SFLSIZ)
A          SFLDSP SFLDSPCTL
A          ROLLUP(10)
A          10             SFLEND(*MORE)
A          F3              5S 0H  SFLSCROLL
A          F2              4S 0H  SFLRCDNBR(CURSORS *TOP)
A          SFLSIZ          5S 0P
A
A          1 30'Panel Title'
A          4 5'Single selection list:'
A          R SFLRCD2      SFL
A          CTLFLD          1Y 0H  SFLCHCCTL
A          F1              11A 0 13 10
A          R SFLCTLRCD2   SFLCTL(SFLRCD2)
A          SFLMLTCHC(&NUMSEL *RSTCSR)
A          SFLPAG(5) SFLSIZ(&SFLSIZ)
A          SFLDSP SFLDSPCTL
A          10             SFLEND(*MORE)
A          ROLLUP(10)
A          F2              4S 0H  SFLRCDNBR(CURSORS *TOP)
A          F3              5S 0H  SFLSCROLL
A          SFLSIZ          5S 0P
A          NUMSEL          4Y 0H
A          11 5'Multiple selection list:'

```

図 61. 選択リスト用 DDS の例

選択リストの作成

選択リストはサブファイルを使用して作成されます。サブファイルの詳細については、『第 4 章 サブファイルを使用したレコードのグループの表示』を参照してください。選択リストごとにサブファイル・レコード様式およびサブファイル制御レコード様式を指定する必要があります。サブファイル・レコード様式の中で、選択項目のテキスト用出力フィールドを指定する必要があります。レコード内に出力専用フィールドを 1 つだけ指定することができます。

選択リストにデフォルト選択項目を指定するには、次に変更するサブファイル (SFLNXTCHG) キーワードまたは制御フィールドのいずれかを使用します。SFLNXTCHG キーワードを使用するためには、サブファイル・レコード内のテキスト・フィールド上でキーワードを指定する必要があります。制御フィールドを使用するためには、サブファイル・レコードに制御フィールドを指定し、そのフィールドにサブファイル選択制御 (SFLCHCCTL) キーワードを指定する必要があります。制御フィールドには以下の値を使用することができます。

制御値	出力における意味	入力における意味
0	選択可能	選択されていない
1	選択済み	選択済み
2	選択不可。選択項目のヘルプが使用可能であるのでないかぎり、選択項目にカーソルを置くことはできません。 ^{1,2}	
3	選択不可。選択項目にカーソルを置くことは許可されていません。	

制御値	出力における意味	入力における意味
4	選択不可。選択項目のヘルプが使用可能である場合でも、選択項目にカーソルを置くことはできません。 ^{1,2}	

注:

1. 非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されているディスプレイのみに適用されます。
2. 選択項目が表示される最初の選択項目で、リストには他にカーソル配置可能選択項目がない場合、選択項目は選択不可になり、カーソル配置可能になります。そうでない場合は、無効データ・ストリーム・エラーが出されます。

アプリケーションは、`get-next-changed` 命令を使用して、どの選択項目が選択済みであるかを判別します。`get-next-changed` 命令は、すべての変更されたレコードを戻します。ユーザーがデフォルト選択項目を選択解除した場合、`get-next-changed` 命令は選択解除された選択項目レコードを戻します (その制御値が変更されたため)。`get-next-changed` 命令で選択済み選択項目のみを戻すようにするには、サブファイル戻り選択済み項目 (SFLRTNSEL) キーワードをサブファイル制御レコードに指定します。次回の `get-next-changed` 命令では選択済み選択項目を戻します。その後、デフォルト選択項目に対して更新操作を実行し、その制御値を 0 に変更します。

151 ページの表 15 の構成 A および B のディスプレイ装置では、選択項目の前にラジオ・ボタン (単一項目) およびチェック・ボックス (複数項目) が置かれます。サブファイル単一選択項目選択リスト (SFLSNGCHC) およびサブファイル複数選択項目選択リスト (SFLMLTCHC) キーワードに `*NOSLTIND` が指定されていないかぎり、そのように処理されます。サブファイル・レコード様式の最初のフィールドで指定する位置は、入力フィールドの位置です (文字ベースの非グラフィカル・ディスプレイの場合)。151 ページの表 15 の構成 A および B のディスプレイ装置では、最初のフィールドの位置は最初のラジオ・ボタンまたはチェック・ボックスの位置です。

注:

1. 選択標識を抑制した場合、サブファイル・レコード様式の最初のフィールドで指定する位置は、最初の選択項目の先頭文字の位置です。
2. 選択リストがプルダウン・メニューの中にある場合、指定する位置はプルダウン・メニュー枠によって異なります。

SFLSCROLL キーワードは、レコードの現行ページの最上部にあるレコードの相対レコード番号を戻すために使用されます。ユーザーが `Enter` を押すと、SFLSCROLL は、ページの最上部に現在表示されているレコードの相対レコード番号を戻します。ROLLUP キーワードがあるために制御がアプリケーションに戻る場合、SFLSCROLL はサブファイル内の最終レコードの相対レコード番号に 1 を加えた数を戻します。ROLLDOWN キーワードがあるために制御がアプリケーションに戻る場合、SFLSCROLL は常に 1 を戻します。

正確なサブファイル・レコードでサブファイルを選択項目のリストの最上部に再表示するには、サブファイル・レコード番号 (SFLRCDNBR) キーワードを使用します。SFLRCDNBR(*TOP) を隠しフィールドとして指定し、SFLSCROLL キーワードにより戻された相対レコード番号を使用してください。サブファイルにさらにレコードを追加し、サブファイルを再表示します。

選択リスト内の選択標識の制御

選択標識とは、選択フィールドまたは選択リスト内の選択項目の前に置かれる標識のことです。それは選択項目を選択したり、選択項目が選択済みであることを示すために使用されます。選択標識の例としてラジオ・ボタンがあります。ラジオ・ボタンは単一項目選択フィールドおよび単一項目選択リスト内の選択項目

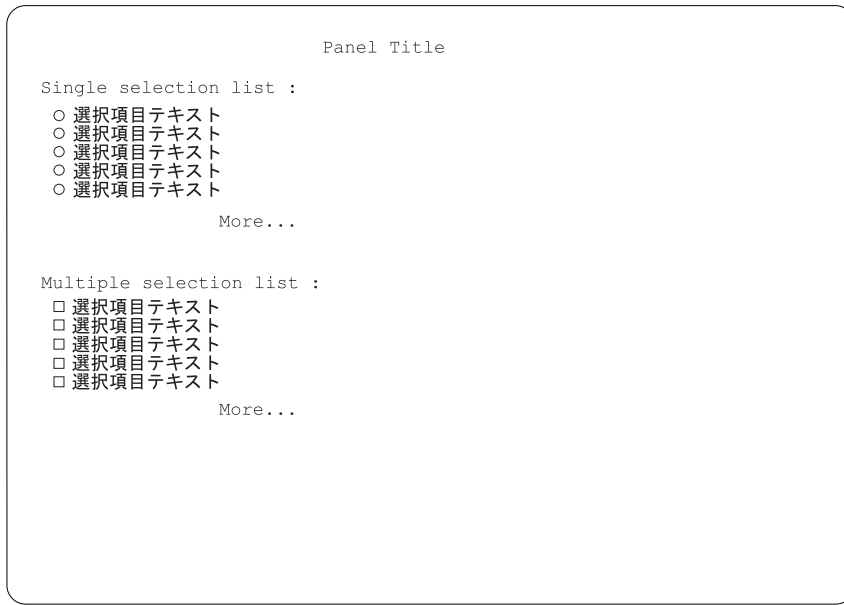
の前に表示されます。デフォルトは、選択リストに選択標識が表示されないというものです。
SFLSNGCHC および **SFLMLTCHC** キーワードに ***SLTIND** パラメーターを指定することにより、選択リストに選択標識を表示させることができます。

非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されていないディスプレイ装置の場合、***SLTIND** 値は無視されます。図 62は、選択リスト内の選択標識を使用可能にするための **DDS** の例です。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R SFLRCD          SFL
A          CTLFLD          1Y 0H  SFLCHCCTL
A          F1              11A 0 6 10
A          R SFLCTLRCD          SFLCTL(SFLRCD)
A          SFLSNGCHC(*SLTIND)
A          SFLPAG(5) SFLSIZ(&SFLSIZ)
A          SFLDSP SFLDSPCTL
A          ROLLUP(10)
A          10              SFLEND(*MORE)
A          F3              5S 0H  SFLSCROLL
A          F2              4S 0H  SFLRCDNBR(CURSORS *TOP)
A          SFLSIZ          5S 0P
A          1 30'Panel Title'
A          4 5'Single selection list:'
A          R SFLRCD2          SFL
A          CTLFLD          1Y 0H  SFLCHCCTL
A          F1              11A 0 13 10
A          R SFLCTLRCD2          SFLCTL(SFLRCD2)
A          SFLMLTCHC(&NUMSEL *RSTCSR *SLTIND)
A          SFLPAG(5) SFLSIZ(&SFLSIZ)
A          SFLDSP SFLDSPCTL
A          10              SFLEND(*MORE)
A          ROLLUP(10)
A          F2              4S 0H  SFLRCDNBR(CURSORS *TOP)
A          F3              5S 0H  SFLSCROLL
A          SFLSIZ          5S 0P
A          NUMSEL          4Y 0H
A          11 5'Multiple selection list:'
```

図 62. 選択リスト内の選択標識を使用可能にするための **DDS**

151 ページの表 15の構成 A および B のディスプレイ装置では、選択リストは次のような外観になります。



RV3W0079-0

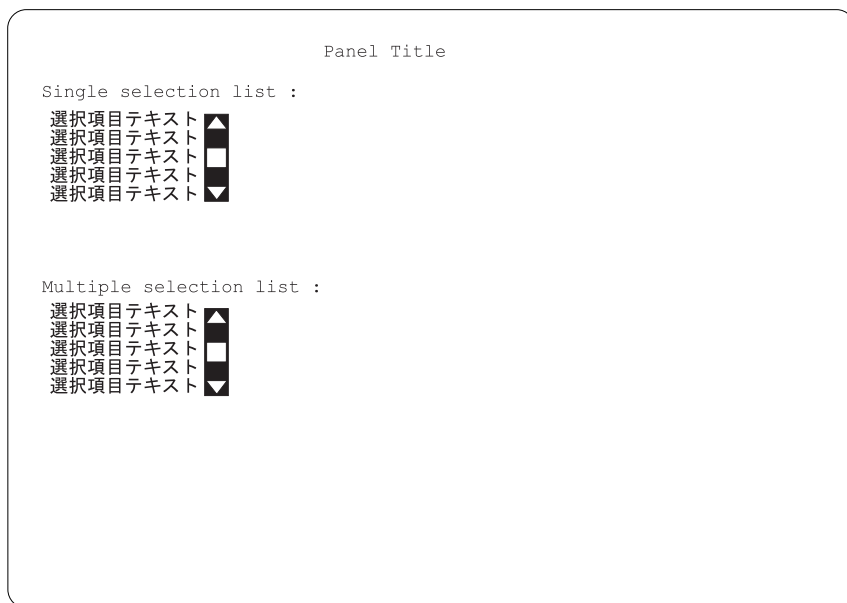
図 63. グラフィカル・ディスプレイ装置上の選択標識

スクロール・バーの概要

スクロール・バーは、特定の方向に表示可能な他の情報があり、ポインティング・デバイスまたはページ・キーを使用してその情報をビュー内に移動できることをユーザーに示す画面の一部です。スクロール・バーは、すべてのサブファイルに対して定義できます。このセクションの例では、単一項目選択リストおよび複数項目選択リストで使用されるスクロール・バーの例を示しています。

181 ページの図 66 の DDS は、以下の画面を表示します。

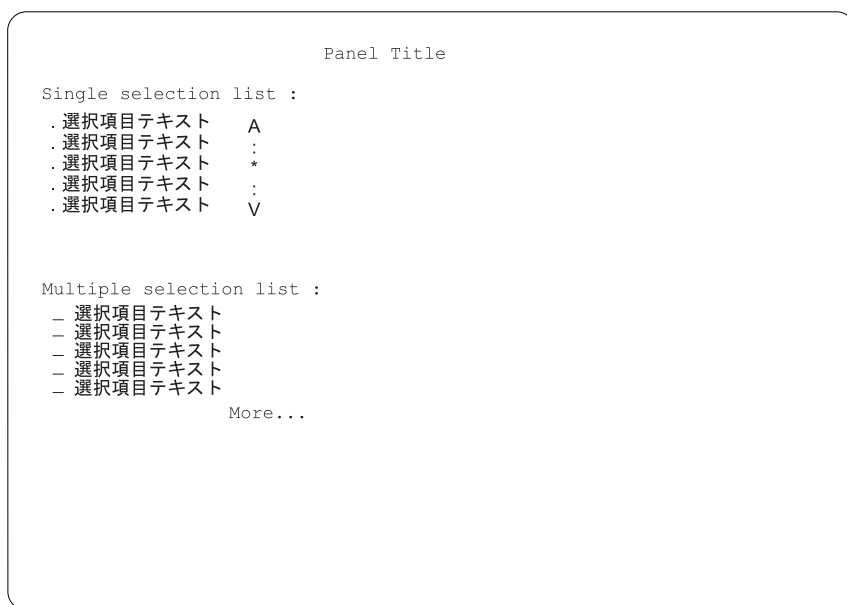
151 ページの表 15 の構成 A および B のディスプレイ装置では、選択リストおよびスクロール・バーは次のような外観になります。



RV3W078-0

図 64. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上のスクロール・バー

153 ページの表 16の構成 F のディスプレイ装置では、選択リスト (垂直形式) は次のような外観になります。



RV3W080-0

図 65. 下線機能を持つ非グラフィカル・ディスプレイ装置上のスクロール・バー

スクロール・バーの作成

スクロール・バーを作成するには、サブファイル終了 (SFLEND) キーワード上でスクロール・バー (*SCRBAR) 値を使用します。 *SCRBAR 値は、グラフィカル・ディスプレイ装置上でグラフィカル・スクロール・バーを作成します。非グラフィカル・ディスプレイ装置上では文字スクロール・バーが作成され

ます。スクロール・バーは、リスト内の最長の選択項目の右側に表示されます。スクロール・バーをさらに右方に表示させるためには、選択リスト選択項目用テキストにブランクを追加してください。

ほとんどの場合、スクロール・バーにより示されるサブファイル・レコードの数は、サブファイルに書き込まれたレコードの数です。SFLEND キーワードのオプションがオンになっている場合、これ以上のレコードがサブファイルに書き込まれないことを示しています。スクロール・バーにより示されるレコードの数にSFLPAG 値は加算されません。サブファイルの最下部に到達し、PAGEDOWN キーワードが活動状態ではない場合、スクロール・バー・ボックスは、下方のスクロール・バー・ボタンのすぐ上に表示されます。

SFLEND キーワードのオプションがオフになっている場合、さらにレコードがサブファイルに書き込まれることを示しています。スクロール・バーにより示されるレコードの数にSFLPAG 値が加算されます。SFLPAG 値を加算すると、最後のサブファイル・レコードの後にさらにサブファイル・レコードが存在しているかのようにスクロール・バーが表示されます。最後のサブファイル・レコードに到達し、PAGEDOWN キーワードが活動状態である場合、スクロール・バー・ボックスは、下方のスクロール・バー・ボタンのすぐ上には表示されません。これは、表示されるレコードがさらにあることを示しています。PAGEDOWN キーワードが活動状態である場合、ユーザーがページ送りまたは非表示レコードへのスクロールを試行すると、制御はアプリケーションに戻ります。その後アプリケーションは、サブファイルにさらにレコードを書き込むことができます。

注: PAGEDOWN が活動状態であり、書き込まれるレコードの数がSFLSIZ 値より少ない場合には、SFLPAG 値は、スクロール・バーにより示されるレコードの数に加算されません。この場合、スクロール・バーにより示されるサブファイル・レコードの数は、SFLSIZ 値です。SFLSIZ 値は、サブファイルの合計サイズを表示し、ユーザーが表示したいサブファイル・レコードのみをアプリケーションが埋められるようにするために使用されます。

PAGEDOWN キーワードが活動状態である場合、部分ページは表示されません。ユーザーが部分ページにページ送りしようとするすると、制御はアプリケーションに戻ります。サブファイル・ロール値 (SFLROLVAL) はこれをオーバーライドする可能性があります。SFLROLVAL キーワードについての詳細は、iSeries Information Center の DDS のトピックを参照してください。PAGEDOWN キーワードが活動状態でない場合、SFLPAG 値から 1 を引いた値が、スクロール・バーにより示されるレコードの数に加算されます。SFLPAG 値から 1 を引いた値を加算すると、部分ページを表示することが可能になります。

注: スクロール・バーが水平サブファイルで表示される場合、以下が該当する場合には上部スクロール・ボタンは使用できない可能性があります。

- 部分ページに到達した
- 水平サブファイルの 1 桁目のみにレコードが表示されている

さまざまな状況でスクロール・バーがサイズ変更される方法の要約について、181 ページの表 17を参照してください。

表 17. スクロール・バーがサイズ変更される方法

SFLEND 標識 の設定	PAGEDOWN の状態	書き込まれた レコードの数	スクロール・バーにより示されるレコードの数に以下の値が加算される (「はい」または「いいえ」)			
			SFLPAG	SFLPAG - 1	最終レコードの 相対レコード番 号	SFLSIZ
オン	活動状態	≥SFLSIZ	いいえ	いいえ	はい	いいえ
		<SFLSIZ	いいえ	いいえ	いいえ	はい
	非活動状態	≥SFLSIZ	いいえ	はい	はい	いいえ
		<SFLSIZ	いいえ	はい	はい	いいえ
	活動状態	≥SFLSIZ	はい	いいえ	はい	いいえ
		<SFLSIZ	いいえ	いいえ	いいえ	はい
オフ	非活動状態	≥SFLSIZ	はい	はい	はい	いいえ
		<SFLSIZ	はい	はい	はい	いいえ

スクロール・バー用 DDS の例

図 66 の DDS は、178 ページの『スクロール・バーの概要』で示された画面を作成します。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R SFLRCD          SFL
A          CTLFLD          1Y 0H    SFLCHCCTL
A          F1              11A 0 6 10
A          R SFLCTLRCD      SFLCTL(SFLRCD)
A                               SFLSNGCHC
A                               SFLPAG(5) SFLSIZ(&SFLSIZ)
A                               SFLDSP SFLDSPCTL
A                               ROLLUP(10)
A 10                               SFLEND(*SCRBAR)
A          F3              5S 0H    SFLSCROLL
A          F2              4S 0H    SFLRCDNBR(CURS *TOP)
A          SFLSIZ          5S 0P
A                               1 30'Panel Title'
A                               4 5'Single selection list:'
A          R SFLRCD2      SFL
A          CTLFLD          1Y 0H    SFLCHCCTL
A          F1              11A 0 13 10
A          R SFLCTLRC2    SFLCTL(SFLRCD2)
A                               SFLMLTCHC(&NUMSEL *RSTCSR)
A                               SFLPAG(5) SFLSIZ(&SFLSIZ)
A                               SFLDSP SFLDSPCTL
A 10                               SFLEND(*SCRBAR *MORE)
A                               ROLLUP(10)
A          F2              4S 0H    SFLRCDNBR(CURS *TOP)
A          F3              5S 0H    SFLSCROLL
A          SFLSIZ          5S 0P
A          NUMSEL          4Y 0H
A                               11 5'Multiple selection list:'

```

図 66. スクロール・バー用 DDS の例

スクロール・バーの操作

スクロール・バーの操作方法の要約については、表 18を参照してください。

表 18. スクロール・バーの操作

ユーザーの操作	動作
上部スクロール・ボタンを 1 度クリックする	サブファイルは、サブファイルの下の方に 1 レコード分スクロールされる
下部スクロール・ボタンを 1 度クリックする	サブファイルは、サブファイルの上の方に 1 レコード分スクロールされる
スクロール・ボックスの上のシャフトを 1 度クリックするか次ページ・キーを押す	サブファイルは、サブファイルの下の方に 1 ページ分スクロールされる
スクロール・ボックスの下のシャフトを 1 度クリックするか前ページ・キーを押す	サブファイルは、サブファイルの上の方に 1 ページ分スクロールされる
スクロール・ボックスを選択ボタンでドラッグし、選択ボタンを放す	サブファイル・ページは、スクロール・ボックスで示される位置に対応するようにスクロールされる。

押しボタンの概要

押しボタンとは、ユーザーが押しボタンを選択したときに開始されるアクションを表すテキストまたはグラフィック (あるいはその両方) のラベルが付いたボタンです。

183 ページの図 71の DDS は、以下の画面を表示します。

151 ページの表 15の構成 A および B のディスプレイ装置では、押しボタンは次のような外観になります。

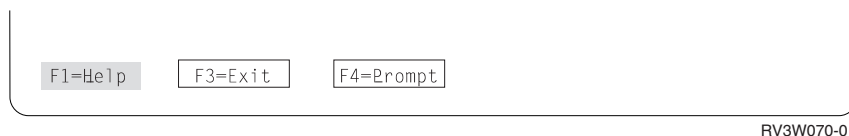


図 67. 拡張インターフェースを持つグラフィカル・ディスプレイ装置上の押しボタン

151 ページの表 15の構成 C のディスプレイ装置では、押しボタンは次のような外観になります。

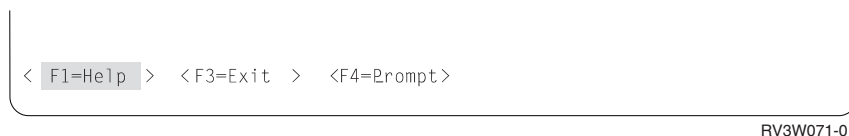


図 68. 下線機能を持つ非グラフィカル・ディスプレイ装置上の押しボタン

151 ページの表 15の構成 D および E のディスプレイ装置では、押しボタンは次のような外観になります。

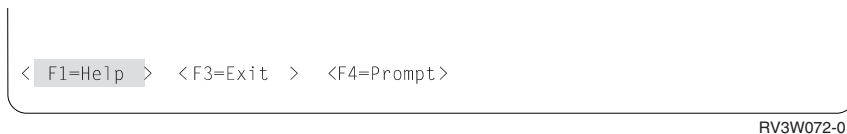


図 69. 下線機能がない非グラフィカル・ディスプレイ装置上の押しボタン

151 ページの表 15 の構成 F のディスプレイ装置では、押しボタンは次のような外観になります。



図 70. 拡張インターフェースがないディスプレイ装置上の押しボタン

押しボタン用 DDS の例

図 71 の DDS は、182 ページの『押しボタンの概要』で示された画面を作成します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R RECORD
A
A          PSHFLD1      2Y 0B 23 4PSHBTNFLD(*RSTCSR (*NUMCOL 3))
A                                PSHBTNCHC(1 'F1=>Help' HELP)
A                                PSHBTNCHC(2 &F3 CF03)
A                                CHCCTL(2 &CTL)
A 02                                PSHBTNCHC(3 'F4=>Prompt' CF04)
A                                CHCAVAIL((*COLOR RED))
A          F3          15A P
A          CTL          1Y 0H
```

図 71. 押しボタン用 DDS の例

押しボタンの作成

押しボタン・フィールドは、押しボタン・フィールド (PSHBTNFLD) キーワードで定義されます。それぞれの押しボタン・フィールドには、1 つ以上の押しボタン選択項目 (PSHBTNCHC) キーワードを含める必要があります。

押しボタンは縦方向または横方向 (デフォルト) に表示することができます。PSHBTNFLD キーワードの *NUMROW 値は、押しボタンを縦方向に表示する場合に使用される行数を指定します。PSHBTNFLD キーワードの *NUMCOL 値は、押しボタンを横方向に表示する場合に使用される桁数を指定します。PSHBTNFLD キーワードにパラメーターが指定されていない場合、押しボタンは、1 行に収まるだけの数の桁数で表示されます。

選択項目制御 (CHCCTL) キーワードを使用して、個々の押しボタンの可用性を制御します。選択可能項目 (CHCAVAIL) および選択不能項目 (CHCUNAVAIL) キーワードを使用して、個々の押しボタンのカラーまたは属性を制御します。

押しボタンごとにキーを指定します。以下の例では、F4=Prompt 押しボタンが選択された時に CF04 キーが戻されます。

```
PSHBTNCHC(3 'F4=>Prompt' CF04)
```

キーを指定しなかった場合、押しボタンが選択された時に Enter が戻されます。

押しボタン選択項目で定義されたキーは、押しボタン・フィールドが含まれるレコードで自動的に使用可能になります。

押しボタンが含まれるフィールドには、選択されている押しボタン選択項目の番号が含まれています。ユーザーが (押しボタンそのものを選択する代わりに) 押しボタンに関連付けられたキーを押した場合、押しボタンの番号は戻されません。選択をしなかった場合には、ゼロが戻されます。

押しボタンごとにヘルプを指定できます。

選択項目の選択可能性の制御

CHCCTL キーワードを使用して、選択フィールド内の選択項目および押しボタン選択項目の選択可能性を制御できます。選択リスト内にある選択項目の選択可能性の制御については、175 ページの『選択リストの作成』を参照してください。隠しフィールドの名前を CHCCTL キーワードに指定します。ご使用のプログラムでこのフィールドに値を設定し、選択項目が選択可能、選択不可、または選択済み (複数項目選択フィールドのみ) のいずれであるかを指定することができます。図 72 は、制御値とそれぞれの意味について記述しています。

制御値	選択項目の設定
0	選択可能 (または選択されていない)
1	選択済み
2	選択不可。選択項目のヘルプが選択可能であるのではないが、選択項目にカーソルを置くことはできません。
3	選択不可。選択項目にカーソルを置くことは許可されています。
4	選択不可。選択項目のヘルプが選択可能である場合でも、選択項目にカーソルを置くことはできません。

図 72. CHCCTL キーワードの制御値

注: 図 72 のカーソル制限は、非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されているディスプレイのみに適用されます。

値 1 は単一項目選択フィールドでは無視されます (デフォルトの選択は単一項目選択フィールド自体で選択を設定することにより行われるため)。この値は押しボタンでも無視されます。値 1 は複数項目選択フィールドで使用され、複数の選択項目のデフォルトを選択済み選択項目にします。

ユーザーが選択不可能な項目を選択した場合に表示されるメッセージを指定することもできます。メッセージを指定しない場合、システムはデフォルトのメッセージを表示します。

185 ページの図 73 は、選択項目の選択可能性を制御するための DDS の例で、通常のレコードでの単一項目選択フィールドを示しています。


```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R RECORD
A          2 2'Flavor . . . '
A          F1          2Y 0B 2 16SNGCHCFLD
A          CHOICE(1 '>Chocolate ')
A          CHCCTL(1 &CTLCHOC MSG1112 QUSER/A)
A 01        CHOICE(2 '>Strawberry ')
A          CHCCTL(2 &CTLSTRA &MSG &LIB/&MSGF)
A          CHOICE(3 '>Vanilla ')
A          CHCCTL(3 &CTLVANI);
A          CHOICE(5 '>Peach ')
A          CTLCHOC          1Y 0H
A          CTLSTRA          1Y 0H
A          CTLVANI          1Y 0H
A          MSG              7A P
A          LIB              10A P
A          MSGF             10A P

```

図 73. 選択項目の選択可能性を制御するための DDS

151 ページの表 15の構成 C、および 153 ページの表 16の構成 D、E、および F のディスプレイ装置では、単一項目選択フィールドは次のような外観になります。

```

Flavor . . . _ 1. Chocolate
                2. Strawberry
                3. Vanilla
                5. Peach

```

選択テキスト *Vanilla* は、選択不可であることを示すため、グレーで表示されます。
 図 74. 使用不可の選択項目がある単一項目選択フィールド：実行時に CTLCHOC および CTLSTRA は 0 (選択可能) に設定され、CTLVANI は 2 (選択不可) に設定されると想定します。

単一項目選択フィールドにおける自動選択

自動選択機能は、ユーザーがカーソルを選択項目に置いて Enter を押すことにより、単一項目選択フィールド内の選択項目を選択することを可能にします。選択項目を次のいずれかを行って選択する必要はありません。(1) 選択番号をタイプ入力する。(2) カーソルを選択項目に置き、スペース・バーを押す。または (3) ポインターを選択項目に置き、左マウス・ボタンを押す。自動選択は、プルダウン・メニュー内の単一項目選択フィールドではデフォルトです。プルダウン・メニュー以外の単一項目選択フィールドのデフォルトは手動選択です。SNGCHCFLD キーワードの *AUTOSLT パラメーターは、Enter が押された時に選択項目が自動的に選択されることを示します。*NOAUTOSLT パラメーターは、選択項目を手動で選択する必要があることを示します。*AUTOSLTENH パラメーターは、非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されているディスプレイのみ、自動選択が有効であることを示します。

単一項目選択フィールドにおける自動入力

自動入力機能は、選択項目が選択された時に、選択された選択項目をプログラムに戻すことを可能にします。選択項目に戻すために Enter を押す必要はありません。*AUTOENT パラメーターは、選択項目が選択されるとすぐに Enter キーが戻されることを示します。いずれかの選択項目において 2 桁の選択番号が必要になる場合を除いて、すべてのディスプレイ装置で選択項目が自動的に戻されます。*NOAUTOENT パラメーターは、選択項目を選択した後、ユーザーが Enter を押すまで選択項目は戻されないことを示しま

す。*NOAUTOENT がデフォルトです。*AUTOENTNN パラメーターは、選択項目の数字選択が必須でない場合にのみ、選択項目が選択されるとすぐに選択項目が戻されることを示します。

略号の定義

以下の項目に対して略号を定義できます。

- メニュー・バー選択項目
- 選択フィールド選択項目 (単一と複数)
- 選択リスト選択項目 (単一と複数)
- 押しボタン

より大 (>) 文字を使用して略号を定義できます。略号を示すには、略号にしたい文字の直前に > 文字を付けます。> 文字はテキスト長の一部として数えられません。> 文字を、略号のポインターとしてではなくテキスト内の文字として使用したい場合、> 文字を 2 つ連続して使用する必要があります。> 文字をテキスト内の最終文字として指定する場合、> 文字はテキストの一部として表示され、略号は表示されません。図 75 は一例です。

キーワード仕様	表示されるテキスト
MNUBARHC(1 PULLFILE '>File')	<u>File</u>
MNUBARHC(2 PULLFIN 'F>inish')	<u>Finish</u>
CHOICE(1 'Save >As...')	Save <u>A</u> s...
CHOICE(2 'X >= 1')	X <u>>=</u> 1
CHOICE(3 'X >>>= 1')	X <u>>>=</u> 1

> 文字を略号として指定することはできません。
図 75. 略号が有効である DDS の例

注: この例で強調して表示される文字は、ご使用のディスプレイでは下線が付きます。

略号は空白にすることはできません。1 つの略号のみを選択テキストに指定することができます。複数の略号が指定された場合、最初の略号のみが選択可能になります。図 76 は、誤ったコーディングの例です。

キーワード仕様	エラー
MNUBARHC(1 PULLFILE '>File')	複数の選択項目に対して同じ
MNUBARHC(2 PULLFIN '>finish')	略号が指定されている。
CHOICE(1 'S>ave >As...')	2 つの略号文字が
CHOICE(4 'X >>>= >1')	指定されている。

図 76. 略号が無効である DDS の例

大文字または小文字のいずれかの略号文字をタイプ入力することにより、略号を選択できます。これは、すべての言語において当てはまります。システムは、該当する言語用の上段専用規則を使用して、入力した略号を識別します。2 バイト文字は略号にすることはできません。

システムは、フィールドでの略号と数字両方の選択はサポートしないため、略号は、選択項目が数字を使用して表示されない場合のみ作動します。

どの構成が略号をサポートするかを判別するには、151 ページの表 15を参照してください。

選択項目のカラーおよび属性の定義

メニュー・バー選択項目、選択フィールド選択項目、選択リスト選択項目、および押しボタン選択項目を表示する際、システムは通常、共通ユーザー・アクセス® (CUA) のデフォルトのカラーおよび表示属性を使用します。これらのデフォルトを使用したくない場合、CHCAVAIL、CHCUNAVAIL、および CHCSLT キーワードを使用して、ほとんどの選択項目に対して使用されるカラーおよび属性を定義することができます。表 19は、各グラフィカル項目においてどのキーワードを使用できるかを示しています。

表 19. カラーおよび表示属性を定義するために使用されるキーワード

グラフィカル項目	DDS キーワード		
	CHCAVAIL	CHCUNAVAIL	CHCSLT
メニュー・バー選択項目	○		○
選択フィールド	○	○	○
押しボタン	○	○	
単一項目選択リスト	○	○	○
複数項目選択リスト	○	○	○

非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されているディスプレイ装置上の選択カーソルは、各選択項目の状態に指定したカラーおよび属性の反転表示形式を使用します。たとえば、選択可能な選択項目にピンク (通常表示) を指定した場合、選択カーソルが選択可能な選択項目に置かれたときには選択カーソルは反転表示のピンクになります。選択可能な選択項目にピンク反転表示を指定した場合、選択カーソルはピンク (通常表示) になります。同様に、CHCAVAIL、CHCUNAVAIL、または CHCSLT を指定せず、CUA のデフォルトのカラーおよび属性を使用した場合、選択カーソルはそれらのカラーおよび属性の反転表示となります。

メニュー・バー選択項目は選択可能または選択済みのいずれかであるため、メニュー・バーでは CHCAVAIL キーワードおよび CHCSLT キーワードのみを使用することができます。選択フィールドでは、選択文字 (番号やラジオ・ボタンなど) を使用している場合には、CHCAVAIL キーワードおよび CHCUNAVAIL キーワードのみを使用してください。CHCSLT キーワードはこれらの場合、無視されません。ただし、プルダウン・メニューが選択標識を含まない (PULLDOWN(*NOSLTIND) の指定) という指定をした場合、グラフィカル・ディスプレイ装置または文字ベースのグラフィカル・ディスプレイ装置上のプルダウン・メニュー内の選択フィールドに対して CHCSLT を使用することができます。

押しボタン選択項目は選択可能または選択不可のいずれかであるため、押しボタンでは CHCAVAIL キーワードおよび CHCUNAVAIL キーワードのみを使用することができます。

表示属性 (*DSPATR) パラメーターは、区切り文字を表示する方法を指定します。パラメーターは次の形式で表記されます。

(*DSPATR value1 <value2 <value3...>>)

表示属性の有効値は以下のとおりです。

値	意味
BL	明滅
CS	桁の区切り記号
HI	高輝度
ND	非表示
RI	反転表示
UL	下線

モノクロ・ディスプレイ装置上の選択フィールド内の選択不可能な項目に対するデフォルト表示属性は、標準 (または低) 輝度です。さらに、モノクロ・ディスプレイ装置上にある選択不可能な項目の先頭文字は、アスタリスク (*) で上書きされます。

表示属性 CS、HI、および BL では、5292、3179、および 3197 の C1 および C2 型ディスプレイ装置上のフィールドがカラー・フィールドとして表示される場合があります。表示属性 HI、RI、および UL では、区切り行が表示されなくなります。詳しくは、iSeries Information Center の **DDS** トピックで、CHCAVAIL キーワード、CHCUNAVAIL キーワード、および CHCSLT キーワードを参照してください。

図 77では、メニュー・バー内の選択項目は、カラー・ディスプレイ装置では白で、モノクロ・ディスプレイ装置では高輝度で表示されます。メニュー・バー選択項目が選択されると、それはカラー・ディスプレイ装置では緑で表示され、モノクロ・ディスプレイ装置では標準 (高輝度ではない) 表示に戻ります。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          R MNUBAR          MNUBAR
A          MNUFLD          2Y 0B 1 2
A
A          MNUBARHC(1 PULLFILE +
A          '>File          ')
A 02      MNUBARHC(2 PULLEDIT +
A          &EDITTXT)
A          MNUBARHC(3 PULLVIEW +
A          '>View          ')
A 04      MNUBARHC(4 PULLOPT +
A          '>Options          ')
A          MNUBARHC(5 PULLHELP +
A          '>Help          ')
A          MNUBARSEP((*COLOR WHT))
A          CHCAVAIL((*COLOR WHT) (*DSPATR HI))
A          CHCSLT((*COLOR GRN))
A          EDITTXT          20A P
          .
          .
          .
```

図 77. メニュー・バー選択項目のために CHCAVAIL および CHCSLT を使用する DDS

189 ページの図 78では、選択可能な選択フィールド選択項目は、カラー・ディスプレイ装置ではピンクで、モノクロ・ディスプレイ装置では高輝度で表示されます。選択不可の選択フィールド選択項目は、カラー・ディスプレイ装置では青緑色で表示されます。モノクロ・ディスプレイ装置では、選択不可能な項目は標準 (または低) 輝度で表示され、選択項目内の先頭文字はアスタリスク (*)で上書きされます。

```

|...+...1...+...2...+...3...+...4...+...5...+...6...+...7...
A          F1          2Y 0B 1 1SNGCHCFLD
A    01          CHOICE(1 '>Chocolate ')
A          CHOICE(2 '>Strawberry ')
A          CHOICE(3 '>Vanilla ')
A          CHCCTL(1 &CTLCHOC MSG1112 QUSER/A)
A          CHCCTL(2 &CTLSTRA &MSG &LIB/&MSGF)
A          CHCCTL(3 &CTLVANI)
A          CHCAVAIL((*COLOR PNK) (*DSPATR HI))
A          CHCUNAVAIL((*COLOR TRQ))
A          CTLCHOC          1Y 0H
A          CTLSTRA          1Y 0H
A          CTLVANI          1Y 0H
A          MSG              7A P
A          LIB              10A P
A          MSGF             10A P

```

図 78. 選択フィールドのために *CHCAVAIL* および *CHCUNAVAIL* を使用する DDS

図 79では、選択された選択フィールド選択項目は、黄色で表示されます。このフィールドに *CHECK(ER)* (自動入力) が指定されている場合、制御は選択項目の選択直後に戻ります。

```

|...+...1...+...2...+...3...+...4...+...5...+...6...+...7...
A          R PULLDOWN          PULLDOWN(*NOSLTIND)
A          F1          2Y 0B 1 1SNGCHCFLD
A    01          CHOICE(1 '>Undo ')
A          CHOICE(2 '>Mark ')
A          CHOICE(3 '>Copy ')
A          CHCCTL(1 &CTLUNDO MSG1112 QUSER/A)
A          CHCCTL(2 &CTLMARK &MSG &LIB/&MSGF)
A          CHCCTL(3 &CTLCOPY)
A          CHCAVAIL((*COLOR PNK) (*DSPATR HI))
A          CHCUNAVAIL((*COLOR TRQ))
A          CHCSLT((*COLOR YLW))
A          CTLUNDO          1Y 0H
A          CTLMARK          1Y 0H
A          CTCOPY           1Y 0H
A          .
A          .
A          .

```

図 79. 選択フィールドのために *CHCAVAIL*、*CHCUNAVAIL*、および *CHCSLT* を使用する DDS

図 80には、単一項目選択リスト用 DDS ソースが示されています (この例には正確なサブファイル定義のために必要なすべてのキーワードが示されているわけではありません)。リスト内で選択可能な選択項目は黄色で表示されます。選択不可能な項目は赤で表示されます。選択済み選択項目は緑で表示されます。

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...

```
A      R SFLREC      SFL
A      CTLFLD      1Y 0H  SFLCHCCTL
A      R SFLCTLRCD  SFLCTL(SFLREC)
A      SFLSNGCHC
A      .
A      .
A      .
A      CHCAVAIL((*COLOR YLW))
A      CHCUNAVAIL((*COLOR RED))
A      CHCSLT((*COLOR GRN))
```

図 80. 単一項目選択リスト選択項目のために CHCAVAIL、CHCUNAVAIL、および CHCSLT を使用する DDS

図 81には、複数項目選択リスト用 DDS ソースが示されています (この例には正確なサブファイル定義のために必要なすべてのキーワードが示されているわけではありません)。リスト内で選択可能な選択項目は黄色で表示されます。選択不可能な項目は赤で表示されます。すべての選択済み選択項目は緑で表示されず。

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...

```
A      R SFLREC      SFL
A      CTLFLD      1Y 0H  SFLCHCCTL
A      R SFLCTLRCD  SFLCTL(SFLREC)
A      SFLMLTCHC
A      .
A      .
A      .
A      CHCAVAIL((*COLOR YLW))
A      CHCUNAVAIL((*COLOR RED))
A      CHCSLT((*COLOR GRN))
```

図 81. 複数項目選択リスト選択項目のために CHCAVAIL、CHCUNAVAIL、および CHCSLT を使用する DDS

継続入力フィールドの概要

継続入力フィールドとは、関連付けられた入力フィールドのセットのことです。継続入力フィールドは、いずれの制御装置に接続されたディスプレイでもサポートされます。非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置は、データが入力されフィールド内で編集されるときに、継続入力フィールドを単一入力フィールドとして処理します。

注: 非プログラマブル・ワークステーション用の拡張インターフェースをサポートしない制御装置は、継続入力フィールドを別々の入力フィールドとして処理します。文字の挿入および削除は、一度に 1 つのセグメントだけ行ってください。セグメントの末尾に到達しても、カーソルは次のセグメントに自動的に移動しません。

191 ページの図 82は、継続入力フィールドを使用して長方形のテキスト入力フィールドを作成する方法を示しています。複数の行にまたがって折り返される単一の入力フィールドを使用しないように、この形式の使用を考慮してください。

注: 最終行の末尾にある空のスペースも、継続入力フィールドの一部です。このスペースに別のフィールドを定義することはできません。

入力テキスト . . . _____

図 82. 長方形に配置された継続入力フィールド

継続入力フィールドでは、複数行の入力フィールドをウィンドウまたはディスプレイの内部に定義することが可能になります。

フィールド用 DDS は以下のようになります。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
      A           F1           90A B 3 4CNTFLD(30)
```

CNTFLD キーワードはこれが継続入力フィールドであることをシステムに通知し、そのパラメータはフィールドの幅をシステムに通知します。システムはフィールドを列に分割し、指定された幅に到達するために必要なだけの行数を使用します。

継続入力フィールドでの語の折り返しの指定

継続入力フィールドで語の折り返し機能を指定するには、語の折り返し (WRDWRAP) キーワードを使用します。このキーワードは、ファイル、レコード、またはフィールドの各レベルで使用できます。

注: WRDWRAP は DBCS 継続入力フィールドでは使用できません。

WRDWRAP キーワードの詳細については、35 ページの『フィールドの語の折り返しの指定』を参照してください。

継続入力フィールドでの DBCS に関する考慮事項

DDS は、以下の DBCS データ・タイプをサポートします。

- J** 専用 (シフト文字付き DBCS のみが許可されます)
- E** 択一 (SBCS のみ、またはシフト文字付き DBCS 文字のみのいずれかが許可されます)
- O** 混用 (SBCS またはシフト文字付き DBCS 文字のいずれかが許可されます - 混合)
- G** グラフィック/純粋 (シフト文字なし DBCS 文字のみ許可されます)

これらのデータ・タイプには以下の制約事項があります。

- J** それぞれの継続入力フィールド・セグメントの幅は、4 バイト以上の偶数でなければなりません。
- E** それぞれの継続入力フィールド・セグメントの幅は、4 バイト以上の偶数でなければなりません。
- O** それぞれの継続入力フィールド・セグメントの幅は、4 バイト以上の幅でなければなりません。
- G** それぞれの継続入力フィールド・セグメントの幅は、4 バイト以上の偶数でなければなりません。

DBCS 継続入力フィールドの長さは、継続入力フィールドの各セグメントの DBCS データにシフト文字を付ける SO/SI 文字のペアを考慮に入れる必要があります。フィールド・データを DBCS 継続入力フィールドに確実に収めるためには、以下の合計フィールド長が必要です。

J または E (DBCS データあり)

データ長 + (セグメント数 - 1) * 2

- O** データ長 + (セグメント数 - 1) * 3

G または E (SBCS データあり)

データ長

注: 2 番目の式の計算における (セグメント数 - 1) * 3 の部分は、最初のセグメントの後にある継続入力フィールドのセグメント上の DBCS データにシフト文字を付ける必要がある SO/SI セットの分を考慮に入れています。DBCS 文字が分割される場合には必ずセグメントの末尾に NULL を置く必要があるという可能性についても、さらに考慮事項として含まれています。

継続入力フィールドの DBCS データを戻す方法

フィールドが DBCS 専用または DBCS 択一 (DBCS データあり) フィールドである場合、次のデータはアプリケーションに戻されません。

- 中間および最後のセグメントの先頭にある余分の SO 文字
- 最初および中間のセグメントの末尾にある余分の SI 文字

フィールドが DBCS 混用フィールドである場合、フィールドがアプリケーションに戻される前に以下のものが除去されます。

- セグメントの末尾にある、1 つのヌルのみで構成されるか空であるすべての 1 バイト・サブフィールド。これは、セグメント境界に 2 バイト・データが入った結果として自動的に挿入されたと想定される、すべての SO/SI ペアを除去するためです。余分の SI/SO 文字の除去は、制御装置が非プログラマブル・ワークステーション用の拡張インターフェースをサポートするかどうかにかかわらず、すべてのハードウェアにおいて実行されます。
- 以下が該当する場合にセグメントを終了させる 1 バイト・ヌル
 - ヌルの数が 3 つ以下である
 - 直前の文字が SI 文字またはヌルではない
 - 次のセグメントが SO 文字で始まる

これは、セグメント境界に 2 バイト・データが入った結果として自動的に挿入されたと想定される、すべてのヌルを除去するためです。

継続入力フィールドでのキーボード機能

システムは、以下の定義により、非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されているディスプレイ装置上の継続入力フィールド内で指定されたローカル・キーボード機能を処理します。

文字データ

置換モードでは、固有の文字データ処理はありません。文字データが、セット内の最初またはいずれかの中間のフィールドの最後の文字位置に入力されると、カーソルはセット内の次のフィールドの最初の文字位置に移動します。文字データが、継続フィールド内の最後のフィールドの最後の文字位置に入力されると、前方フィールド出口処理が実行されます (196 ページの『前方フィールド出口処理』を参照してください)。

挿入モードでカーソルの方向がフィールドの方向と一致する場合、文字データ・キーが押されると以下のアクションが実行されます。

- 継続入力フィールド内の最後のフィールドの最後の文字位置にあるヌルは削除されます。このフィールドの最後の位置にヌルがない場合、オペレーター・エラー 0012 が通知されます。
- 現行カーソル位置 (および論理的にその後) にある継続入力フィールド内のすべてのフィールド・データは、位置が 1 つ分桁送りされます。セット内の現行および残りのフィールド (最後を除く) の最後の文字位置にある各データ文字は、次のフィールドの最初の文字位置に桁送りされます。

- 入力されるデータ文字はカーソル位置に書き込まれます。
- カーソルは次のカーソル位置に進みます。
- DBCS 専用、DBCS 択一 (DBCS データあり)、および DBCS 純粹の各フィールドでは、継続フィールドの最後のセグメントの最後の 2 つの文字位置 (DBCS 専用または DBCS 択一フィールドのための SI の前) にある 2 つのヌルは削除されます。そのセグメントの最後の 2 つの位置にヌルがない場合、オペレーター・エラー 0012 が通知されます。それ以外の場合、現行カーソル位置および論理的にその後にあるフィールド・セグメントのセット内のすべてのフィールド・データ (SO または SI 文字は含まない) は、カーソルの方向に位置が 2 つ分析送りされます。現行および残りのセグメント (最後を除く) の最後の 2 つの文字位置にある各 2 バイト文字は、次のセグメントの最初の 2 つの文字位置に移動します。入力される DBCS 文字はカーソル位置に書き込まれ、カーソルは次の 2 バイト・カーソル位置に進みます。
- DBCS 混用フィールドでは、継続フィールド・セグメント内のカーソル位置およびその右方にあるデータは、1 つの連続バッファーにコピーされます。挿入された文字はバッファーの開始位置に置かれます。1 バイトのヌルはすべてバッファーから除去され、データはバッファーの先頭方向に桁送りされます。隣接したすべての SI/SO 文字のペア (すなわち、空の 1 バイト・サブフィールド) もバッファーから除去されます。データは再度、バッファーの先頭方向に桁送りされます。そのときデータは、DBCS 混用継続フィールドへのデータ書き込みのアルゴリズムに従って、一度に 1 文字ずつ継続フィールドに戻されます。残った文字位置はヌルに置換されます。さらにカーソルは、次の文字位置に進みます。バッファー内のすべてのデータが継続フィールドに収まらない場合、オペレーター・エラー 0012 が通知され、フィールド・データおよびカーソル位置は変更されません。

挿入モードでカーソルの方向がフィールドの方向と一致しない場合、挿入はサブフィールド内で実行されません。挿入はフィールドとカーソルの方向が一致した場合と同様ですが、挿入はサブフィールド内で実行されます。サブフィールドの範囲は以下のように定義されます。

- カーソルの方向が右から左である場合、サブフィールドは、カーソルから、論理的にカーソルの後にある最初のヌルまでとなります。そうしたヌルがない場合、サブフィールドは、論理的にカーソルの後にあるすべての位置を含みます。
- カーソルの方向が左から右である場合、サブフィールドは、カーソルから、論理的にカーソルの後にある最初のヌルまでとなります。そうしたヌルがない場合、サブフィールドは、論理的にカーソルの後にあるすべての位置を含みます。

フィールド・マーク

文字データと同様に処理されます。

自動形状判別 (ASD) 処理

アラビア語では、カーソルの方向が右から左の場合に ASD が実行されます。

削除

継続入力フィールド内で削除キーが押され、カーソルの方向がフィールドの方向と一致する場合、以下のアクションが実行されます。

- 論理的に現行カーソル位置の後にある継続入力フィールド内のすべてのフィールド・データは、カーソルの方向に位置が 1 つ分析送りされます。セット内の残りのフィールドの最初の文字位置にある各データ文字は、直前のフィールドの最後の文字位置に桁送りされます。
- 継続入力フィールド内の最後の文字位置にヌルが書き込まれます。
- DBCS 専用、DBCS 択一 (DBCS データあり)、および DBCS 純粹の各継続フィールドでは、論理的に現行カーソル位置の後にあるフィールド・セグメントのセット内のすべてのフィールド・データ (SO および SI 文字は含まない) は、カーソルの方向に位置が 2 つ分析送りされます。残りのセグメントの最

初の 2 バイト文字位置にある各 DBCS 文字は、直前のセグメントの最後の 2 バイト文字位置に移動します。2 バイトのヌルが、最後のセグメントの最後の 2 バイト文字位置に書き込まれます。

- DBCS 混用フィールドでは、フィールド・セグメントのセット内のカーソル位置およびその右方にあるデータは、1 つの連続バッファにコピーされます。削除された文字またはサブフィールドは、バッファの開始位置から除去されます。さらに、1 バイトのヌルはすべてバッファから除去され、データはバッファの先頭方向に桁送りされます。隣接したすべての SI/SO 文字のペア (すなわち、空の 1 バイト・サブフィールド) もバッファから除去されます。データは再度、バッファの先頭方向に桁送りされます。そのとき残りのデータは、DBCS 混用継続フィールドへのデータ書き込みのアルゴリズムに従って、一度に 1 文字ずつ継続フィールドに戻されます。残った文字位置はヌルに置換されます。

カーソルの方向とフィールドの方向が一致しないときに継続フィールド内で削除キーが押された場合、削除はサブフィールド内で実行されます。サブフィールドの定義は挿入キーの場合と同じです。

EOF 消去

継続入力フィールド内にある現行カーソル位置 (および論理的にその後) にあるすべてのフィールド位置は、ヌルになります。DBCS 専用、DBCS 択一 (DBCS データあり)、および DBCS 純粹の各フィールドでは、SO および SI 文字はヌルになりません。DBCS 混用フィールドでは、消去が 2 バイト・サブフィールドで開始された場合には、現行カーソル位置に SI 文字が余分に書き込まれる可能性があります。

入力消去

変更されたすべてのフィールドのすべてのフィールド位置はヌルになります。これには、いずれかの継続入力フィールド・セグメントが変更された場合には、すべての継続入力フィールド・セグメントが含まれます。DBCS 専用、DBCS 択一 (DBCS データあり)、および DBCS 純粹の各フィールドでは、SO および SI 文字はヌルになりません。

反転

カーソルの方向が反転します。直前のキー・ストロークがカーソル移動キーではなかった場合、カーソルは現行セグメントの新しい最初の文字位置に位置変更されます。

クローズ

クローズ・キーは、単一の継続フィールド・セグメント上で作動します。埋め込まれたすべてのヌルは除去されます。カーソルの方向はフィールドの方向に設定されます。残りの文字は、継続フィールド・セグメントの最初の文字位置から開始するように桁送りされます。セグメントの残りにはヌルが埋め込まれ、カーソルは論理的に最後の非ヌル文字の後に置かれます。

フィールド・エグジット

継続入力フィールド内でフィールド・エグジット・キーを押すと、以下のアクションが実行されます。

- 現行カーソル位置 (および論理的にその後) にある継続入力フィールド内のすべてのフィールド・データは、ヌルになります。DBCS 専用、DBCS 択一 (DBCS データあり)、および DBCS 純粹の各フィールドでは、SO および SI 文字はヌルになりません。DBCS 混用フィールドでは、ヌルの書き込みが 2 バイト・サブフィールドで開始された場合には、現行カーソル位置に SI 文字が余分に書き込まれる可能性があります。
- 前方フィールド出口処理が実行されます (196 ページの『前方フィールド出口処理』を参照してください)。

フィールド・プラス

フィールド・エグジットと同様に処理されます。

フィールド・マイナス

許可されていません。オペレーター・エラー 0016 が通知されます。

DUP (複写)

継続入力フィールド内で複写キーを押すと、以下のアクションが実行されます。

- 現行カーソル位置 (および論理的にその後) にある継続入力フィールド内のすべてのフィールド・データは、DUP 文字 (1C) に設定されます。DBCS 専用、DBCS 択一 (DBCS データあり)、および DBCS 純粋の各フィールドでは、SO および SI 文字は DUP 文字に置換されません。DBCS 混用フィールドでは、複写キーを押す時には、カーソルは先頭文字に置かれていなければなりません (その文字が 1 バイト文字か 2 バイト文字かにかかわらず当てはまります)。すべてのセグメントの文字は、SO および SI 文字も含めてすべて DUP 文字 ('1C') に置換されます。カーソルが先頭文字にない場合には、オペレーター・エラー 0019 が通知されます。
- 前方フィールド出口処理が実行されます (196 ページの『前方フィールド出口処理』を参照してください)。

漢字

DBCS 択一継続フィールド内で漢字キーを押すと、以下のアクションが実行されます。

- カーソルが最初のセグメントの最初のフィールド位置 (1 バイト・モードの場合)、または最初のセグメントの 2 番目のフィールド位置 (2 バイト・モードの場合) にない場合、オペレーター・エラー 0062 が通知されます。
- それ以外の場合、フィールドが現在 2 バイト・モードである場合には、すべてのセグメントのすべての文字位置をヌルに置換することにより、フィールドは 1 バイト・モードに置き換えられます。さらにカーソルは、最初のセグメントの最初のフィールド位置に置かれます。
- それ以外の場合、フィールドが現在 1 バイト・モードである場合には、すべてのセグメントのすべての文字位置をヌルに置換し、各フィールド・セグメントの先頭と末尾にそれぞれ SO および SI 文字を書き込むことにより、フィールドは 2 バイト・モードに置き換えられます。さらに、カーソルは最初のセグメントのシフトアウト文字の直後に置かれます。

DBCS 混用継続フィールドでは、漢字キーにより、現在非継続混用フィールドに対して行われているように、SO/SI 文字のペア、または SI/SO 文字のペアのいずれかが挿入されます。継続フィールドの場合、この挿入は、文字が DBCS 混用継続フィールドに挿入される時と同じアルゴリズムを使用して実行されます。ただし、SI/SO ペアが挿入される時、作成される空の 1 バイト・サブフィールドは即時に除去されません。さらにカーソルは 2 番目のシフト文字の下に置かれ、キーボードは挿入モードになります。

文字後退

最初のセグメントの最初の位置 (または DBCS 純粋、DBCS 専用、または DBCS 択一のいずれかのフィールドの最初の DBCS 文字) で文字後退キーを押すと、カーソルは直前のフィールドの最後の位置に移動します。(直前のフィールドは継続入力フィールドである場合があります。カーソルは、画面のずっと下の方であったとしても、最後のセグメントの最後の位置に移動します。) 移動後のカーソル位置が DBCS 混用、DBCS 専用、または DBCS 択一のフィールドにある場合、最後の位置にあるすべての SI 文字はスキップされます。最初のセグメント以外のセグメントの最初の位置 (または 2 バイト・モードの DBCS 純粋、DBCS 専用、または DBCS 択一のフィールドの最初の DBCS 文字) で文字後退キーを押すと、カーソルは直前のセグメントの最後の位置に移動します。その移動後の位置が DBCS 専用または DBCS 択一フィールドの SI 文字上である場合には、その SI 文字はスキップされます。さらに、DBCS 混用フィールド内において、最後ではないセグメントの末尾にある 1 バイト・サブフィールドは、1 つのヌル文字だけで構成されている場合には文字後退によりスキップされます。これは、セグメント境界にある DBCS データを分割するときに自動的に挿入されたと想定されるヌルをスキップするためです。

文字前進

最後のセグメントの最後の位置 (または 2 バイト・モードの DBCS 純粋、DBCS 専用、または DBCS 択一のフィールドの最後の 2 バイト文字) で文字前進キーを押すと、カーソルは次のフィールドの最初の位置に移動します。最後のセグメント以外のセグメントの最後の位置 (または 2 バイト・モードの DBCS 純粋、DBCS 専用、または DBCS 択一のフィールドの最後の 2 バイト文字) で文字前進キーを押すと、カーソルは次のセグメントの最初の位置に移動します。その移動後の位置が DBCS 専用または DBCS 択一フィールドの SO 文字上である場合には、その SO 文字はスキップされます。DBCS 混用フィールド内において、セグメントの末尾にある 1 バイト・サブフィールドは、1 つのヌルだけで構成されている場合にはスキップされます。これは、セグメント境界にある DBCS データを分割するときに自動的に挿入されたと想定されるヌルをスキップするためです。

改行

改行キーを押すと、カーソルは通常、画面上でカーソルを配置可能な次の位置に移動します。カーソルが継続入力フィールドにあり、追加の継続入力フィールド・セグメントが次の行または後続の行にある場合、カーソルはそのセグメントの最初の位置に移動します。さらにその継続入力フィールドが、不可視のテキスト・カーソルを持つ強調表示フィールドである場合には、改行キーを押すとその継続入力フィールドから出ます。移動後の位置が、同じフィールドの、DBCS 混用、DBCS 専用、または DBCS 択一のいずれかの新規フィールド・セグメントにある場合、移動後のカーソル位置にあるすべての SO 文字はスキップされます。カーソルが、改行キーが押されたために継続入力フィールドに移動する場合、カーソルは常に最初のセグメントの最初の位置に置かれます。改行キーを押して、カーソルが中間または最後の継続入力フィールド・セグメントに移動することは決してありません。

フィールド前進

フィールド前進は、前方フィールド出口処理を実行します (『前方フィールド出口処理』を参照してください)。

フィールド後退

最初のセグメントの最初の位置 (または DBCS 純粋、DBCS 専用、または DBCS 択一のフィールドの最初の DBCS 文字) にカーソルがないときに押すと、カーソルは最初のセグメントの最初の位置 (または DBCS 純粋、DBCS 専用、または DBCS 択一のフィールドの最初の DBCS 文字) に移動します。それ以外の場合、フィールド後退は、後方フィールド出口処理を実行します (197 ページの『後方フィールド出口処理』を参照してください)。

前方フィールド出口処理

全桁入力および自己検査機能がサポートされていないためにカーソルが継続入力フィールドを出る場合、システムはフィールド・データを妥当性検査しません。

継続入力フィールドが、自動入力または前方エッジ・トリガーとして指定されている場合、システムは最後のセグメントの最後の位置に対して自動入力または前方エッジ・トリガー処理を実行します。

カーソル進行が最初の継続入力フィールド・セグメントで指定されている場合、カーソルがいずれかの継続入力フィールド・セグメントを出る場合には、それはカーソル進行ターゲット・フィールドに移動します。

カーソル進行が最初の継続入力フィールド・セグメントで指定されておらず、カーソルが継続入力フィールドから前方方向に出る場合、カーソルは継続入力フィールドの後続のセグメントをすべてスキップして次の無保護フィールドに移動します。次の無保護フィールドは、どのセグメントにカーソルが含まれていたかに

関係なく、最初の継続フィールド・セグメントを出ることで判別されます。移動後の位置が DBCS 混用、DBCS 専用、または DBCS 択一のフィールドにある場合、最初の位置にあるすべての SO 文字はスキップされます。カーソルは 2 番目の位置に置かれます。

注: 継続入力フィールドが強調表示フィールドとしても定義されている場合、カーソルがフィールドを出るときに、システムは各セグメントの先行フィールド属性を復元します。

後方フィールド出口処理

フィールドを後方に出る場合、カーソルは、継続入力フィールドの前のセグメントをすべてスキップし、直前の無保護フィールドに移動します。カーソル進行はカーソルを別のフィールドに移動させる可能性があります。直前の無保護フィールドは、どのセグメントにカーソルが含まれていたかに関係なく、最初の継続フィールド・セグメントを出ることで判別されます。移動後の位置が DBCS 混用、DBCS 専用、または DBCS 択一のフィールドにある場合、最初の位置にあるすべての SO 文字はスキップされます。カーソルは 2 番目の位置に置かれます。

注: 継続入力フィールド内の DBCS サポートは、任意の制御装置に接続されたディスプレイにおいて使用可能です。ただし、キーボード機能は、非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されているディスプレイのみ使用可能です。

メニュー・バーがアプリケーションと相互作用する方法

メニュー・バーを表示するために MNUBARDSP キーワードが使用されます。MNUBARDSP キーワードは、アプリケーション・レコード (アプリケーション表示を定義するレコード) またはメニュー・バー・レコード上で使用することができます。オプション標識を MNUBARDSP キーワードで使用すると、メニュー・バーを表示する時期を制御できます。MNUBARDSP をアプリケーション・レコードで使用する場合、オプションが付いたいくつかの MNUBARDSP キーワードを指定することにより、アプリケーションは、同じレコードに対して異なるメニュー・バーを表示させることができます。複数の MNUBARDSP キーワードが有効である場合、システムは最初のものを使用します。

アプリケーション・レコードでの MNUBARDSP キーワードの定義

アプリケーション表示を定義するレコードで MNUBARDSP キーワードが使用された場合、システムはアプリケーションのためにすべてのメニュー・バー操作を処理します。システムは、選択されたメニュー・バー選択項目の番号を、MNUBARDSP キーワードで指定されアプリケーション・レコードで定義された隠しフィールドに戻します。メニュー・バー選択項目が選択されていない場合、0 が隠しフィールドに戻されません。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          HELP ALTHELP CF03
A          MNUBARSW(CA10) MNUCNL(CA12)
A          R MNUBAR          MNUBAR
A          MNUFLD          2Y 0B 1 2
A          MNUBARHC(1 PULLFILE +
A          '>File          ')
A 02       MNUBARHC(2 PULLEDIT +
A          &EDITTXT)
A          MNUBARHC(3 PULLVIEW +
A          '>View          ')
A 04       MNUBARHC(4 PULLOPT +
A          '>Options        ')
A          MNUBARHC(5 PULLHELP +
A          '>Help          ')
A          :
A          :
A          R APPSCR          MNUBARDSP(MENUBAR &MNUCHOICE)
A          FIELD1          10A B 10 12
A          FIELD2          5S 0B 14 12
A          24 1'F1=Help   F3=Exit   +
A          F10=Actions   F12=Cancel '
A          MNUCHOICE          2Y 0H

```

図 83. アプリケーション・レコード上で MNUBARDSP を使用する DDS

図 83の DDS では、以下の処理が実行されます。

1. アプリケーションは、出力がアプリケーションにより提供されるすべてのプルダウン・メニュー・レコードまたはメニュー・バー・レコードを書き込みます。
2. アプリケーションは、MNUBARDSP キーワードが含まれるレコードに対して、書き込み/読み取り操作を行います。
3. システムは、MNUBARDSP キーワードにより識別されたメニュー・バーを表示し、メニュー・バーとプルダウン・メニューの間のすべての相互作用を処理します。
4. ユーザーがあるメニュー・バー選択項目を選択し、プルダウン・メニューに入力データを入力した場合、システムは選択されたメニュー・バー選択項目の番号を、MNUBARDSP で指定された選択項目隠しフィールドに戻します。

システムは、最初にメニュー・バー・レコードを表示し、その後アプリケーション・レコードに対して書き込み/読み取り操作を実行します。活動状態の機能キーおよびコマンド・キーは、アプリケーション・レコードで定義されたもので、メニュー・バー・レコードで定義されたものではありません。MNUCNL キーワードまたは MNUBARSW キーワードを使用したい場合には、ファイル・レベルで、またはアプリケーション・レコードでそれらを定義してください。

通常、OVERLAY キーワードを使用せずに画面にレコードを書き込むと、レコードが表示される前に画面全体が消去されてしまいます。システムは、メニュー・バー・レコードとアプリケーション・レコードが論理的に 1 つのレコードであるかのように、メニュー・バー・レコードの続きにアプリケーション・レコードを表示します。システムは、アプリケーション・レコードがメニュー・バー・レコードを消去することを自動的に防ぎます。図 83では、システムがアプリケーション・レコードを書き込む際、システムはメニュー・バー・レコードを除いて画面全体を消去します。メニュー・バー・レコードは、他の仕様に關係なく、OVERLAY キーワードが含まれている場合のように常に処理されます。たとえば、メニュー・バー・レコードに CLRL キーワードが指定されている場合、それはメニュー・バー・レコードが処理される時には使用されません。OVERLAY キーワードおよび CLRL キーワードはアプリケーション・レコードでは通常通り処理されます。ただし、メニュー・バー・レコードは消去されません。

メニュー・バー・レコードでの MNUBARDSP キーワードの定義

メニュー・バーが含まれるレコードで MNUBARDSP キーワードを使用する場合、アプリケーションは、メニュー・バーが表示される時期を制御します。システムは、発生するすべてのプルダウン・メニュー相互作用の処理を引き続き行い、選択されたメニュー・バー選択項目の番号をメニュー・バー・フィールド自体に戻します。メニュー・バー選択項目が選択されなかった場合、0 がメニュー・バー・フィールドに戻されます。アプリケーションは、メニュー・バーを表示するためのメニュー・バー・レコードの書き込みと、どの選択項目が選択されたかを（選択された場合）判別するための読み取りの両方を行う必要があります。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     HELP ALTHelp CF03
A                                     MNUBARSW(CA10) MNUCNL(CA12)
A           R MNUBAR                   MNUBAR OVERLAY
A 01                                     MNUBARDSP
A           MNUFLD                   2Y 0B 1 2
A                                     MNUBARHC(1 PULLFILE +
A                                     '>File      ')
A 02                                     MNUBARHC(2 PULLEDIT +
A                                     &EDITTXT)
A                                     MNUBARHC(3 PULLVIEW +
A                                     '>View      ')
A 04                                     MNUBARHC(4 PULLOPT +
A                                     '>Options  ')
A                                     MNUBARHC(5 PULLHELP +
A                                     '>Help     ')
A           EDITTXT                   20A P
A           :
A           :
A           R APPSCR                   OVERLAY
A           FIELD1                   10A B 10 12
A           FIELD2                   5S 0B 14 12
A                                     24 1'F1=Help  F3=Exit  +
A                                     F10=Actions F12=Cancel '
```

図 84. メニュー・バー・レコード上で MNUBARDSP を使用する DDS

以下に、図 84 の DDS を使用したシナリオの 1 つが示されています。

1. アプリケーションは、MNUBARDSP キーワードを活動状態にしてメニュー・バー・レコードを書き込み、メニュー・バーを表示させます。
2. アプリケーションは、アプリケーション・レコードに対して書き込み/読み取り操作を行います。
3. システムは、メニュー・バーとプルダウン・メニューの間の相互作用を処理します。
4. 入力で、アプリケーションはアプリケーション・レコードを受け取ります。
5. アプリケーションは、メニュー・バー・レコードを読み取り、どのメニュー・バー選択項目が選択されたかを（選択された場合）判別します。

活動状態のコマンド・キーおよび機能キーは、アプリケーション・レコードで定義されたものだけで、メニュー・バー・レコードで定義されたものではありません。MNUCNL キーワードまたは MNUBARSW キーワードを使用したい場合には、ファイル・レベルで、またはアプリケーション・レコードでそれらを定義してください。

以下に、図 84 の DDS を使用した別のシナリオが示されています。

1. アプリケーションは、アプリケーション・レコードを書き込みます。
2. アプリケーションは、MNUBARDSP キーワードを活動状態にしてメニュー・バー・レコードに対して書き込み/読み取り操作を実行し、メニュー・バーを表示させます。

3. システムは、メニュー・バーとプルダウン・メニューの間の相互作用を処理します。
4. 入力で、アプリケーションはメニュー・バー・レコードを受け取り、どのメニュー・バー選択項目が選択されたかを (選択された場合) 判別します。
5. アプリケーションは、画面に入力された入力データを受け取るためにアプリケーション・レコードを読み取ることもできます。

有効なコマンド・キーおよび機能キーは、メニュー・バー・レコードで定義されたものだけです。

MNUCNL キーワードまたは MNUBARSW キーワードを使用したい場合には、ファイル・レベルで、またはメニュー・バー・レコードでそれらを定義してください。

プルダウン・メニューからの入力データの受け取り

プルダウン・メニュー・レコードに出力データが含まれている場合、メニュー・バー・レコードを書き込む前にそれを書き込む必要があります。たとえば、オプション標識が設定されるかもしれず、出力フィールドへの入力があるかもしれません。これらのプルダウン・メニュー・レコードは、書き込み時には表示されません。システムは、メニュー・バーが表示されるまでレコード出力を処理および保管します。

メニュー・バー・レコードも表示せずに書き込むことができます。システムが、プルダウン・メニュー・レコードまたはメニュー・バー・レコードの (MNUBARDSP キーワードのオプションをオンにせずに) 書き込みをしようとする時に、対応するメニュー・バーが表示されている場合、エラーが発生し、レコードは書き込まれません。

プルダウン・メニューが表示されているときに有効なアテンション ID (AID) キー (取り消しキーとメニュー・バー切り替えキーを除く) を押すと、制御はアプリケーションに戻り、プルダウン・メニューから入力データを受け取ることができます。入力データを戻す有効な AID キーは、Enter キーと、プルダウン・メニュー・レコードで定義されたすべての CFxx キーです。(CAxx キーは、制御をアプリケーションに戻しますが、入力データは戻しません。)背景画面用に定義されたキーは、プルダウン・メニュー・レコードに対しても定義されているのでないかぎり、無効です。そのため、背景のキーとプルダウン・メニューのキーを、ファイル・レベルでいったん定義してください。

プルダウン・メニューで有効な AID キー (入力を戻すもの) をいったん押すと、アプリケーションは、読み取り中のレコード (アプリケーション・レコードまたはメニュー・バー・レコードのいずれか) に対する入力データを受け取ります。このレコードに戻されたメニュー・バー選択項目の番号を参照することにより、アプリケーションは、どのプルダウン・メニュー・レコードに入力データがあるかを判別できます。その時アプリケーションはそのプルダウン・メニュー・レコードを読み取る必要があります。画面に対しては入出力操作は行われません。プルダウン・メニューからの入力データがアプリケーションに戻され、プルダウン・メニューは表示されたままになります。

プルダウン入力パラメーターを使用したプルダウン・メニューからの入力データの受け取り

1 つ以上のプルダウン・メニュー・レコードに 1 つの単一項目選択フィールドしか含まれない場合、MNUBARDSP キーワード上にプルダウン入力 (PULLINPUT) パラメーターを使用できます。これにより、単一選択フィールド選択項目を受け取るためにプルダウン・メニュー・レコードを読み取る代わりに、メニュー・バー選択項目と共に、単一選択フィールド選択項目を受け取ることが可能になります。

201 ページの図 85は、MNUBARDSP キーワードでの PULLINPUT パラメーターの使用方法について示しています。


```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A      HELP ALTHELP CF03
A      MNUBARSW(CA10) MNUCNL(CA12)
A      R MNUBAR      MNUBAR
A      MNUFLD      2Y 0B 1 2
A      MNUBARCHC(1 PULLFILE +
A      '>File      ')
A      MNUBARCHC(2 PULLEDIT +
A      '>Edit      ')
A      MNUBARCHC(3 PULLVIEW +
A      '>View      ')
A      R PULLFILE      PULLDOWN
A      2 1'File name . . . '
A      FNAME      10A I 2 18
A      R PULLEDIT      PULLDOWN
A      F1      2Y 0B 1 1SNGCHCFLD
A      CHOICE(1 '>Copy      ')
A      CHOICE(2 '>Delete      ')
A      R PULLVIEW      PULLDOWN
A      F1      2Y 0B 1 1SNGCHCFLD
A      CHOICE(1 '>All      ')
A      CHOICE(2 '>Some...      ')
A      F2      2Y 0B 4 1SNGCHCFLD
A      CHOICE(1 'By >date      ')
A      CHOICE(2 'By >subject      ')
A      R APPSCR      MNUBARDSP(MNUBAR &MNUCHOICE +
A      &PULLINPUT)
A      FIELD1      10A B 10 12
A      FIELD2      5S 0B 14 12
A      24 1'F1=Help F3=Exit +
A      F10=Actions F12=Cancel '
A      MNUCHOICE      2Y 0H
A      PULLINPUT      2S 0H

```

図 85. プルダウン入力 (PULLINPUT) パラメーター用 DDS

MNUBARDSP キーワードに PULLINPUT パラメーターが指定されている場合、以下のいずれかの値がアプリケーションに戻されます。

注: 値は、MNUBARDSP キーワードでレコードに定義した隠しフィールド (図 85の PULLINPUT) に戻されます。

PULLINPUT の内容

意味

- 0** 何も選択されていません。
- n** プルダウン・メニューの選択項目 **n** が選択されました。
- 1** プルダウン・メニュー・レコードには、1 つの単一項目選択フィールド以外のデータが入っています。その内容を受け取るには、プルダウン・メニュー・レコードを読み取る必要があります。

表 20は、図 85 の DDS を使用して MNUCHOICE フィールドおよび PULLINPUT フィールドに戻される値を示しています。

表 20. MNUCHOICE および PULLINPUT に戻される値

選択されたメニュー・バー選択項目	該当するフィールドに戻される値	
	MNUCHOICE	PULLINPUT
File	1	-1
Edit (プルダウン・メニューでは選択なし)	2	0

表 20. MNUCHOICE および PULLINPUT に戻される値 (続き)

選択されたメニュー・バー選択項目	該当するフィールドに戻される値	
	MNUCHOICE	PULLINPUT
Edit (プルダウン・メニューの「Copy」選択項目が選択された)	2	1
View	3	-1

入力データを受け取った後のプルダウン・メニューの除去

アプリケーションは、プルダウン・メニューから入力データを受け取った後、特定の条件の下ではプルダウン・メニューを除去する場合があります。たとえば、(1) (同じディスプレイ・ファイル内の) ウィンドウ以外のレコード、または (2) (同じディスプレイ・ファイル内の) RMVWDW キーワードが含まれるウィンドウ・レコードのいずれかに対してアプリケーションが書き込みまたは読み取りを行う場合に、アプリケーションはプルダウン・メニューを除去します。アプリケーションが (RMVWDW が含まれない) ウィンドウ・レコードに対して書き込みまたは読み取りを行う場合には、プルダウン・メニューは画面に表示されたままになります。これにより、アプリケーション、UIM ヘルプ、またはアプリケーション・ヘルプは、プルダウン・メニューを除去しないで画面にヘルプ・ウィンドウを書き込むことが可能になります。

プルダウン・メニューを除去し、ウィンドウ以外の別のアプリケーション画面を表示する場合、アプリケーションは該当するアプリケーション・レコードを書き込み、プルダウン・メニューが除去されます。

プルダウン・メニューを除去し、(更新されている可能性がある) 現行アプリケーション画面を保持する場合、アプリケーションは現行 (または変更された) アプリケーション・レコードに対して別の書き込み/読み取り (または読み取り) 操作を実行します。

プルダウン・メニューを除去し、ウィンドウを表示する場合、アプリケーションは、RMVWDW キーワードを指定してウィンドウを書き込みます。

プルダウン・メニューを除去し、別のプログラムを呼び出す場合、アプリケーションはダミー・レコードを書き込んだ後、他のプログラムを呼び出します。

画面上のプルダウン・メニューを残して、ウィンドウ (おそらくヘルプ・ウィンドウ) を表示する場合、アプリケーションは、ウィンドウ・レコードを (RMVWDW を指定せずに) 書き込みます。

UIM ヘルプがウィンドウに表示されているときにはプルダウン・メニューは画面に表示されたままになります。アプリケーション・ヘルプ・レコードがウィンドウであり、RMVWDW キーワードが指定されていない場合には、アプリケーション・ヘルプが表示されているときにもプルダウン・メニューが画面に表示されたままになります。

表示前のプルダウン・メニューの更新

MNUBARHC キーワード上でフィールド戻しパラメーターを使用して、アプリケーションがプルダウン・メニューを更新できるようにすることができます。メニュー・バー選択項目が選択された後、プルダウン・メニューが表示される前にプルダウン・メニューが更新されます。フィールド戻しパラメーターは、選択された選択項目の番号を戻す隠しフィールドで、次の目的で使用されます。(1) 制御は、入力データがプルダウン・メニューに入力されたからではなく、プルダウン・メニューが表示される前に戻されたということを利用してアプリケーションが判別するため。(2) どのプルダウン・メニュー・レコードに対して更新および書き込みをするかをアプリケーションが判別するため。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A           R MNUBAR           MNUBAR
A           MNUFLD           2Y 0B 1 2
A
A           MNUBARHC(1 PULLFILE +
A           '>File           ' &RTNFLD)
:
:
A           MNUBARHC(4 PULLOPT +
A           '>Options           ' +
A           &RTNFLD)
:
A           RTNFLD           2Y 0H
:
A           R PULLFILE
A
A           F1           2Y 0B 1 1SNGCHCFLD
A           PULLDOWN
:
:
A           R PULLOPT
A           PULLDOWN
A           F1           2Y 0B 1 1SNGCHCFLD
:
:

```

図 86. フィールド戻しパラメーター用 DDS

図 86の DDS では、ユーザーが 1 または 4 のメニュー・バー選択項目を選択した場合、制御は、RTNFLD フィールドに設定された選択番号と共にアプリケーションに戻ります。アプリケーション・レコードのメニュー・バー・フィールドまたは選択フィールドには 0 が入り、プルダウン・メニューの入力データを受け取っていないことを示します。アプリケーションは、RTNFLD フィールドの内容を取得するためにメニュー・バー・レコードを読み取る必要があります。そのときアプリケーションはその選択項目のプルダウン・メニュー・レコードを更新し、それを書き込みます。アプリケーションは、表示を要求するためにメニュー・バー・レコードまたはアプリケーション・レコードを読み取る必要があります。プルダウン・メニューを更新するために制御が戻された後、書き込まれる次のレコードは、MNUBARHC キーワードで指定されたプルダウン・メニューでなければなりません。この例では、選択項目 1 が選択された場合、レコード PULLFILE が書き込まれなければなりません。選択項目 4 が選択された場合、レコード PULLOPT が書き込まれなければなりません。その後システムはその選択項目のプルダウン・メニューを表示し、メニュー・バーとプルダウン・メニューの相互作用の制御を再開します。プルダウン・メニューへの読み取り操作は、プルダウン・メニューへの入力データを受け取るまで許可されません。書き込み/読み取り操作は決して許可されません。

アプリケーション・ヘルプの定義

メニュー・バー選択項目、選択フィールド選択項目、名前付きフィールド、固定情報フィールド、メニュー・バー、およびプルダウン・メニューに対してアプリケーション・ヘルプを定義できます。

選択項目レベルのヘルプの定義

メニュー・バー選択項目、単一項目選択フィールド、および複数項目選択フィールドに対するヘルプを、HLPARA キーワードを使用して定義できます。204 ページの図 87は、メニュー・バー選択項目のヘルプ用 DDS コーディングの例です。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          HELP
A          R MNUBAR          MNUBAR
A          H          HLPARA(*FLD MNUFLD 1)
A          HLPARCD(FILEHLP LIB/FILE)
A          H          HLPARA(*FLD MNUFLD 2)
A          HLPARCD(EDITHLP LIB/FILE)
A          H          HLPARA(*FLD MNUFLD 3)
A          HLPARCD(VIEWHLP LIB/FILE)
A          H          HLPARA(*FLD MNUFLD 4)
A          HLPARCD(OPTHLP LIB/FILE)
A          H          HLPARA(*FLD MNUFLD 5)
A          HLPARCD(HLPHLP HLPLIB/HLPFILE)
A          MNUFLD          2Y 0B 1 2
A          MNUBARCHC(1 PULLFILE +
A          '>File          ')
A 02          MNUBARCHC(2 PULLEDIT +
A          &EDITTXT)
A          MNUBARCHC(3 PULLVIEW +
A          '>View          ')
A 04          MNUBARCHC(4 PULLOPT +
A          '>Options          ')
A          MNUBARCHC(5 PULLHELP +
A          '>Help          ')
A          EDITTXT          10 P

```

図 87. メニュー・バー選択項目のヘルプ用 DDS

図 88は、単一選択フィールド選択項目のヘルプ用 DDS コーディングの例です。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          HELP
A          R PULLEDIT          CF04 CF06
A          WDWBORDER((*CHAR '+-||+-+')
A          PULLDOWN
A          H          HLPARA(*FLD F1 1)
A          HLPARCD(UNDOHLP LIB/FILE)
A          H          HLPARA(*FLD F1 2)
A          HLPARCD(MARKHLP LIB/FILE)
A          H          HLPARA(*FLD F1 3)
A          HLPARCD(COPYHLP LIB/FILE)
A          F1          2Y 0B 1 1SNGCHCFLD
A          CHECK(ER)
A 01          CHOICE(1 '>Undo          ')
A          CHOICE(2 '>Mark          ')
A          CHOICE(3 '>Copy          ')

```

図 88. 単一選択フィールド選択項目のヘルプ用 DDS

HLPARA キーワードで *FLD 特殊値を使用すると、ヘルプ領域がフィールド用であることを示します。*FLD の後に、ヘルプを定義する対象となるフィールドの名前を指定します。フィールドの名前の後に、ヘルプを定義する対象となる選択項目の番号を指定します。

選択項目に対して指定するヘルプ領域は、選択テキストが占める領域 (それにいずれかの側に属性バイト位置を追加する) です。圧縮が実行される場合、ヘルプ領域は選択項目と共に移動します。選択項目のオプションがオフである場合、その選択項目のヘルプ領域は存在しないため、その選択項目のヘルプ仕様書のオプションもオフになります。

任意のヘルプ仕様書において、選択項目レベルのヘルプ用テキストは、DDS レコード (HLPARCD キーワード) または UIM パネル・グループ (HLPPNLGRP) を使用して定義できます。

ヘルプ仕様書は、メニュー・バー・レコードまたはプルダウン・レコードにより包含される領域内のヘルプ領域を定義します。

UIM ヘルプを使用する場合、メニュー・バー選択項目用の項目固有ヘルプには、メニュー・バー選択項目およびそのプルダウン・メニュー用のヘルプ・モジュールが含まれます。DDS を使用する場合、メニュー・バー選択項目用の項目固有ヘルプは、メニュー・バー選択項目用ヘルプです。メニュー・バー選択項目用のヘルプ領域には、テキストの両側にスペースが付加された選択項目用テキストが含まれています。選択項目と選択項目の間にある中間のスペースは、全般ヘルプ領域の一部です。

メニュー・バー用の項目固有ヘルプは、カーソルが次のいずれかの領域に置かれている場合に表示されません。

- メニュー・バー選択項目 (その選択項目用のヘルプがある場合)。
- 活動状態のヘルプ領域が含まれない活動状態のプルダウン・レコード内の任意の場所。カーソルがプルダウン・メニュー枠上にある場合、項目固有ヘルプはメニュー・バー選択項目に対して表示されます。

プルダウン・メニュー選択項目用の項目固有ヘルプは、カーソルがプルダウン・メニューの活動状態のヘルプ領域にある場合に表示されます。プルダウン・メニュー選択項目用のヘルプ領域は、番号の最初の桁から始まり、選択フィールドの枠で終わります。プルダウン・メニューとメニュー・バー用のヘルプは、画面用全般ヘルプの一部です。プルダウン・メニューが表示されているとき、プルダウン・メニューとオーバーラップする部分以外のすべてのヘルプ領域は活動状態となります。オーバーラップする領域に対するヘルプは、全般ヘルプのみで表示することができます。基本画面ヘルプ領域の一部がプルダウン・メニューとオーバーラップする場合、オーバーラップしないヘルプ領域の部分は活動状態のままとなります。

ヘルプ・リストは 2 つあります。1 つは基本画面用、もう 1 つはメニュー・バーおよびそのプルダウン・メニュー用です。メニュー・バーが表示される時、メニュー・バーのヘルプ・リストが作成されます。メニュー・バーが画面から除去される時にヘルプ・リストは破棄されます。メニュー・バーのヘルプ・リストの例については、206 ページの図 89を参照してください。メニュー・バーまたはプルダウン・メニューが再度書き込まれる時にヘルプ・リストは更新されます。カーソルがメニュー・バーまたはプルダウン・メニューにあるとき、メニュー・バー・ヘルプ・リストの中から項目固有ヘルプが検索されます。メニュー・バー・ヘルプ・リストは、基本画面ヘルプ・リストの上位にあると見なされます。全般ヘルプが表示されているとき、メニュー・バーおよびプルダウン・メニューのヘルプは、一般ヘルプの直後に表示されます。基本画面ヘルプはメニュー・バー・ヘルプおよびプルダウン・メニュー・ヘルプの後に表示されます。DDS ヘルプが使用され、ユーザーが最初にプルダウン・メニュー・ヘルプを選択する場合には、ユーザーはページ戻ししてメニュー・バー・ヘルプを表示したり、ページ送りして基本画面ヘルプを表示したりすることができます。

「File」メニュー・バー選択項目のヘルプ
「File」プルダウン選択項目 1 のヘルプ
「File」プルダウン選択項目 2 のヘルプ
⋮
「View」メニュー・バー選択項目のヘルプ
「View」プルダウン選択項目 1 のヘルプ
「View」プルダウン選択項目 2 のヘルプ
⋮

RBAHG505-0

図 89. メニュー・バーのヘルプ・リスト

フィールド用ヘルプの定義

フィールドまたは固定情報のヘルプは、HLPARA キーワードを使用して定義できます。名前付きフィールドの場合、*FLD 特殊値およびフィールドの名前を使用してヘルプ領域を指定します。固定情報フィールドの場合、*CNST 特殊値および固定情報フィールドの ID を使用してヘルプ領域を指定します。HLPARA キーワードに指定する ID は、固定情報フィールドのための HLPID キーワードに指定された値と同じでなければなりません。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     HELP
A           R RECORD
A           H                         HLPARA(*FLD FIELD)
A                                     HLPRCD(FIELDHLP LIB/FILE)
A           FIELD           10A  B  5  5
```

図 90. 名前付きフィールド用ヘルプ

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     HELP
A           R RECORD
A           H                         HLPARA(*CNST 1)
A                                     HLPRCD(HLPCNST1 LIB/FILE)
A           2  2'Constant field' HLPID(1)
```

図 91. 固定情報フィールド用ヘルプ

ディスプレイ・ファイルが作成される時、ヘルプ領域の実際の座標は、DDS コンパイラーにより決定されます。これらの座標は、DDS リストの展開ソースのセクションに示されています (フィールドが選択フィールドである場合を除く)。

フィールドまたは固定情報フィールドの位置または長さが変更され、ファイルが再作成される場合、フィールドのヘルプ領域は更新されて、新しい位置または長さが反映されます。

メニュー・バーとプルダウン・メニューのキーの相互作用

図 92は、メニュー・バーおよびプルダウン・メニューの操作で作成される可能性がある画面の例を示しています。大文字の A、B、C、D、および E は、考えられるカーソル位置を示しています。

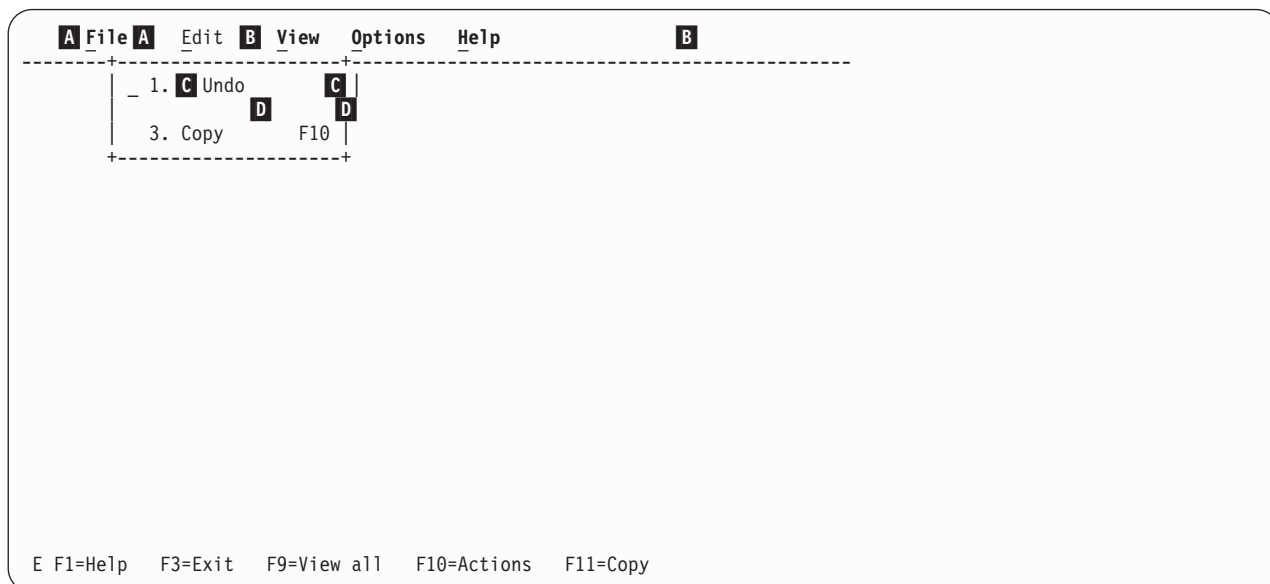


図 92. カーソル位置

表 21は、カーソル位置で特定のキーを押したときに実行されるアクションを説明しています。

表 21. さまざまなカーソル位置で実行されるアクション

カーソル位置	該当するキーを押したときに実行されるアクション			
	ヘルプ	取り消し	メニュー・バー切り替え	Enter
A	「File」のメニュー・バー選択項目用ヘルプが表示されます。(プルダウン・メニューのヘルプを含みます。)	プルダウン・メニューは除去されます。カーソルは「Edit」のメニュー・バー選択項目に移動します。	プルダウン・メニューは除去されます。カーソルは画面作業域に戻ります。	「File」のプルダウン・メニューが表示され、「Edit」のプルダウン・メニューは除去されます。
B	全般ヘルプが表示されます。	プルダウン・メニューは除去されます。カーソルは「Edit」のメニュー・バー選択項目に移動します。	プルダウン・メニューは除去されます。カーソルは画面作業域に戻ります。	「カーソルがメニュー・バー選択項目にありません」のメッセージが表示されます。カーソルは移動しません。
C	「Undo」選択項目のヘルプが表示されます。	プルダウン・メニューは除去されます。カーソルは「Edit」のメニュー・バー選択項目に移動します。	プルダウン・メニューは除去されます。カーソルは画面作業域に戻ります。	ユーザーが選択を行った場合、制御はアプリケーションに戻ります。選択しない場合、カーソルは入力フィールドに移動し、選択を促すメッセージが送られます。

表 21. さまざまなカーソル位置で実行されるアクション (続き)

カーソル位置	該当するキーを押したときに実行されるアクション			
	ヘルプ	取り消し	メニュー・バー切り替え	Enter
D	「Edit」のメニュー・バー選択項目用ヘルプが表示されます。(プルダウン・メニューのヘルプを含みます。)	プルダウン・メニューは除去されます。カーソルは「Edit」のメニュー・バー選択項目に移動します。	プルダウン・メニューは除去されます。カーソルは画面作業域に戻ります。	ユーザーが選択を行った場合、制御はアプリケーションに戻ります。選択しない場合、カーソルは入力フィールドに移動し、選択を促すメッセージが送られます。
E	機能キーのヘルプが表示されます。	プルダウン・メニューは除去されます。カーソルは「Edit」のメニュー・バー選択項目に移動します。	プルダウン・メニューは除去されます。カーソルは画面作業域に戻ります。	ユーザーが選択を行った場合、制御はアプリケーションに戻ります。選択しない場合、ビーブ音が鳴り、カーソルは入力フィールドに移動します。

カーソル移動

タブ・キーまたはカーソル・キーを使用すると、作成したアプリケーション画面上でカーソルを移動することができます。

タブ・キーを押す

タブ・キーを押すと、カーソルを画面上でフィールドからフィールドへ移動させることができます。進行方向は、左から右、上から下となります。

153 ページの表 16の構成 F のディスプレイ装置では、以下のようになります。

- メニュー・バー選択項目は個々のフィールドとなります。
- 単一項目選択フィールドは 1 つのフィールドとなります。
- 複数項目選択フィールドは個々のフィールドとなります。

プルダウン・メニューが表示されていないときにこれらの画面の 1 つでタブ・キーを押すと、カーソルは、メニュー・バー内で選択項目から選択項目へ移動します。その後、カーソルは画面の残りの部分で入力フィールドから入力フィールドへ移動します。これらの画面の 1 つでプルダウン・メニューが表示されている場合、タブ・キーを押すと、以下の処理が行われます。

1. カーソルはメニュー・バー内で選択項目から選択項目へ移動します。
2. 選択された選択項目はスキップオーバーします。
3. カーソルはプルダウン・メニュー内で入力フィールドから入力フィールドへ移動します。
4. カーソルはメニュー・バー内の最初の選択項目に戻ります (その選択項目のプルダウン・メニューは自動的に表示されません)。

プルダウン・メニューが表示されているときには、基本画面の入力フィールドにデータを入力することも入力フィールドにタブ移動することもできません。

151 ページの表 15 の構成 A、B、および C、さらに 153 ページの表 16 の構成 D および E のディスプレイ装置では、メニュー・バーは単一のフィールドとなります。プルダウン・メニューが表示されていない場合、タブ・キーを押すと、カーソルは、メニュー・バー内で選択項目から選択項目へ移動します。カーソルは循環します。メニュー・バー選択項目が選択されると、カーソルはプルダウン・メニュー内の最初の入力フィールドに置かれます。タブ・キーを押すと、カーソルはプルダウン・メニュー内で入力フィールドから入力フィールドへ移動した後、次のメニュー・バー選択項目に移動します。プルダウン・メニューが表示されているときには、基本画面の入力フィールドにデータを入力することも基本画面の入力フィールドにタブ移動することもできません。

カーソル・キーを押す

153 ページの表 16 の構成 F のディスプレイ装置では、カーソル・キーを押すとカーソルは該当する方向に位置 1 つ分移動します。

151 ページの表 15 の構成 A、B、および C、さらに 153 ページの表 16 の構成 D および E のディスプレイ装置では、カーソルは、それが置かれているフィールドのタイプにより異なる仕方で移動します。

- メニュー・バーでは、カーソル右移動またはカーソル左移動キーを押すと、カーソルは選択項目から選択項目へ移動します。カーソルは循環しません。
- 基本画面では、カーソル・キーを押すとカーソルは該当する方向に位置 1 つ分移動します。
- カーソルが選択フィールドに置かれている場合には、以下を参照してください。
 - 164 ページの『垂直項目選択フィールドにおけるカーソル移動』。
 - 165 ページの『水平項目選択フィールドにおけるカーソル移動』。

プログラミングの例

メニュー・バーおよびプルダウン・メニューの表示に必要な DDS の例と DDS コーディングの作動の仕方についての説明が以下にあります。

アプリケーション・レコードでの MNUBARDSP キーワードの使用

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     HELP ALTHELP CF03
A                                     MNUBARSW(CA10) MNUCNL(CA12)
A                                     CF04(04) CF06(06)
A           R MNUBAR                  MNUBAR
A           H                          HLPARA(*FLD MNUFLD 1)
A                                     HLPRCD(FILEHLP HLP LIB/HLPFILE)
A           H                          HLPARA(*FLD MNUFLD 2)
```

```

A          HLPKCD(EDITHLP HLPLIB/HLPFILE)
A          HLPARA(*FLD MNUFLD 3)
A          HLPKCD(VIEWHLP HLPLIB/HLPFILE)
A          HLPARA(*FLD MNUFLD 4)
A          HLPKCD(OPTHLP HLPLIB/HLPFILE)
A          HLPARA(*FLD MNUFLD 5)
A          HLPKCD(HELPHLP HLPLIB/HLPFILE)
A          HLPARA(*FLD MNUFLD)
A          HLPKCD(MNUBARHLP HLPLIB/HLPFILE)
A          MNUFLD          2Y 0B 1 2
A          MNUBARHC(1 PULLFILE +
A          '>File      ')
A 02          MNUBARHC(2 PULLEDIT +
A          &EDITTXT)
A          MNUBARHC(3 PULLVIEW +
A          '>View      ')
A 04          MNUBARHC(4 PULLOPT +
A          '>Options    ' &RTNFLD)
A          MNUBARHC(5 PULLHELP +
A          '>Help      ')
A          MNUBARSEP((*COLOR WHT))
A          CHCAVAIL((*COLOR YLW) (*DSPATR HI))
A          CHCSLT((*COLOR GRN))
A          EDITTXT          20A P
A          RTNFLD          2Y 0H
A          R PULLEDIT
A          PULLDOWN
A          HLPARA(*FLD F1 1)
A          HLPKCD(UNDOHLP HLPLIB/HLPFILE)
A          HLPARA(*FLD F1 2)
A          HLPKCD(MARKHLP HLPLIB/HLPFILE)
A          HLPARA(*FLD F1 3)
A          HLPKCD(COPYHLP HLPLIB/HLPFILE)
A          F1          2Y 0B 1 1SNGCHCFLD
A          CHECK(ER)
A 01          CHOICE(1 '>Undo      ')
A          CHOICE(2 &MARKTXT)
A          CHOICE(3 '>Copy      ')
A          CHCCTL(1 &CTLUNDO MSG1112 QUSER/A)
A          CHCCTL(2 &CTLMARK &MSG &LIB/&MSGF)
A          CHCCTL(3 &CTLCOPY)
A          CHCACCEL(1 'F4')
A          CHCACCEL(2 'F6')
A          CHCAVAIL((*COLOR WHT))
A          CHCUNAVAIL((*COLOR BLU))
A          MARKTXT          20A P
A          CTLUNDO          1Y 0H
A          CTLMARK          1Y 0H
A          CTLCOPY          1Y 0H
A          MSG              7A P
A          LIB              10A P
A          MSGF             10A P
A          R PULLOPT
A          PULLDOWN
A          HLPARA(*FLD F1 1)
A          :
A          :

```

```

A          F1          2Y 0B 1 1SNGCHCFLD
                :
                :
A          R APPSCR          MNUBARDSP(MENUBAR &MNUCHOICE +
A                                     &PULLINPUT)
A          FIELD1          10A B 10 12
A          FIELD2          5S 0B 14 12
A                                     24 1'F1=Help  F3=Exit  +
A                                     F10=Actions  F12=Cancel  '
A          MNUCHOICE          2Y 0H
A          PULLINPUT          2S 0H

```

説明

アプリケーション・レコードで MNUBARDSP キーワードを使用するこの例では、アプリケーションは APPSCR レコードに対して書き込み/読み取り操作を行います。これにより、MENUBAR レコードおよび APPSCR レコードが表示されます。MENUBAR レコードの EDITTXT フィールドには、テキスト >Edit が含まれているため、メニュー・バーの 2 番目の選択項目のテキストとして Edit が表示されます。

注: Edit の E は、画面では下線が付きます。

F10 キーを押すと、メニュー・バー (常時活動状態) をすばやく使用できます。F10 を押すと、カーソルがメニュー・バーの最初の選択項目に移動します。F10 を再度 (または F12 を) 押すと、カーソルはアプリケーション画面上の元の位置に戻ります。153 ページの表 16 の構成 F のディスプレイ装置では、タブ・キーを押すと、カーソルはメニュー・バー内で選択項目から選択項目へ移動します。カーソルは、選択済みメニュー・バー選択項目をスキップオーバーします。その後、カーソルは画面全体で入力フィールドから入力フィールドへ移動します。151 ページの表 15 の構成 A、B、および C、さらに 153 ページの表 16 の構成 D および E のディスプレイ装置では、カーソルがメニュー・バー内に置かれている場合には、カーソル移動キーまたはタブ・キーで、カーソルはメニュー・バー内で選択項目から選択項目へ移動します。

ユーザーが「Edit」のアクションを選択する場合、システムはプルダウン・メニュー・レコード (PULLEDIT) を表示します。CHECK(ER) は自動入力を指定します。ユーザーが F1 に値を入力すると、ユーザーが Enter キーを押さなくても制御はアプリケーションに戻ります。ユーザーが F4 または F6 を押す場合にも、制御はアプリケーションに戻ります。

アプリケーションが APPSCR レコードに対して書き込み/読み取り操作を実行したため、APPSCR レコードはアプリケーションに戻ります。フィールド MNUCHOICE には、メニュー・バー選択項目 2 が選択されたことを示す 2 が入ります。PULLINPUT フィールドには、PULLEDIT レコードの単一選択フィールド選択項目 (F1 の内容) が入ります。

ユーザーが「Options」のアクションを選択する場合、制御は、フィールド RTNFLD に 4 の設定、フィールド MNUCHOICE に 0 の設定でアプリケーションに戻ります。アプリケーションは、フィールド MNUCHOICE の 0 から、制御がプルダウン・メニューの更新のために戻されたということを判別します。アプリケーションはレコード MENUBAR を読み取り、フィールド RTNFLD に設定された選択項目の番号を取得します。アプリケーションはレコード PULLOPT を更新した後、レコード PULLOPT を書き込みます。その後システムは、PULLOPT を「Options」選択項目のプルダウン・メニューとして表示します。システムは、アプリケーションが APPSCR レコードへの読み取り操作を実行する時のメニュー・バー相互作用の制御を再開します。

151 ページの表 15 の構成 A および B のディスプレイ装置では、メニュー・バー区切り文字およびプルダウン・メニューは実線で表示されます。

151 ページの表 15の構成 C、および 153 ページの表 16の構成 D、E、および F のディスプレイ装置では、メニュー・バー区切り文字はダッシュで構成されます。プルダウン・メニューの両側および下部の枠は、それぞれコロンおよびピリオドで構成されます。

151 ページの表 15の構成 C のディスプレイ装置では、メニュー・バー選択項目にはそれぞれ略号があります。151 ページの表 15の構成 A および B のディスプレイ装置では、PULLEDIT プルダウン・メニューの単一選択フィールド選択項目には、それぞれ略号があります。これは、CRTDSPF コマンドまたは CHGDSPF コマンドに ENHDSP(*NO) が指定されているのでないかぎり、当てはまります。この例では、略号は各選択項目の先頭文字になるようにセットアップされています。

メニュー・バー・レコードでの MNUBARDSP キーワードの使用

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     HELP ALTHELP CF03(03)
A                                     MNUBARSW(CA10) MNUCNL(CA12 12)
A                                     CF04(04) CF06(06)
A                                     R MNUBAR
A 01                                MNUBAR
A                                     MNUBARDSP
A                                     H
A                                     HLPARA(*FLD MNUFLD 1)
A                                     HLPARCD(FILEHLP HLPLIB/HLPFILE)
A                                     H
A                                     HLPARA(*FLD MNUFLD 2)
A                                     HLPARCD(EDITHLP HLPLIB/HLPFILE)
A                                     H
A                                     HLPARA(*FLD MNUFLD 3)
A                                     HLPARCD(VIEWHLP HLPLIB/HLPFILE)
A                                     H
A                                     HLPARA(*FLD MNUFLD 4)
A                                     HLPARCD(OPTHLP HLPLIB/HLPFILE)
A                                     H
A                                     HLPARA(*FLD MNUFLD 5)
A                                     HLPARCD(HELPHLP HLPLIB/HLPFILE)
A                                     MNUFLD          2Y 0B 1 2
A                                     MNUBARCHC(1 PULLFILE +
A                                     '>File      ')
A 02                                MNUBARCHC(2 PULLEDIT +
A                                     &EDITTXT)
A                                     MNUBARCHC(3 PULLVIEW +
A                                     '>View      ')
A 04                                MNUBARCHC(4 PULLOPT +
A                                     '>Options   ' &RTNFLD)
A                                     MNUBARCHC(5 PULLHELP +
A                                     '>Help      ')
A                                     MNUBARSEP((*COLOR WHT) +
A                                     (*CHAR '-'))
A                                     EDITTXT          20A P
A                                     RTNFLD          2Y 0H
A                                     R PULLEDIT
A                                     PULLDOWN
A                                     H
A                                     HLPARA(*FLD F1 1)
A                                     HLPARCD(UNDOHLP HLPLIB/HLPFILE)
A                                     H
A                                     HLPARA(*FLD F1 2)
A                                     HLPARCD(MARKHLP HLPLIB/HLPFILE)
A                                     H
A                                     HLPARA(*FLD F1 3)
A                                     HLPARCD(COPYHLP HLPLIB/HLPFILE)
A 01                                F1          2Y 0B 1 1SNGCHCFLD
A                                     CHOICE(1 '>Undo      ')
A                                     CHOICE(2 &MARKTXT)
A                                     CHOICE(3 '>Copy      ')
A                                     CHCCTL(1 &CTLUNDO MSG1112 QUSER/A)
A                                     CHCCTL(2 &CTLMARK &MSG &LIB/&MSGF)
A                                     CHCCTL(3 &CTLCOPY)
A                                     CHCACCEL(1 'F4')
A                                     CHCACCEL(2 'F6')
```

```

A          MARKTXT          20A P
A          CTLUNDO          1Y 0H
A          CTLMARK          1Y 0H
A          CTLCOPY          1Y 0H
A          MSG              7A P
A          LIB              10A P
A          MSGF             10A P
A
A          R PULLOPT
A
A          H                  PULLDOWN
A          H                  HLPARA(*FLD F1 1)
A          H                  :
A          H                  :
A          F1                2Y 0B 1 1SNGCHCFLD
A          H                  :
A          H                  :
A          R APPSCR
A          FIELD1            10A B 10 12
A          FIELD2            5S 0B 14 12
A          H                  24 1'F1=Help F3=Exit +
A          H                  F10=Actions F12=Cancel '

```

説明

メニュー・バー・レコードで MNUBARDSP キーワードを使用するこの例では、アプリケーションは、メニュー・バーを表示するために MNUBAR レコードに書き込みを行います。その後アプリケーションは、APPSCR レコードに対して書き込み/読み取り操作を実行し、アプリケーション画面を表示させます。

前述の例のように、F10 で、メニュー・バー (常時活動状態) をすばやく使用できます。F10 を押すと、カーソルがメニュー・バーの最初の選択項目に移動します。F10 を再度 (または F12 を) 押すと、カーソルはアプリケーション画面上の元の位置に戻ります。

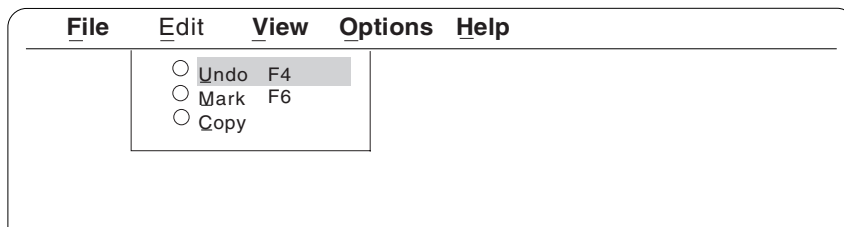
153 ページの表 16の構成 F のディスプレイ装置では、タブ・キーを押すと、カーソルは、メニュー・バー内では選択項目から選択項目へ、画面全体では入力フィールドから入力フィールドへ移動します。カーソルは、選択済みメニュー・バー選択項目をスキップオーバーします。

151 ページの表 15の構成 A、B、および C、さらに 153 ページの表 16の構成 D および E のディスプレイ装置で、カーソルがメニュー・バー内に置かれている場合には、カーソル移動キーまたはタブ・キーによって、カーソルはメニュー・バー内で選択項目から選択項目へ移動します。

ユーザーが「Edit」のアクションを選択する場合、システムはプルダウン・メニュー・レコード (PULLEDIT) を表示します。フィールド F1 に CHECK(ER) が指定されていないため、ユーザーは、選択項目番号を入力し Enter キーを押すか、または F4 または F6 を押すことにより、値を入力できます。値を入力するか、またはアクセラレーター・キーを押すと、制御はアプリケーションに戻ります。アプリケーションが APPSCR レコードに対して書き込み/読み取り操作を実行していたため、APPSCR レコードはアプリケーションに戻ります。その後アプリケーションは MNUBAR レコードを読み取り、選択された選択項目を判別します (フィールド MNUFLD に 2 が戻されます)。入力フィールド・パラメーターが MNUBARDSP キーワードに指定されなかったため、アプリケーションはレコード PULLEDIT を読み取り、プルダウン・メニューの入力データを受け取ります。

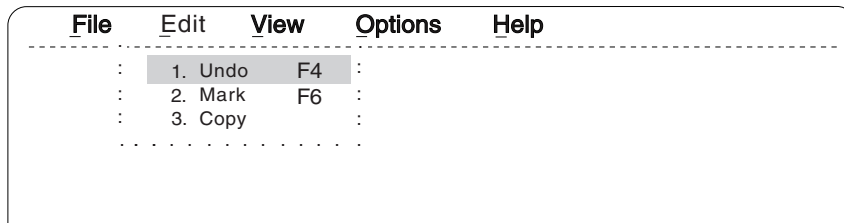
画面の外観

151 ページの表 15の構成 A および B のディスプレイ装置では、画面は次のような外観になります。



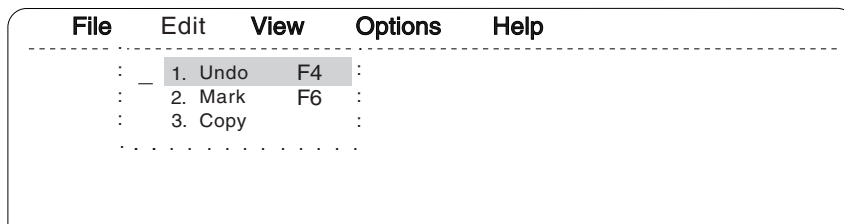
RV2W855-1

151 ページの表 15の構成 C のディスプレイ装置では、画面は次のような外観になります。



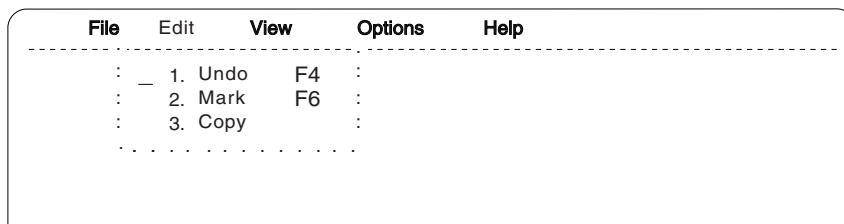
RV2W067-0

153 ページの表 16の構成 D および E のディスプレイ装置では、画面は次のような外観になります。



RV2W068-1

153 ページの表 16の構成 F のディスプレイ装置では、画面は次のような外観になります。



RV3W075-0

単純ホット・スポット

非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置は、単純ホット・スポットを備えています。ホット・スポットとは、クリックするとある機能が実行される、画面の領域のことです。ディスプレイ装置とともにマウスがなければなりません。ホット・スポットは、151 ページの表 15の構成 A および B で使用可能です。InfoWindow II ディスプレイ装置では、ホット・スポットは左マウス・ボタンで選択する必要があります。

非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置により、以下のホット・スポットが提供されています。

コマンド・キー・エミュレーション

前ページおよび次ページ・キー・エミュレーション

Enter キー・エミュレーション

ユーザーは、左マウス・ボタンをダブルクリックすることにより、InfoWindow II ディスプレイ装置で Enter 機能を実行できます。

コマンド・キー・エミュレーション

ユーザーがコマンド・キーをクリックする時、ポインター装置カーソルは、コマンド・キーの先頭文字と最終文字の間 (両端を含む) になければなりません。ユーザーがコマンド・キーを選択すると、キーボードはロックされ、コマンド・キーが処理されます。実行される機能は DDS により定義されます。システムは、実際のコマンド・キーが押された場合のように機能を実行します。これには、応答標識を設定することも含まれます。

ユーザーがコマンド・キーをクリックすると、システムは、以下のいずれかが検出されるまで左に向かってコマンド・キー・テキストをスキャンします。

属性

2 つのブランク

2 つのヌル

1 つのブランクと 1 つのヌル

行の桁 1

コマンド・キー・テキスト・ストリングの先頭が検出されると、その後システムは、以下のいずれかと一致するものが検出されるまで右に向かってテキストをスキャンします。

Fx= x は 1 から 9 のいずれかです。

Fyx= y は 0、1、または 2 のいずれかです。y が 0 である場合、x は 1 から 9 のいずれかです。y が 1 である場合、x は 0 から 9 のいずれかです。y が 2 である場合、x は 0 から 4 のいずれかです。

PFx= x は 1 から 9 のいずれかです。

PFyx= y は 0、1、または 2 のいずれかです。y が 0 である場合、x は 1 から 9 のいずれかです。y が 1 である場合、x は 0 から 9 のいずれかです。y が 2 である場合、x は 0 から 4 のいずれかです。

PFx x は 1 から 9 のいずれかです。

PFyx y は 0、1、または 2 のいずれかです。y が 0 である場合、x は 1 から 9 のいずれかです。y が 1 である場合、x は 0 から 9 のいずれかです。y が 2 である場合、x は 0 から 4 のいずれかです。

システムは一致したものを検出すると、コマンド・キー機能を実行します。

前ページおよび次ページ・キー・エミュレーション

ユーザーは、プラス (+) およびマイナス (-) 文字をクリックして、それぞれページ送り (次ページ) およびページ戻し (前ページ) を行うことができます。ユーザーが + または - 文字をクリックする時、キーボー

ドはロックされ、次ページ AID キーおよび前ページ AID キーが生成されます。実行される機能は DDS により定義されます。システムは、実際のキーが押された場合のように機能を実行します。

この機能をインプリメントするための 1 つの方法は、以下を行うことです。

1. サブファイル終了 (SFLEND) キーワードに *MORE を指定する。

これにより、CPX6AB2 および CPX6AB1 メッセージに定義されたテキストをディスプレイ・ファイルが使用するようになります。これらのメッセージのデフォルトのテキストは、それぞれ「続く…」および「終わり」です。

2. CPX6AB2 メッセージのテキストを「続く: +/-」に変更する。
3. CPX6AB1 メッセージのテキストを「終わり: -」に変更する。

プログラマブル・マウス・ボタン - 概要

プログラマブル・マウス・ボタン機能は、アテンション標識 (AID) をさまざまなポインター装置イベントに関連付けられるようにします。AID コードは通常、キーボード上のさまざまなコマンド・キーに関連付けられています。これらのキーは、ユーザーからシステムまたはアプリケーションにアクション要求を伝えるために使用されます。AID を生成するコマンド・キーの中には、Enter、ヘルプ、次ページ、前ページ、および 24 個のコマンド・アテンションまたは機能キーがあります。単一イベント AID および 2 イベント AID をプログラムできます。ポインター装置イベントと、それに関連付けられた AID を 18 対まで定義できます。これらのイベントは、2 つのキーボード状態 (シフトおよび非シフト) で、それぞれ 3 つのイベント (上げ、下げ、およびダブルクリック) を持つ 3 つのボタンにより構成されます。

単一イベント AID の定義も 1 つの AID コードを単一のポインター装置イベントと関連付けますが、2 イベント AID の定義も AID を 2 つの連続したポインター装置イベントと関連付けます。

コマンド・キーまたはイベント ID を 1 つか 2 つのポインター装置イベントに関連付けるには、プログラマブル・マウス・ボタン (MOUBTN) キーワードを使用します。このキーワードは、ファイルまたはレコード・レベルで指定できます。

注:

1. この機能は、非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されているディスプレイのみに使用可能です。
2. サポートされるポインター装置は、マウスまたはマウスをエミュレートする装置のみです。

ポインター装置イベント

マウス (3 ボタン) では、2 つのキーボード状態 (シフトおよび非シフト) で、それぞれ 3 つのイベント (上げ、下げ、ダブルクリック) を持つ 3 つのボタンがあり、18 のポインター装置イベントが考えられます。ポインター装置イベントは以下のとおりです。

- 左ボタン押し
- 左ボタン・リリース
- 左ボタン・ダブルクリック
- 右ボタン押し
- 右ボタン・リリース
- 右ボタン・ダブルクリック
- 中央ボタン押し
- 中央ボタン・リリース

- 中央ボタン・ダブルクリック
- シフト左ボタン押し
- シフト左ボタン・リリース
- シフト左ボタン・ダブルクリック
- シフト右ボタン押し
- シフト右ボタン・リリース
- シフト右ボタン・ダブルクリック
- シフト中央ボタン押し
- シフト中央ボタン・リリース
- シフト中央ボタン・ダブルクリック

注:

1. シフト・ポインター装置イベントでは、シフト・キーまたはシフト・ロック・キーを押しておく必要があります。キャップス・ロック状態およびシフト・ロック状態はシフトとは見なされません。シフト・ポインター装置イベントとして使用する場合には、シフト・キーを放しても、キャップス・ロック状態またはシフト・ロック状態はリセットされません。
2. 非プログラマブル・ワークステーション (NWS) には、左および右ボタンの機能を NWS 内で切り替えるためのセットアップ・オプションがあります。システムはそのことを認識していません。これは、左手用および右手用マウスの概念を提供するものとなります。この資料では、マウス・ボタンの説明をする場合にはすべて、左および右ボタンは通常の左および右の定義に従う右手用マウスを想定しています。

戻される AID コード

ポインター装置イベントに関連付けられた AID は、現在サポートされている任意の AID、または X'70' と X'7F' の間のホスト定義の AID 値とすることができます。以下の AID がサポートされています。

X'31' から X'3C'

CA/CF01 から CA/CF12 (Cmd 1 から 12)

X'70' から X'7F'

E00 から E15 (EVENTS)

X'B1' から X'BC'

CA/CF13 から CA/CF24 (Cmd 13 から 24)

X'BD' CLEAR (消去)

X'F1' Enter またはレコード前進

X'F3' HELP (エラー状態ではないヘルプ)

X'F4' ROLLDOWN (次ページ)

X'F5' ROLLUP (前ページ)

X'F6' PRINT (印刷)

X'F8' HOME (レコード・バックスペース)

プログラマブル・マウス・ボタン - 利点

マウス・ボタンは、ウィンドウのレイヤーの再配列やオブジェクトの選択などのために使用できます。たとえば、アプリケーション・プログラマーは、中央ボタン下げを単一イベント AID としてプログラムするこ

とにより、複数のウィンドウが画面にある場合にウィンドウの再配列を可能にすることができます。ユーザーが中央のボタンを押すと、カーソルはポインター装置カーソル位置に移動し、ホスト定義の AID が戻されます。アプリケーションは、ホスト定義の AID を使用して再配列要求を認識します。行と桁のカーソル・アドレスが、オーバーレイされたウィンドウ内でポインター装置イベントが発生したことを示している場合、ウィンドウを再配列できます。行と桁のカーソル・アドレスが、オーバーレイされたウィンドウ内でポインター装置イベントが発生しなかったことを示している場合、アプリケーションは、AID を無視したり、メッセージを通知したりすることができます。

注: EVENT ID (E00 から E15) が使用される場合、キーボードを使用して実行される同等の機能を定義しようと思うかもしれません。たとえば、ユーザーがテキスト・カーソルを配置して機能キーを押すと、マウス・ボタンを押したのと同様の機能を実行できるようにするかもしれません。これにより、非プログラマブル・ワークステーション用の拡張インターフェースをサポートしない制御装置に接続されたディスプレイおよびポインター装置を持たない非プログラマブル・ディスプレイでの機能が使用可能になります。

プログラマブル・マウス・ボタンの操作

単一イベントとしてプログラムされたポインター装置イベントが実行され、他の機能でより高い優先順位を持つものがない場合、以下が実行されます。

1. キーボードがロックされず (機能キーの場合と同様)
2. カーソルがポインター装置カーソル位置に移動します。
3. 指定された AID がホストに戻されます。
4. AID または通常のカーソル移動の結果として入力フィールド・データの妥当性チェックが実行される場合、データの妥当性チェックが実行されます。
5. 指定された AID が通常通りにインバウンド入力フィールド・データを戻す場合、インバウンド入力フィールド・データが組み込まれます。インバウンド・データの形式は、標準的なインバウンド・データに類似しています。
6. 制御はアプリケーションに戻されます。

アプリケーションの側では、ポインター装置イベントとそれに対応するコマンド・キーの間を区別する方法はありません。ただし、ポインター装置イベントは、どのコマンド・キーとも関連付けられていない EVENT ID (E00 から E15) と関連付けることができます。これにより、ポインター装置イベントを検出する方法が確保できます。

2 イベント・ポインター装置イベントが実行される時、システムはリーディング・エッジ・イベントを探します。リーディング・エッジ・イベントを受け取ると、以下が実行されます。

注: インバウンド・データは、トレーリング・エッジ・イベントが発生するまで戻されません。

1. プログラマブル 2 イベント状態に入ります。
2. マーカー・ボックスの表示が可能な非プログラマブル・ワークステーション上のポインター装置カーソルの位置の周りにマーカー・ボックスが描画されます。マーカー・ボックスは、文字の周りの 4 本の青い線として表示されます。
3. ポインター装置のカラーは、白の表示が可能な非プログラマブル・ワークステーションでは、白に変更されます。
4. システムは、トレーリング・エッジ・イベントを探します。

キー・ストロークおよびホスト・データ・ストリームは、プログラマブル 2 イベント状態を取り消します。一部のポインター装置イベントは、トレーリング・エッジ・イベントの待機中に無視されます。詳しく

は、『プログラマブル・マウス・ボタン - NWS の考慮事項』を参照してください。トレーリング・エッジ・イベントを受け取ると、以下が実行されます。

1. マーカー・ボックスが消去されます
2. ポインター装置カーソルのカラーが入力禁止に変更されます
3. キーボードがロックされます
4. テキスト・カーソルがポインター装置カーソルの位置に移動します

指定されたホスト定義 AID が通常通りにインバウンド入力フィールド・データを戻す場合、インバウンド・データが組み込まれます。行と桁の終了位置が戻されます。

注: RTNCSRLOC キーワードを使用してカーソルの開始位置を取得することもできます。これは、2 イベント定義を使用する場合にはカーソルの終了位置とは異なる場合があります。

プログラマブル・マウス・ボタンを使用すると、画面上の他のポインター装置機能を禁止する場合があります。たとえば、シフト左ボタン押しおよびリリース用のプログラマブル・マウス・ボタンより、コピー・アンド・ペースト機能の方が低い優先順位です。ポインター装置イベントの優先順位についての詳細は、220 ページの『プログラマブル・マウス・ボタン - イベント処理の優先順位』を参照してください。

プログラマブル・マウス・ボタン - NWS の考慮事項

ポインター装置イベントの多くでは、テキスト・カーソルが移動するか、AID がホストに送られるかのいずれかの結果となります。テキスト・カーソルが入力フィールドにあった場合、テキスト・カーソルの移動が可能になるか AID が送られる前に、入力フィールド要件 (全桁入力必須など) が検査されます。その結果として、エラー・コードが通知され、ポインター装置イベントが処理されない可能性があります。たとえば、全桁入力フィールドにヌルが含まれていることを示す 0014 エラー・コードが通知される場合があります。

NWS は、ボタンを押したり放したりする時には常にイベントをシステムに渡します。ボタンが押され、放され、ユーザー指定のダブルクリック時間内に再度押された場合に、NWS はダブルクリック・イベントをシステムに渡します。システムは、ボタン押しイベント、ボタン・リリース・イベント、およびダブルクリック・イベントを感知し、最後にはボタン・リリース・イベントを感知します。

次のいずれかに該当する場合は、ポインター装置イベントはシステムにより無視されます。

- キーボードがロックされている。例外は、キーボードがロックされている場合にはキューに入れるように定義できる単一イベント・プログラマブル・マウス・ボタンです。
- キーボードがシステム要求状態または ss メッセージ状態である。
- キーボードがオペレーター・エラー状態である。例外は、オペレーター・エラーをリセットできる左ボタン下げおよびシフト左ボタン下げです。
- 画面が WP モードである。
- システムには、保管済み先行入力キー・ストロークがある。
- システムには保留読み取りがない。例外は、単一イベント・プログラマブル・マウス・ボタンです。キーボードがアンロックされると、システムは通常、保留読み取りがあります。ただし、3270DE は保留読み取りがない状態でキーボードをアンロックする時があります。ポインター装置イベントを処理すると、この場合、混乱の原因となります。

プログラマブル・マウス・ボタン - イベント処理の状態

ポインター装置イベントの処理は、さまざまな状態で行われます。予期しない状態で以下のイベントを受け取ると、ポインター装置イベント処理の状態がリセットされます。

- ほとんどのマウス・ボタン・イベント (別に指定されているものを除く)
- ほとんどのキーボード・イベント (シフト・キーのメーク/ブレイクなどを除く)
- 全ホスト画面更新

状態は以下のようにリセットされます。

- スクロール・バーのドラッグ・アンド・ドロップ状態が活動状態である場合、ドラッグ・アンド・ドロップ状態はリセットされ、スクロール・バーは元の状態に書き直されます。
- コピー・アンド・ペースト状態が活動状態である場合、コピー・アンド・ペースト状態はリセットされ、終了点標識は画面から除去されます。
- プログラマブル・マウス・ボタン 2 イベント状態が活動状態である場合、状態はリセットされ、AID は通知されません。状態をリセットすることが望ましくない場合には状態をリセットしないようにするため、2 イベント状態では以下のイベントが無視されます。
 - リーディング・エッジ・イベントに関連付けられたマウス・ボタン・リリース・イベント。
 - トレーリング・エッジ・イベントに到達するために生成される必要があるマウス・ボタン押しイベント。トレーリング・エッジは、同じボタンで同じシフト状態でのボタン・リリースまたはダブルクリックでなければなりません。
 - トレーリング・エッジ・イベントに到達するために生成される必要があるマウス・ボタン・リリース・イベント。トレーリング・エッジは、同じボタンで同じシフト状態でのダブルクリックでなければなりません。

状態をリセットするマウス・ボタン・イベントはそれ以上処理されません。キーボード・イベントおよびホスト画面更新は、マウス・イベント処理の状態をリセットした後に通常どおり処理されます。

プログラマブル・マウス・ボタン - イベント処理の優先順位

このセクションは、以下のイベントのイベント処理の優先順位を説明しています。

- 非シフト左ボタン押し
- 非シフト左ボタン・リリース
- 非シフト左ボタン・ダブルクリック
- シフト左ボタン押し
- シフト左ボタン・リリース
- シフト右ボタン押し
- 他のすべてのイベント

非シフト左ボタン押しイベントの処理

シフト・キーが押されておらず、システムが左ボタン押しイベントを受け取る場合、システムはポインター装置カーソルの位置を判別し、以下の機能を、リストされている順序で確認しながら 1 つだけ実行します。

1. オペレーター・エラーが画面にある場合、オペレーター・エラーをリセットするためにポインター装置カーソルを使用することができます。ポインター装置カーソルの位置により、以下のいずれかの機能が実行されます。
 - a. ポインター装置カーソルがオペレーター・エラー行にある場合には、ユーザーがリセット・キーを押した場合と同じ機能が実行されます。リセット機能は、ポインター装置カーソルが画面の最終行にあり、別個に表示可能なオペレーター・エラー行 (行 25 または行 28) が使用されている場合にも実行されます。

- b. そうでない場合は、ポインター装置イベントは無視され、それ以降の検査は実行されません。
2. ポインター装置カーソルが選択フィールド選択項目にある場合には、以下の処理が実行されます。
- a. 選択項目がカーソル配置可能選択項目である場合、カーソルはポインター装置カーソルの位置に移動します。選択カーソルが作成されます。スペース・バー・キーの機能が実行されます (これには、選択項目が選択不可である場合にオペレーター・エラー 0084 を通知することが含まれます)。カーソルは、選択項目内の最初のテキスト属性と最後のテキスト属性の間になければなりません。選択標識が使用される場合、カーソルは、選択項目内の選択標識属性と最後のテキスト属性の間になければなりません。
 - b. カーソル配置不可能またはヌルの選択項目にある場合、カーソルは移動せず、オペレーター・エラー 0084 が通知されます。
3. ポインター装置カーソルがスクロール・バー文字 (矢印文字またはシャフト文字) の上にある場合、ページ送り AID がホストに送られる可能性があります。ページ送り AID がホストに送られる場合、フィールド MDT はオンに設定されます。テキスト・カーソルはポインター装置カーソルの位置に移動する可能性があります。これは、以下のいずれかに依存します。
- スクロール・バー定義コマンドの「ポインター装置の相互作用でカーソルをスクロール・バー上に移動する」フラグの設定
 - このスクロール・バーが選択フィールドと関連付けられている場合には、カーソルの位置
- キーボードがロックされます (ユーザーが通常の AID キーを押した場合のように)。
- a. ポインター装置カーソルがスライダーの上のシャフトにある場合、スクロール移動量が X'00000000' である前ページ AID がホストに送られます。同様に、ポインター装置カーソルがスライダーの下のシャフトにある場合、次ページ AID が送られます。
 - b. ポインター装置カーソルが上部矢印文字上にあり、スライダーがまだスクロール・バーの上部にない場合、スクロール移動量が X'00000001' である前ページ AID がホストに送られます。同様に、ポインター装置カーソルが下部矢印文字上にあり、スライダーがまだスクロール・バーの下部にない場合、スクロール移動量が X'00000001' である次ページ AID がホストに送られます。ポインター装置カーソルが上部矢印文字上にあり、スライダーがすでにスクロール・バーの上部にある場合、左ボタン押しは無視されます。下部矢印文字も同様に機能します。
 - c. ポインター装置カーソルがスライダーにある場合には、ドラッグ・アンド・ドロップ機能が開始されなければなりません (スクロール・バー・スライダー・ドラッグ・アンド・ドロップ状態)。NWS は、ポインター装置カーソル移動をシステムに渡すように通知されます。スクロール・バー・ドラッグ・アンド・ドロップ状態では、移動イベントごとにシステムはポインター装置カーソルの行を計算し、この値を処理された最後の行と比較します。行の値が異なる場合、スクロール・バー文字は書き直されます。スライダーはシャフト内で再配置されます。ポインター装置カーソルが 1 行上に移動した場合、スライダーは 1 行上に移動します。ポインター装置カーソルが、スライダーの上のシャフトに存在する行より多く上に移動する場合には、スライダーはスクロール・バー・シャフトの上部に移動します。左ボタンをリリースすると、ドロップ機能が実行されます。ページ送り AID 要求の説明については、『左ボタン・リリース』を参照してください。左ボタン・リリース以外のポインター・イベントが発生した場合、またはキーボード・キーが押された場合、またはホストにより画面更新が行われた場合、以下が実行されます。
- スクロール・バー文字は書き直され、元の状態になります。
 - スクロール・バー・スライダー・ドラッグ・アンド・ドロップ状態がリセットされます。
 - ページ送り AID は送られません。
 - MDT は設定されません。

4. ポインター装置カーソルがライト・ペン・フィールド (最初のフィールド位置から最後のフィールド位置まで) にある場合、システムは、ライト・ペン・ヒント・スイッチがポインター装置カーソルの位置で活動化された場合のようにイベントを処理します。
5. このマウス・ボタン・イベントがプログラマブル・マウス・ボタン構造化フィールドでプログラムされている場合、そのイベントは前述のように処理されます。
6. ポインター装置カーソルが単純ホット・スポットにある場合には、ホット・スポット機能が実行されず。ホット・スポットは、ポインター装置が古いアプリケーションの一部を駆動することを可能にします。ホット・スポットであると見なされるためには、ポインター装置カーソルが入力フィールドに置かれてはなりません。非シフト左ボタン押しイベントのホット・スポット機能は以下のとおりです。
 - コマンド・キー・エミュレーション。詳しくは、215 ページの『コマンド・キー・エミュレーション』を参照してください。
 - 前ページおよび次ページ・キー・エミュレーション。詳しくは、215 ページの『前ページおよび次ページ・キー・エミュレーション』を参照してください。
 - Enter キー・エミュレーション。詳しくは、223 ページの『非シフト左ボタン・ダブルクリック・イベントの処理』を参照してください。
7. プルダウン取り消しモードが活動状態である場合、カーソルはポインター装置カーソルの位置に移動し、キーボードはロックされ (通常の AID のように処理され)、さらに指定された AID がホストに戻されます。選択フィールドが画面に書き込まれ、選択フィールド定義大構造の中でプルダウン取り消し AID が指定された場合、プルダウン取り消しモードは活動状態です。

注: プルダウン・メニューの中にはコマンド・キーまたは「続く -/+」が含まれている可能性があるため、プルダウン取り消しモードはホット・スポットより低い優先順位です。
8. 上記が当てはまらない場合、カーソルはポインター装置カーソルの位置に移動します。カーソルはテキスト・カーソルまたは選択カーソル (強調表示された入力フィールドの場合) である可能性があります。入力可能な位置のみへのカーソル移動がオンに設定されている場合であっても、システムは、カーソル配置不能テキスト位置へのテキスト・カーソルの移動を許可します。

非シフト左ボタン・リリース・イベントの処理

シフト・キーが押されておらず、システムが左ボタン・リリース・イベントを受け取る場合、システムはポインター装置カーソルの位置を判別し、以下の機能を、リストされている順序で確認しながら 1 つだけ実行します。

1. スクロール・バー・スライダ・ドラッグ・アンド・ドロップ状態が活動状態である場合、スクロール・バー文字は更新される可能性があります。スクロール・バーの詳細については、220 ページの『非シフト左ボタン押しイベントの処理』を参照してください。ポインター装置カーソルの行位置が、スライダが最後に書き込まれた時の行位置とは異なる場合、スクロール・バー文字が書き込まれる必要があります。最終スライダ位置が元のスライダ位置 (ドラッグ・アンド・ドロップが開始された時点) とは異なる場合、以下が実行されます。
 - a. MDT がオンに設定されます。
 - b. テキスト・カーソルはスクロール・バー・スライダに移動する可能性があります。
 - c. ページ送り AID が、スクロールされる行数または桁数を示すスクロール移動量と共にホストに送られます。
 - d. キーボードがロックされます (通常の AID と同様に処理されます)

特定の AID は、スライダが移動する方向に依存します。スライダが移動しなかった場合には、AID は送られません。すべての場合において、スクロール・バー・スライダ・ドラッグ・アンド・ドロップ状態がリセットされます。

2. このマウス・ボタン・イベントがプログラマブル・マウス・ボタン構造化フィールドでプログラムされている場合、そのイベントは前述のように処理されます。
3. そうでない場合は、ポインター装置イベントは無視されます。

非シフト左ボタン・ダブルクリック・イベントの処理

シフト・キーが押されておらず、システムが左ボタン・ダブルクリック・イベントを受け取る場合には、システムはポインター装置カーソルの位置を判別し、以下の機能を、リストされている順序で確認しながら 1 つだけ実行します。

1. このマウス・ボタン・イベントがプログラマブル・マウス・ボタン構造化フィールドでプログラムされている場合、そのイベントは前述のように処理されます。
2. そうでない場合は、直前の左ボタン押しイベントでカーソルを配置しただけであった場合には、キーボードはロックされ、Enter AID がホストに送られます。これはホット・スポット機能です。
3. そうでない場合は、非シフト左ボタン・ダブルクリック・イベントは無視されます。

注: ユーザーは、以下のいずれかを行ったはずです。

- 選択フィールド選択項目を選択した
- スクロール・バーに対して操作を行った
- ポインター装置で選択可能な AID を発生させた
- ホット・スポットを選択した (コマンド・キーなど)
- 単にカーソルを配置するだけのデフォルトのアクション以外の、他の左ボタン押しイベントの機能を発生させた。

シフト左ボタン押しイベントの処理

シフト・キーが押されており、システムが左ボタン押しイベントを受け取る場合、システムはポインター装置カーソルの位置を判別し、以下の機能を、リストされている順序で確認しながら 1 つだけ実行します。

1. オペレーター・エラーが画面にある場合、オペレーター・エラーをリセットするためにポインター装置カーソルを使用することができます。ポインター装置カーソルの位置により、以下のいずれかの機能が実行されます。
 - a. ポインター装置カーソルがオペレーター・エラー行にある場合には、ユーザーがリセット・キーを押した場合と同じ機能が実行されます。リセット機能は、ポインター装置カーソルが画面の最終行にあり、別個に表示可能なオペレーター・エラー行 (行 25 または行 28) が使用されている場合にも実行されます。
 - b. そうでない場合は、ポインター装置イベントは無視され、それ以降の検査は実行されません。
2. このマウス・ボタン・イベントがプログラマブル・マウス・ボタン構造化フィールドでプログラムされている場合、そのイベントは前述のように処理されます。
3. そうでない場合は、コピー・アンド・ペースト機能 (しばしばカット・アンド・ペーストと呼ばれる) が開始されます。コピー・アンド・ペースト状態が設定されます。ポインター装置カーソルの位置は、選択の末端地点を示すためにマークされます。ポインター装置カーソルの文字位置の上と下に 1 本の線が描画されます。シフト左ボタン・リリース以外の他のポインター装置イベントが発生した場合、またはユーザーがいずれかのキーを押した場合、またはホストからなんらかのデータを受け取った場合、コピー・アンド・ペースト状態はリセットされ、2 本の線は除去されます。シフト状態の左ボタンをリリースすると、コピー・アンド・ペースト処理は継続し (224 ページの『シフト左ボタン・リリース・イベントの処理』を参照)、以降の貼り付け操作で使用されます (225 ページの『シフト右ボタン押しイベントの処理』を参照)。

シフト左ボタン・リリース・イベントの処理

シフト・キーが押されており、システムが左ボタン・リリース・イベントを受け取る場合、システムはポインター装置カーソルの位置を判別し、以下の機能を、リストされている順序で確認しながら 1 つだけ実行します。

1. コピー・アンド・ペースト状態が活動状態である場合、ユーザーはコピーの 2 番目の末端地点をマークしていることとなります。2 番目の末端地点は、2 番目の末端地点の周りに 3 本の線を描画し、最初の末端地点の周りに 1 本の線を追加することによりマークされます。これは、コピー・データを囲む大括弧のように見えます。ユーザーは、開始点または終了点のどちらを先にマークしてもかまいません。その時、マークされたコピー・データはディスプレイ (たとえば、348X NWS) に保管され、後で使用されます (225 ページの『シフト右ボタン押しイベントの処理』を参照)。パフォーマンス上の理由で、コピー・アンド・ペースト・データは、ディスプレイからディスプレイのシステム管理バッファにコピーされます (平衡型ケーブルを経由せずに)。データがディスプレイ・メモリー内部でコピーされた後、コピー・アンド・ペースト・データに、システムが多少の書式設定を行う可能性があります。
 - ディスプレイが SBCS ディスプレイ (DBCS に非対応) であり、データが非表示領域からコピーされた場合、システムは、コピー・アンド・ペースト・データの最初の位置に非表示属性を書き込みます。非表示領域は、非表示属性がコピー・アンド・ペースト・データに対して有効であった領域です。
 - ディスプレイが DBCS ディスプレイであっても DBCS 純粋フィールドに対応していない場合、以下の書式設定が実行されます。

注: DBCS 純粋フィールドのサポートには、画面位置を必要としない拡張 SO/SI 属性のサポートが含まれます。

- コピー・アンド・ペースト・データの最初のバイトが DBCS データであり (SO 文字が前に置かれる)、DBCS データの開始点が DBCS 文字の 2 番目のバイトである場合、コピーの開始点は位置が 1 つ減少します (文字全体が含まれるようにするため)。
- コピー・アンド・ペースト・データの最後のバイトが DBCS データであり、DBCS データの終了点が DBCS 文字の最初のバイトである場合、コピーの終了点は位置が 1 つ増加します (文字全体が含まれるようにするため)。
- コピー・アンド・ペースト・データの最初のバイトが DBCS データである場合 (SO 文字が前に置かれる)、システムは、コピー・アンド・ペースト・データの最初のバイトの前に SO を書き込みます。
- コピー・アンド・ペースト・データの最後のバイトが DBCS データである場合、システムは、コピー・アンド・ペースト・データの最後のバイトの後に SI を書き込みます。
- データが非表示領域からコピーされた場合、システムは、コピー・アンド・ペースト・データの最初の SO/SI 以外の文字上に非表示属性を書き込みます。コピー・アンド・ペースト・データ内のその文字が DBCS 文字である場合、システムは 2 つの非表示属性を最初の 2 つの SO/SI 以外の文字上に書き込みます。

画面上の末端地点標識は、以下のいずれかが発生した場合にリセットされます。

- 4 つのカーソル移動キー以外のキーが押される。システムは、末端地点をリセットせずにユーザーがテキスト・カーソルを移動することを許可しています。
- 別のポインター装置イベントが発生する。
- ホストが画面を更新する。

NWS の電源が遮断されるか、ユーザーが別のコピー・アンド・ペースト操作のコピーの部分を完了する場合のみ、コピー・アンド・ペースト・データは失われます。

2. このマウス・ボタン・イベントがプログラマブル・マウス・ボタン構造化フィールドでプログラムされている場合、そのイベントは前述のように処理されます。

シフト右ボタン押しイベントの処理

システムは以下の機能を、リストされている順序で確認しながら 1 つだけ実行します。

1. このマウス・ボタン・イベントがプログラマブル・マウス・ボタン構造化フィールドでプログラムされている場合、そのイベントは前述のように処理されます。
2. システムが右ボタン押しイベントを受け取り、ユーザーが以前にコピー・アンド・ペースト操作のデータを選択している場合、システムは選択済みデータをテキスト・カーソルの位置で自動入力します。これは、ポインター装置カーソルの位置とは独立して行われます。システムは以下のステップを実行します。
 - テキスト・カーソルが入力フィールドに置かれていない場合、オペレーター・エラー X'0005' が通知されます。
 - コピー・アンド・ペースト・データがディスプレイから主記憶装置に読み取られます。読み取られるデータの長さは、コピー・アンド・ペースト・データの長さ、入力フィールドに収まるデータの長さの短い方です。

注: 継続フィールドは単の入力フィールドと見なされます。

- 非表示属性の後にあるデータは、別の非表示属性が検出されるまで、ブランクに変換されます。
- 属性はブランクに変換されます。
- GUI で使用される文字はブランクに変換されます。
- nul はブランクに変換されます。
- システムはすべての通常のフィールドを検査します (たとえば、数字のみ、英数字のみなど)。これにより、オペレーター・エラーが発生する可能性があります。オペレーター・エラーが通知された場合には、エラー地点以降のデータは貼り付けされません。
- フィールドが上段専用フィールドである場合、データは上段専用になります。
- ディスプレイが DBCS ディスプレイであり、フィールドが DBCS フィールドである場合、以下の処理が実行されます。
 - フィールドが DBCS 専用フィールドである場合、コピー・アンド・ペースト・データは SO で始まる必要があります (そうでない場合は、オペレーター・エラー「0092」が通知されます)。DBCS 文字は、フィールド終了 SI が検出されるか、または SI がコピー・アンド・ペースト・データ内で検出されるまで、フィールドに入力されます。SO および SI 文字は貼り付けされません。
 - フィールドが DBCS 択一フィールドである場合、コピー・アンド・ペースト・データは適切な形式でなければなりません。そうでない場合は、オペレーター・エラー 0092 が通知されます。オペレーター・エラー 0092 は、SO 文字が許可されていないか (フィールドが SBCS である場合)、データが SO 文字で始まる必要があるか (フィールドが DBCS である場合) のいずれかであることを示しています。DBCS の貼り付けは、DBCS 専用フィールドに関して前述されている方法と同じです。
 - フィールドが DBCS 純粋フィールドである場合、コピー・アンド・ペースト・データは SO で始まる必要があります (そうでない場合は、オペレーター・エラー「0092」が通知されます)。DBCS 文字は、フィールドの末尾が検出されるか、または SI がコピー・アンド・ペースト・データ内で検出されるまで、フィールドに入力されます。SO および SI 文字は貼り付けされません。
 - フィールドが DBCS 混用フィールドである場合、コピー・アンド・ペースト・データはフィールドに自動入力されます。コピー・アンド・ペースト・データが SO 文字で始まる場合、カーソルが DBCS 文字の下にある場合には SO が貼り付けされます。カーソルが SBCS 文字の下にある

場合、SO は貼り付けされません。コピー・アンド・ペースト・データが SO 文字で始まらない場合、カーソルが DBCS 文字の下にある場合には SI が貼り付けされます。フィールド内の最後の文字が DBCS 文字である場合、システムは、最後の DBCS 文字の後に SI の場所を予約します。

- ユーザーが置換モードで作業しており、貼り付けデータが長すぎて入力フィールドに収まらない場合、フィールドにできるだけ多くのデータが入力され、カーソルが入力フィールドの最後の位置に置かれた後、オペレーター・エラー 0012 が通知されます。
- ユーザーが挿入モードで作業しており、貼り付けデータが入力フィールドの末尾にあるヌルの数より多くの位置を必要としている場合、フィールドにできるだけ多くのデータが入力され、カーソルが最後のエクスポート文字に置かれた後、オペレーター・エラー 0012 が通知されます。

注: コピー・アンド・ペースト・データがリセットされることはありません。ユーザーは、最新のコピー・アンド・ペースト・データを複数回貼り付けることができます。

3. ユーザーが以前にコピー・アンド・ペースト操作のデータを選択していない場合、システムはイベントを無視します。

他のすべてのポインター装置イベントの処理

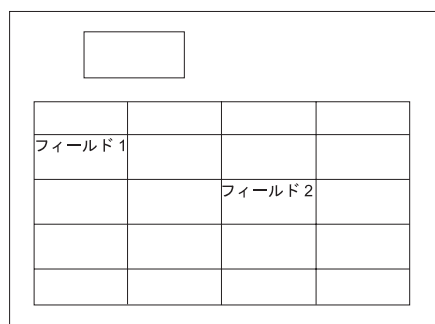
1. マウス・ボタン・イベントがプログラマブル・マウス・ボタン構造化フィールドを使用してプログラムされている場合、そのイベントは前述のように処理されます。
2. そうでない場合は、イベントは無視されます。

グリッド・ライン構造の概要

グリッド・ライン構造には、水平線、縦線、およびボックスが含まれます。それらは、DBCS ディスプレイ装置のみで表示されます。必要なハードウェアの詳細については、228 ページの『グリッド・ライン構造におけるハードウェア要件』を参照してください。グリッド・ライン構造の例は、図 93 に示されています。

さらにグリッド・ライン・キーワードは、以下の機能を実行することを可能にします。

- 指定した長方形内のグリッド・ラインを消去する
- 指定したグリッド・ライン構造を消去する
- カラー、線のタイプなどのグリッド・ライン構造の属性を制御する



RBAHG506-0

図 93. グリッド・ライン構造

グリッド・ライン構造用 DDS の例

図 94の DDS は、226 ページの図 93のグリッド・ライン構造を作成します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     GRDATR((*COLOR BLU) (*LINTYP SLD))
A
A                                     GRDREC
A 96                                GRDRCD
A                                     GRDCLR
A                                     GRDBOX((*POS (3 10 4 20)) +
A                                     (*TYPE PLAIN) +
A                                     (*COLOR RED) (*LINTYP THK) +
A                                     (*CONTROL &CNTL1))
A
A                                     GRDBOX((*POS (10 2 10 60)) +
A                                     (*TYPE HRZVRT 2 15) +
A                                     (*CONTROL &CNTL2))
A                                     CNTL1          1S 0P
A                                     CNTL2          1S 0P
A
A                                     R REC1
A
A                                     FIELD1          7A B 15 5CHGINPDFT
A
A                                     FIELD2          7A B 18 35CHGINPDFT
A
```

図 94. グリッド・ライン構造用 DDS の例

注:

1. グリッド・ライン構造が入出力フィールドのすぐ下に定義される場合には、そのフィールドに CHGINPDFT キーワードを追加することをお勧めします。
2. 一般的に、グリッド・ライン構造は、ディスプレイ・ファイルにコード化されている順序に書き込まれます。
3. ディスプレイ・ファイルが作成されるか、またはそれが非拡張ディスプレイ (ENHDSP *NO) に変更される場合、グリッド・ライン・レコードは画面に書き込まれません。

レコード GRDREC がオプション標識 96 をオンにして書かれている場合、GRDCLR キーワードにより、画面のすべてのグリッド・ライン構造が消去されます。その後残りのキーワードは、コード化されている順序で処理されます。プログラム・システム間フィールド CNTL1 に 1 が入っている場合、最初の GRDBOX キーワードは無視されます。CNTL1 に 0 が入っている場合、PLAIN ボックス・グリッドが表示されます。このボックスは、3 行、10 桁目で始まります。その縦の長さは 4 行、幅は 20 桁です。ボックスは太い線のタイプを持ち、赤色です。CNTL1 に -1 が入っている場合、定義された位置にあるグリッド・ライン構造は消去され、画面上の他のすべてのグリッド・ライン構造には影響がありません。

プログラム・システム間フィールド CNTL2 に 1 が入っている場合、2 番目の GRDBOX キーワードは無視されます。CNTL2 に 0 が入っている場合、水平および縦方向に罫線が引かれたボックスが表示されます。このボックスは、10 行、2 桁目で始まります。その縦の長さは 10 行、幅は 60 桁です。ボックスには 2 行ごとに水平罫線があり、15 桁ごとに垂直罫線があります。GRDBOX キーワードには定義済み属性がないため、属性のデフォルトは、ファイル・レベルで GRDATR キーワードに定義されているものになります。これらの属性は実線の線タイプで、カラーは青です。CNTL1 に -1 が入っている場合、定義された位置にあるグリッド・ライン構造は消去され、画面上の他のすべてのグリッド・ライン構造には影響がありません。

グリッド・ライン構造はレコードの中で、他のデータ・フィールドとは別個に定義されます。グリッド・ライン・レコードは、他のすべてのデータ・レコードとは独立して表示されます。さらにグリッド・ライン・

レコードは、他のデータ・レコードとは独立して消去されます。装置ファイルに DFRWRT(*YES) が含まれている場合、書かれているすべてのレコードは、GET 操作が完了するまでバッファに入れられます。これには、すべてのグリッド・ライン・レコードが含まれます。グリッド・ライン・レコードには入力が発生しないため、READ または PUTGET 操作はグリッド・ライン・レコードに対して実行できません。FRCDTA キーワードはグリッド・ライン・レコードに対して許可されています。この場合、書かれているグリッド・ライン・レコードは、DFRWRT キーワードに関係なく、即時に表示されます。

グリッド・ライン構造およびウィンドウ

GUI ウィンドウが表示される時にグリッド・ライン構造が画面にある場合、ウィンドウを表示する前に、制御装置はウィンドウの下のグリッド・ラインをすべて除去します。GUI ではないウィンドウ (たとえば、UIM ヘルプ・ウィンドウ) が表示される時にグリッド・ライン構造が画面にある場合、すべてのグリッド・ラインは画面から除去されます。ウィンドウが画面から除去される時、すべてのグリッド・ライン構造を含め、画面が復元されます。

グリッド・ライン・レコードは、ウィンドウを定義したり、ウィンドウ参照レコードを指定したりすることができます。グリッド・ライン構造がウィンドウで定義される場合、すべての開始行および開始桁パラメータは、ウィンドウの始まりを起点とします。これには、DDS で定義される開始行および開始桁パラメータ、およびプログラム・システム間フィールドで実行時に設定される開始行および開始桁値の両方が含まれます。グリッド・ライン構造は、ウィンドウの外側に表示させることができます。縦の長さ、幅、および長さパラメータは、画面サイズと比較して大きすぎる場合には、画面の最後で切り捨てられます。

ウィンドウが表示されている場合、グリッド・ライン・レコードを基本画面に書き込むと、ウィンドウ・レコードに USRRSTDSP キーワードが定義されているのでないかぎり、ウィンドウが除去されます。基本画面は、グリッド・ライン構造を表示する前に復元されます。

グリッド・ライン構造におけるハードウェア要件

グリッド・ラインのサポートには、DBCS ハードウェアが必要です。このハードウェアに日本語 DOS を実行する機能があることが必要です。日本語 DOS は、以下の PS/55 システムでサポートされます。

- 5523-S、5523-V、8551-S
- 5535-S (ラップトップ)
- 5530-T/V/W、5541-T、5551-S/TV/W、5561-W (デスクトップ)
- 5571-T/V、5580-W/Y (床置き式のモデル)

注: PS/1 および PS/2[®] システムは、日本語 DOS をサポートしません。

デスクトップおよび床置き式の PS/55 システムでは、日本語キーボード (5576-001/002/003/A01) および DBCS 対応ディスプレイ (5574) が必要です。5530 システムには、内蔵ディスプレイがあります。平衡型通信アダプターとして、5250 アダプター/A (ID#65X1092) が必要です。

i5/OS システムは、接続されたディスプレイがグリッド・ライン構造を表示可能であることを検知することができます。グリッド・ライン構造をサポートしないディスプレイにグリッド・ライン・レコードが書き込まれる場合、レコードは無視されます。ただし、グリッド・ライン・レコード上のウィンドウ・キーワードは処理されます。

プリンターが DBCS をサポートしている場合には、Print Screen キーを使用してグリッド・ライン構造 (フィールド・レベル以外のファイル) を印刷できます。

HTML タグの挿入

ワールド・ワイド・ウェブ (または短縮して Web (ウェブ)) は、インターネットで使用可能な膨大な量の情報へのアクセスを提供する、グラフィカル・インターフェースです。Web は、テキストおよび非テキスト・オブジェクト (ビデオ、音、グラフィックスなど) を含む文書にインターネット・ユーザーがアクセスすることを可能にします。文書には他の文書への「リンク」(ハイパーリンク) を含めることができ、そのリンク先の文書にも他の文書へのリンクが含まれることがあります。ハイパーリンクを含む Web 文書内のテキストのことを、ハイパーテキストと呼びます。すべての文書は他の文書にリンクできるため、文書から文書への参照の連鎖は実質的に無限です。

Web 上の文書を調整して、リンクで示される情報にエンド・ユーザーが任意でアクセスできるようにすると、さまざまなソースからのマルチメディア情報を提供することができます。

i5/OS 5250 ワークステーション・ゲートウェイの DDS サポートを使用すると、既存のアプリケーションを変更して、それらがワールド・ワイド・ウェブを介してインターネットを使用できるように設定できます。i5/OS 5250 ワークステーション・ゲートウェイは、すべての 5250 データ・ストリームを HTML (ハイパーテキスト・マークアップ言語) 文書に変換し、その文書を i5/OS からエクスポートします。ディスプレイ・ファイル・ソースへのわずかな変更だけでクライアント Web ブラウザーのグラフィック機能を活用できるようにする HTML タグを、ディスプレイ・ファイルに挿入することができます。

HTML キーワードには、DDS ファイル内で行/桁位置が与えられます。しかし、行および桁の配置は HTML 文書では意味を持ちません。ディスプレイ・ファイル内の行および桁により、作成される HTML 文書内の HTML タグの順序が決定されます。たとえば、2 行/4 桁目にある HTML キーワードは、2 行/6 桁目の位置を持つ HTML キーワードの前に表示されます。まったく同じ行および桁の位置を持つ HTML キーワードは、DDS ファイルで定義されているものと同じ順序で HTML 文書に配置されます。

HTML フィールドのオーバーラップの解決

以下の例は、以下のコーディングを使用するとき HTML が表示される場所を示しています。

例 1

HTML フィールドの開始桁は出力フィールドの 2 つ前です。HTML はそのフィールドの前に表示されません。

```
A 01      FLD1A      20  0 15  7DFTVAL('Output Field')
A 01      15  5HTML('<p>HTML code')
```

will result in:

```
HTML code
Output Field
```

例 2

HTML フィールドの開始桁は出力フィールドの 1 つ前です。これは、HTML が出力フィールドの属性バイトから開始することを意味します。HTML はそのフィールドの前に表示されます。

```
A 01      FLD1A      20  0 15  6DFTVAL('Output Field')
A 01      15  5HTML('<p>HTML code')
```

次のような結果になります。

```
HTML code
Output Field
```

例 3

HTML フィールドの開始桁は出力フィールドと同じです。HTML はそのフィールドの先頭文字の前に表示されます。

```
A 01      FLD1A      20  0 15 6DFTVAL('Output Field')
A 01                                  15 6HTML('<p>HTML code')
```

次のような結果になります。

```
HTML code Output Field
```

例 4

HTML フィールドの開始桁は出力フィールドの開始桁の 1 つ後です。HTML は、出力フィールドの先頭文字の後に表示されます。

```
A 01      FLD1A      20  0 15 6DFTVAL('Output Field')
A 01                                  15 7HTML('<p>HTML code')
```

次のような結果になります。

```
0HTML codeoutput Field
```

例 5

HTML フィールドの開始桁は出力フィールドの終了桁の 1 つ後です。これは、HTML フィールドが終了属性とオーバーラップすることを意味します。HTML は、出力フィールドの最終文字の後に表示されます。

```
A 01      FLD1A      20  0 15 6DFTVAL('Output Field')
A 01                                  15 27HTML('<p>HTML code')
```

次のような結果になります。

```
Output FieldHTML code
```

例 6

HTML フィールドの開始桁は出力フィールドの終了桁の 2 つ後です。HTML は出力フィールドの後に表示されます。

```
A 01      FLD1A      20  0 15 6DFTVAL('Output Field')
A 01                                  15 28HTML('<p>HTML code')
```

次のような結果になります。

```
Output Field
HTML code
```

注:

1. HTML フィールドと DDS フィールドのマージは、入力フィールドでは発生しません。マージは出力フィールドのみで発生します。
2. 装置照会により、装置が i5/OS 5250 ワークステーション・ゲートウェイ仮想端末であることが示された場合に、HTML タグがデータ・ストリームに挿入されます。そうでない場合、通常のディスプレイの場合には、HTML タグは無視されます。

プログラミングの例

以下の例が示すように、IMG HTML キーワードを既存のディスプレイ・ファイルに追加して、画面にグラフィック・イメージを表示することができます。

図 95は、HTML キーワードを追加する前の DDS の例を示しています。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A
A FILE LEVEL KEYWORDS
A
A          DSPSIZ(24 80 *DS3)
A          CA03(03)
A
A          R REC1
A          CA01(01)
A          CA02(02)
A          5 30'Description'
A          DSPATR(HI)
A          DSPATR(UL)
A          5 13'Item'
A          DSPATR(HI)
A          DSPATR(UL)
A          5 65'Price'
A          DSPATR(HI)
A          DSPATR(UL)
A          FLD001      10A 0 6 13
A          FLD002      25A 0 6 30
A          FLD003       6A 0 6 65
A          1 36'Catalog'
```

図 95. HTML キーワードを追加する前の DDS コーディング

232 ページの図 96は、HTML キーワードを追加した後の DDS の例を示しています。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A
A FILE LEVEL KEYWORDS
A
A          DSPSIZ(24 80 *DS3)
A          CA03(03)
A
A          R REC1
A          CA01(01)
A          CA02(02)
A          5 30'Description'
A          DSPATR(HI)
A          DSPATR(UL)
A          5 13'Item'
A          DSPATR(HI)
A          DSPATR(UL)
A          5 65'Price'
A          DSPATR(HI)
A          DSPATR(UL)
A          FLD001      10A  0  6 13
A          FLD002      25A  0  6 30
A          FLD003      6A   0  6 65
A          7 2HTML('<img src="http://www.ice.com +
A                  /bin/sundae.gif')
A          1 36'Catalog'

```

図 96. HTML キーワードを追加した後の DDS コーディング

i5/OS 5250 ゲートウェイ・ディスプレイ装置では、図 96の DDS ソースを使用した場合には次のグラフィック・イメージが表示されます。

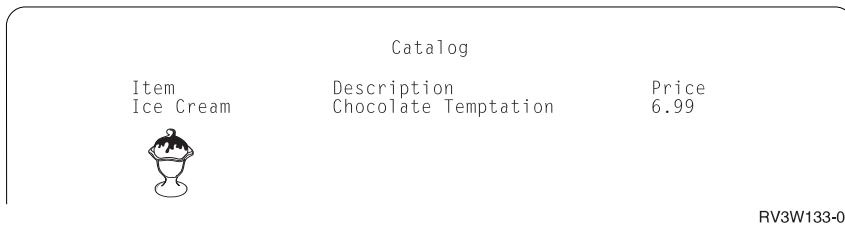


図 97. i5/OS 5250 ゲートウェイ・ディスプレイでのグラフィック・イメージ

制約事項

5250 ゲートウェイは、F1 機能キーを一般ヘルプとして予約します。F1 は、他で使用するために再定義できません。

Web ブラウザーの制限のため、5250 ゲートウェイは、DDS で許可されているグラフィック構造の多くをサポートしていません。

以下のキーワードは許可されていますが、サポートされていません。それらを表示しようとすると、予測不能の結果になります。

ウィンドウ	(WINDOW キーワード)
メニュー・バー	(MNUBAR キーワード)
プルダウン	(PULLDN キーワード)
選択リスト	(SFLMLTCHC/SFLSNGCHC キーワード)
選択フィールド	(SNGCHCFLD/MLTCHCFLD キーワード)
押しボタン	(PSHBUTTON キーワード)
スクロール・バー	(SCRBAR キーワード)
継続入力フィールド	(CNTENTFLD キーワード)

第 7 章 ディスプレイ・ファイルとディスプレイ・ファイル属性のオーバーライド

ファイル名、ファイルに関連した装置名、またはファイルの他の属性の一部を一時的に変更するために、オーバーライドを使用することができます。オーバーライドを使用すると、プログラムを再コンパイルせずに、プログラムの機能の仕方に小さな変更を加えたり、プログラムが処理するデータを選択することができます。

オーバーライドを使用するかしないかの判別

オーバーライドの以下のプロパティは、オーバーライドが、実行したいタスクに適切であるかを判別する上で役立ちます。

- オーバーライドは、それを出したジョブ、プログラム、またはディスプレイ装置セッションに対してのみ有効です。オーバーライドにより、ファイルの属性が永続的に変更されることはありません。
- オーバーライドは、同時に実行中のジョブが他にあっても、それらのジョブに対しては効力がありません。
- 適用されるオーバーライドは、プログラムによりファイルがオープンされる前か、ファイルをオープンするプログラムがコンパイルされる前のいずれかに指定する必要があります。
- オーバーライド用コマンドは、ディスプレイから対話式に入力するか、またはバッチ・ジョブの一部として入力できます。
- オーバーライド・コマンドは、制御言語 (CL) プログラムに含めたり、プログラム QCMDXEC への呼び出しによって他のプログラムから実行したりできます。
- オーバーライドを使用すると、プログラムを再コンパイルせずに、プログラムの機能の仕方に小さな変更を加えたり、プログラムが処理するデータを選択することができます。

HLL プログラムでのファイル属性のオーバーライド

ファイル属性は、以下の結果として構築されます。

ファイル作成コマンド

これらのコマンドは、ファイルが最初に作成される際にファイル属性を構築します。

ファイルを使用するプログラム

コンパイル時に、ユーザー・プログラムでファイル属性の一部を指定することができます。(指定できる属性は、プログラムが書かれた高水準言語により異なります。)

オーバーライド・コマンド

ファイル記述とユーザー・プログラムで指定されたファイル・パラメーターを組み合わせることにより以前に構築されたファイル属性を、プログラム実行時に、これらのコマンドでオーバーライドできます。

高水準言語 (HLL) プログラムでファイルをオーバーライドする最も簡単な方式は、ディスプレイ・ファイル・オーバーライド (OVRDSPF) コマンドを使用してファイルの属性の一部をオーバーライドすることです。

例

以下のディスプレイ・ファイル作成コマンドを使用して、DISPLAY33 という名前のディスプレイ・ファイルを作成します。

```
CRTDSPF FILE(QGPL/DISPLAY33) SRCFILE(DDSFIL1) +  
  DEV(STATION1) IGCDTA(*YES) WAITFILE(30) LVLCHK(*NO)
```

ご使用のアプリケーション・プログラムは、ディスプレイ・ファイル DISPLAY33 に対して、ディスプレイ装置に STATION50 を、IGCDTA パラメーター (2 バイト文字の処理を決定する) に *NO を指定します。

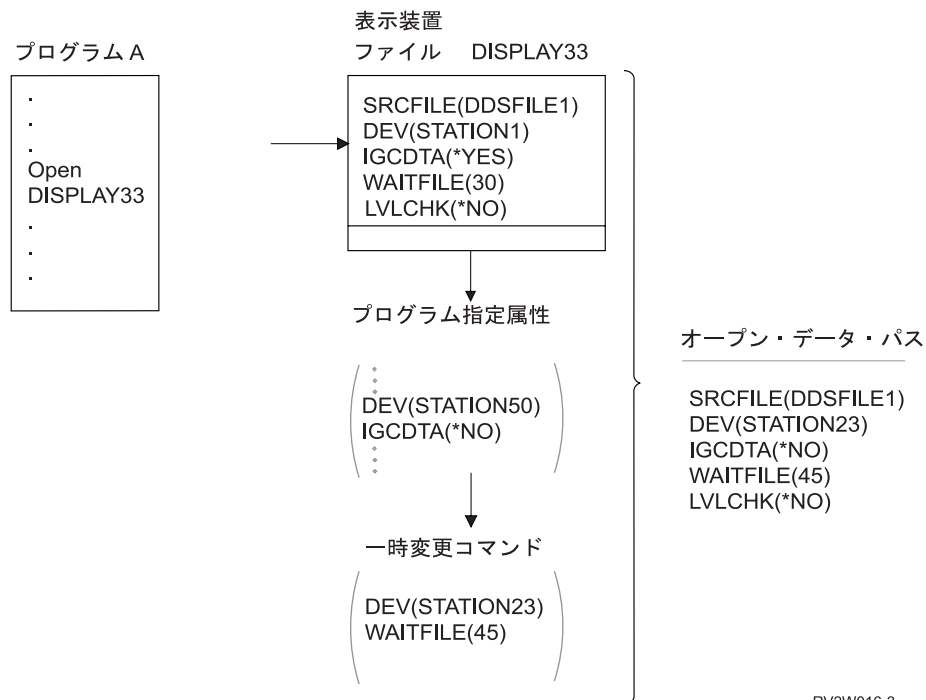
そのアプリケーション・プログラムを実行する前に、ディスプレイ装置を STATION23 に、ファイルの待機時間を 45 秒に変更する必要があります。次のようなオーバーライド・コマンドを使用します。

```
OVRDSPF FILE(DISPLAY33) DEV(STATION23) WAITFILE(45)
```

アプリケーション・プログラムがファイルを開いた時点で、ファイル・オーバーライド、プログラム指定の属性、およびファイル属性が組み合わされて、オープン・データ・パス (ODP) が作成され、この ODP はファイルへのアクセスを管理したりファイルの属性を管理するためにプログラムの実行中に使用されます。ファイル・オーバーライドは、プログラム指定の属性よりも優先されます。プログラム指定の属性は、ファイル指定の属性よりも優先されます。この例では、ファイルが開くと、以下の設定が使用されます。

- ディスプレイ装置: STATION23
- 2 バイト文字処理の値: *NO
- ファイル待機時間: 45 秒
- ファイルのオープン時に、レコード様式のレベル識別コードを検査しない

以下の図は、上記の例を説明しています。



RV2W016-3

HLL プログラムでのファイル名のオーバーライド

高水準言語プログラムにおける別の簡単なファイルのオーバーライド方法として、プログラムで使用するファイルを変更する方法があります。これは、プログラムがコンパイルされた後で、ファイルが移動されたり改名された場合に役立ちます。OVRDSPF コマンドをファイル名のオーバーライドにも使用してください。

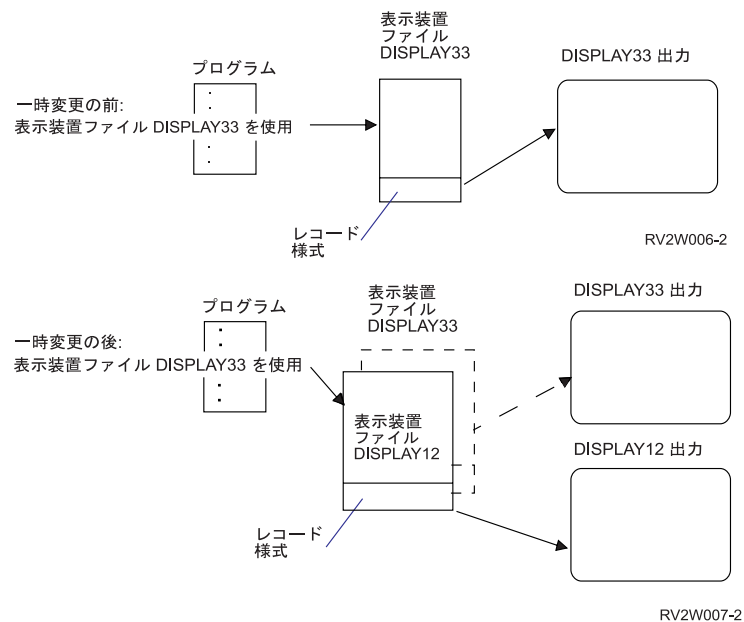
例

DISPLAY33 というディスプレイ・ファイルではなく、DISPLAY12 というディスプレイ・ファイルを使用して、アプリケーション・プログラムからの出力を表示したいとします (アプリケーション・プログラムには DISPLAY33 と指定されている)。プログラムを実行する前に次のように入力します。

```
OVRDSPF FILE(DISPLAY33) TOFILE(DISPLAY12)
```

DISPLAY12 というファイルを使用するためには、あらかじめそのファイルを CRTDSPF コマンドで作成しておかなければなりません。

以下の図は、上記の例を説明しています。



ファイルを、異なるファイル・タイプを持つファイルでオーバーライドしたいかもしれません。たとえば、ディスク・ファイル・オーバーライド (OVRDKTF) コマンドを使用してディスク・ファイルでディスプレイ・ファイルをオーバーライドしたいかもしれません。ファイルを、異なるファイル・タイプを持つファイルでオーバーライドできるかどうかを判断するには、240 ページの『ファイルのリダイレクトを使用したファイル名およびライブラリーまたはファイル・タイプのオーバーライド』 の下の情報を参照してください。異なるファイル・タイプでのオーバーライドについての詳細は、iSeries Information Center のファイルおよびファイル・システムコレクションを参照してください。

HLL プログラムでのファイル名と属性の両方のオーバーライド

高水準言語プログラムにおけるこのファイル・オーバーライドの方式は、単にファイル属性のオーバーライドと、ファイル名あるいはタイプのオーバーライドとを結合したものです。この方式のオーバーライドを使用すれば、プログラムの中で使用するファイルをオーバーライドすることも、オーバーライド・ファイルの属性をオーバーライドすることもできます。ディスプレイ・ファイル名と属性の両方をオーバーライドするには、OVRDSPF コマンドを使用してください。

例

OUTPUT というディスプレイ・ファイルではなく、REPORTS というディスプレイ・ファイルを使用して、アプリケーション・プログラムからの出力を表示したいとします。(アプリケーション・プログラムには OUTPUT と指定されている。) さらに、アプリケーション・プログラムに REPORTS というディスプレイ・ファイルを使用させることに加え、ファイル待機時間を 50 秒にオーバーライドしたいとします。

REPORTS というファイルは、次のコマンドによって作成されるものとします。

```
CRTDSPF FILE(REPORTS) SRCFILE(DDSFIL1) SRCMBR(MEMBER1) WAITFILE(25)
```

プログラムを実行する前に、次のコマンドを入力します。

```
OVRDSPF FILE(OUTPUT) TOFILE(REPORTS) WAITFILE(50)
```

その後アプリケーション・プログラムを呼び出します。ディスプレイ・ファイル REPORTS は、ファイル待機時間が 50 秒で使用されます。

上記の例で使用された単一のオーバーライド・コマンドは、次の 2 つのオーバーライド・コマンドとは等しくありません。

```
オーバーライド 1   OVRDSPF   FILE(OUTPUT)  TOFILE(REPORTS)
オーバーライド 2   OVRDSPF   FILE(REPORTS)  WAITFILE(50)
```

特定のファイルのオープン操作において、呼び出しレベルごとに 1 つのオーバーライドしか適用されません。プログラムが使用するファイルをオーバーライドして、オーバーライド・ファイルの属性もその呼び出しレベルでオーバーライドしたい場合には、1 つのコマンドを使用しなければなりません。2 つのオーバーライドを使用すると、オーバーライド 1 ではディスプレイ・ファイル REPORTS を使用して出力が表示されますが、オーバーライド 2 は無視されます。

プログラムのコンパイル時のオーバーライドの適用

次の 2 つのいずれかの目的がある場合に、プログラムがコンパイルされる時にオーバーライドを適用できます。

- ディスプレイ・ファイルを選択するため
- 入出力操作で使用されるレコード様式をコンパイラーが定義するために使用される外部データ定義を提供するため

ソース・ファイルへのオーバーライドは、他のオーバーライドと同様に処理されます。オーバーライドは、別のファイルまたはデータベース・ファイルの別のメンバーを選択したり、他のファイル属性を変更したりすることができます。

コンパイル中のプログラム内で使用されているファイルが、プログラムで外部記述ファイルとして使用されている場合には、オーバーライドをこれらのファイルに適用することもできます。これらのファイルはコンパイル時にオープンされないため、オーバーライドは通常の方法では適用されません。これらのオーバーラ

イドは、プログラムが入出力操作を使用するためのレコード様式とフィールドを定義するために使用されるファイル名とライブラリーを判別するためだけにコンパイル時に使用されます。オーバーライドで指定される他のすべてのファイル属性は、コンパイル時に無視されます。プログラムのソースで指定されたファイル名が、アプリケーションが必要とするレコード様式を含むファイル名ではない場合にのみ、ファイル・オーバーライドがコンパイル時に活動状態であることが必要です。

コンパイル済みプログラムが実行される時にオープンされるファイル名は、プログラムが実行される時に有効となっているすべてのオーバーライドによって変更された、プログラム・ソースが参照するファイル名により決定されます。コンパイル時に使用されるファイル名は、保持されません。実際にオープンされるファイル内のレコード様式は、プログラムがコンパイルされたときに使用されたレコード様式と互換性のあるものでなければなりません。明らかに、レコードの互換性を確実にするためのもっとも簡単な方法は、コンパイル時に活動状態であったオーバーライドと同じオーバーライドを実行時に活動状態にするという方法です。プログラムが外部記述データを使用し、異なるフィールド・レベル・ファイルが実行時に使用される場合、通常はオーバーライドで LVLCHK(*NO) を指定することが必要になります。詳細は、240 ページの『ファイルのリダイレクトを使用したファイル名およびライブラリーまたはファイル・タイプのオーバーライド』を参照してください。

例

15 秒のファイル待機時間を持つプログラム INVENTORY のソースには、ディスプレイ・ファイル LISTOUT のオープンが含まれているものとします。

```
オーバーライド 1    OVRDBF FILE(RPGSRC) TOFILE(SRCPGMS) MAXRCDLEN(77)
オーバーライド 2    OVRDSPF FILE(OUTPUT) TOFILE(REPORTS)
                    CALL PGM(A)
```

プログラム A

```
オーバーライド 3    OVRDSPF FILE(LISTOUT) TOFILE(OUTPUT)
オーバーライド 4    OVRDBF FILE(RPGSRC) WAITFILE(30)
                    CRTRPGPGM PGM(INVENTORY) SRCFILE(RPGSRC)
                    RETURN
```

```
オーバーライド 5    OVRDSPF FILE(LISTOUT) TOFILE(REPORTS) IGCDDTA(*YES)
                    CALL PGM(INVENTORY)
```

プログラム INVENTORY は、ディスプレイ・ファイル LISTOUT の代わりにディスプレイ・ファイル REPORTS をオープンし、DBCS データを許可します。

プログラム INVENTORY はデータベース・ファイル SRCPGMS から作成 (コンパイル) され、最大 77 文字のレコード長を許可します。オーバーライド 4 (最初に適用) は、オプションのファイル属性をオーバーライドします。オーバーライド 1 (最後に適用) によって、ファイル RPGSRC がデータベース・ファイル SRCPGMS と 77 文字の最大レコード長でオーバーライドされます。

プログラム INVENTORY は、ファイル REPORTS から取得した画面様式で作成されます。オーバーライド 3 (最初に適用) によって、ファイル LISTOUT が OUTPUT でオーバーライドされます。オーバーライド 2 (最後に適用) は、OUTPUT を REPORTS でオーバーライドします。他の属性をここで指定することもできますが、コンパイル時にはレコード様式のみが使用されるため、それは必要ではありません。

実行時にはプログラム A は終了しているため、オーバーライド 3 は活動状態ではなくなります。ですから、オーバーライド 2 は LISTOUT には影響を与えません。しかし、実行時に活動状態であるオーバーラ

イド 5 は、LISTOUT を REPORTS に置換し、DBCS データを許可します。コンパイルと実行時の両方で同じファイルが使用されるため、レベル検査をオンにしておくこともできます。

オーバーライドの削除

オーバーライドを削除したい場合には、オーバーライド削除 (DLTOVR) コマンドを使用することができます。

他のプログラムの呼び出しまたは制御権移動のいずれかを行うアプリケーションで DLTOVR コマンドを使用する場合、オーバーライドは削除される場合とされない場合があります。アプリケーション・プログラムでのオーバーライドの削除についての詳細は、iSeries Information Center の [ファイルおよびファイル・システムコレクション](#) を参照してください。

オーバーライドの表示

すべてのファイル・オーバーライドまたは特定のファイルのファイル・オーバーライドを、DSPOVR (オーバーライド表示) コマンドを使用して表示することができます。

DSPOVR コマンドを使用して、他のプログラムの呼び出しまたは制御権移動のいずれかを行うアプリケーションにより使用されるオーバーライドを表示する場合、どのオーバーライドを表示するかを制御することができます。アプリケーション・プログラムで使用されるオーバーライドの表示についての詳細は、iSeries Information Center の [ファイルおよびファイル・システムコレクション](#) を参照してください。

ファイルのリダイレクトを使用したファイル名およびライブラリーまたはファイル・タイプのオーバーライド

ファイルのリダイレクトとは、オーバーライドを使用してファイル名およびライブラリーまたは処理されるファイルのタイプを変更することを指します。たとえば、あるディスプレイ・ファイルを別のものに置換したり、ICF ファイルの使用からディスプレイ・ファイルの使用に変更することができます。システムはファイルのリダイレクトをサポートする場合としない場合があります。システムがオーバーライドを処理する方法の規則については、243 ページの『オーバーライドを無視または制限するコマンドの認識』を参照してください。

同じファイル・タイプでのファイルのオーバーライド

プログラムで使用されるファイルを同じタイプの別のファイルに置換すると、新しいファイルは元のファイルと同じ方法で処理されます。フィールド・レベル・ファイル、または外部記述データを含む他のいずれかのファイルがリダイレクトされる場合、通常、LVLCHK(*NO) を指定するか、プログラムを再コンパイルするかのいずれかを行うことが必要となります。レベル検査をオフにした場合でも、ファイル内のレコード様式とプログラム内のレコードに互換性を持たせることが必要です。形式に互換性がない場合、結果を予測することができません。

異なるファイル・タイプでのファイルのオーバーライド

異なるファイル・タイプへ変更する場合、装置依存の特性は無視され、レコードは順次に読み書きされます。いくつかの装置パラメーターを、新しい装置ファイルまたはオーバーライドの中で指定する必要があります。その他の装置パラメーターについては、デフォルトが取られます。特定のリダイレクトの組み合わせの影響については、この節で後述します。

最終ファイル・タイプとは異なるファイル・タイプのオーバーライドで指定されたすべての属性は、無視されます。パラメーター SPOOL、SHARE、および SECURE には、この規則は当てはまりません。これらは、装置タイプに関係なく、ファイルに適用されるすべてのオーバーライドで受け入れられます。

リダイレクトの組み合わせによっては、装置の特定の特性のために特殊な問題が生じる場合があります。特に以下の点に注意してください。

- 保管ファイルでは、ファイルのリダイレクトが勧められていません。
- 複数の装置 (MAXDEV または MAXPGMDEV > 1) を使用するディスプレイ・ファイルおよび ICF ファイルがリダイレクトできるのは、ディスプレイ・ファイルまたは ICF ファイルだけです。
- ディスプレイ・ファイルを他のファイル・タイプに、または別のファイル・タイプをディスプレイ・ファイルにリダイレクトするには、入力専用または出力専用フィールドがある場合はオーバーライドを活動状態にして、プログラムを再コンパイルすることが必要になります。ディスプレイ・ファイルでは、これらのフィールドを使用しないレコード・バッファからこれらのフィールドは省かれますが、他のファイル・タイプではそのようなことはないのである。

表 22では、有効なファイルのリダイレクトを要約しています。

表 22. ファイルのリダイレクト

宛先ファイル	元のファイル					
	プリンター	ICF	ディスク	ディスプレイ	データベース	テープ
プリンター	O*	O	O	O	O	O
ICF	O	I/O O I	O I	I/O O I	O I	O I
ディスク	O	O I	O I	O I	O I	O I
ディスプレイ	O	I/O O I	O I	I/O O I	O I	O I
データベース	O	O I	O I	O I	O I	O I
テープ	O	O I	O I	O I	O I	O I
凡例: I = 入力ファイル O = 出力ファイル I/O = 入出力ファイル * = 異なるタイプのプリンターへのリダイレクト						

この図表を使用するには、オーバーライドされるファイル・タイプを「元のファイル」欄で見つけ、オーバーライドするファイル・タイプを「宛先ファイル」欄で見つけます。交差した所では I か O またはその両方の指定がありますが、これらは、入力ファイルまたは出力ファイルとして使用したときに、これらの 2 つのファイル・タイプに対して置換が有効であることを意味しています。

たとえば、ディスク出力ファイルをテープ出力ファイルでオーバーライドしたり、ディスク入力ファイルをテープ入力ファイルでオーバーライドしたりすることができます。図表では、ファイル・タイプの置換のみを示しています。すなわち、入力ファイルを出力ファイルでオーバーライドすることによりプログラムの機能を変更することはできません。

以下の図表は、ディスプレイ・ファイルに関するリダイレクトの各組み合わせに対して取られる特定のデフォルトおよび無視されるものについて説明しています。

表 23. ファイルのリダイレクトの組み合わせ

リダイレクト元	リダイレクト先	取られる特定のデフォルトおよび無視されるもの
プリンター	ディスプレイ	レコードは、前のレコードをオーバーレイしたレコードでそれぞれディスプレイに書き込まれます。プログラム記述ファイルでは、Enter キーを使用してそれぞれのレコードを要求できます。プリンター制御情報は無視されます。
ICF 入力	ディスプレイ	レコードは、一度に 1 つずつディスプレイから取得されます。レコードごとにデータを入力し、レコードが完成したら Enter キーを押してください。
ICF 出力	ディスプレイ	レコードは、前のレコードをオーバーレイしたレコードでそれぞれディスプレイに書き込まれます。
ICF 入出力	ディスプレイ	入力レコードは、一度に 1 つずつディスプレイから取得されます。レコードごとにデータを入力し、レコードが完成したら Enter キーを押してください。出力レコードは、それぞれのレコードで直前の入力または出力レコードをオーバーレイしてディスプレイに書き込まれます。入出力レコードは、本質的に互いに独立しており、任意の方法で結合させることができます。
ディスク入力	ディスプレイ	レコードは、一度に 1 つずつディスプレイから取得されます。レコードごとにデータを入力し、レコードが完成したら Enter キーを押してください。非フィールド・レベルのディスプレイ・ファイルを指定する必要があります。ディスク・ラベル情報は無視されます。
ディスク出力	ディスプレイ	レコードは、前のレコードをオーバーレイしたレコードでそれぞれディスプレイに書き込まれます。Enter キーを使用して、それぞれの出力レコードを要求できます。
ディスプレイ入力	ICF ディスク	レコードは、一度に 1 つずつ ICF ファイルから取得されます。レコードは順次に取得されます。ディスク・ラベル情報を、ディスク・ファイルまたはオーバーライド・コマンドで提供しなければなりません。
	データベース テープ	入力レコードが取得されます。レコードは順次に取得されます。テープ・ラベル情報を、テープ・ファイルまたはオーバーライド・コマンドで指定しなければなりません。
ディスプレイ出力	ICF データベース ディスク	レコードは、一度に 1 つずつ ICF ファイルに書き込まれます。レコードは順次にデータベースに書き込まれます。ディスクに書き込まれるデータの量は、ディスクの交換タイプにより異なります。ディスク・ラベル情報を、ディスク・ファイルまたはオーバーライド・コマンドで提供しなければなりません。
	テープ	レコードは順次にテープに書き込まれます。テープ・ラベル情報を、テープ・ファイルまたはオーバーライド・コマンドで指定しなければなりません。
	プリンター	レコードは印刷され、プリンター・ファイルに指定されているように折り返しまたは切り捨てが行われます。

表 23. ファイルのリダイレクトの組み合わせ (続き)

リダイレクト元	リダイレクト先	取られる特定のデフォルトおよび無視されるもの
ディスプレイ入出力	ICF	入力レコードは画面から 1 つずつ取得されます。それぞれのレコードのデータを入力し、レコードが完成したら Enter キーを押してください。出力レコードは、直前の入力または出力レコードをそれぞれオーバーレイしたレコードで画面に書き込まれます。入出力レコードは本質的に相互に独立しており、任意の方法で組み合わせることが可能です。
データベース入力 (順次に処理される)	ディスプレイ	レコードは、一度に 1 つずつディスプレイから取得されます。レコードごとにデータを入力し、レコードが完成したら Enter キーを押してください。非フィールド・レベルのディスプレイ・ファイルを設定する必要があります。
データベース出力 (順次に処理される)	ディスプレイ	レコードは、前のレコードをオーバーレイしたレコードでそれぞれディスプレイに書き込まれます。Enter キーを使用して、それぞれの出力レコードを要求できます。
テープ入力	ディスプレイ	レコードは、一度に 1 つずつディスプレイから取得されます。レコードごとにデータを入力し、レコードが完成したら Enter キーを押してください。非フィールド・レベルのディスプレイ・ファイルを設定する必要があります。テープ・ラベル情報は無視されます。
テープ出力	ディスプレイ	レコードは、前のレコードをオーバーレイしたレコードでそれぞれディスプレイに書き込まれます。Enter キーを使用して、それぞれの出力レコードを要求できます。

オーバーライドを無視または制限するコマンドの認識

以下の一般的に使用されるコマンドは、オーバーライドを完全に無視します。

- ADDLFM
- ADDPFM
- ALCOBJ
- APYJRNCHG
- CHGOBJOWN
- CHGPTR
- CHGSBSD
- CHGXXXF (すべてのファイル変更コマンド)
- CLRPFM
- CLRSAVF
- CPYIGCTBL
- CRTDKTF
- CRTDUPOBJ
- CRTAUTHLR
- CRTSBSD
- CRTTAPF
- DLCOBJ
- DLTF

- DLTAUTHLR
- DSPDBR
- DSPFD
- DSPFFD
- DSPJRN
- EDTOJAUT
- EDTDLOAUT
- ENDJRNP
- GRTOJAUT
- INZPFM
- MOVOBJ
- RGZPFM
- RMVJRNCHG
- RMVM
- RNMOBJ
- RSTUSRPRF
- RVKOBJAUT
- SAVCHGOBJ
- SAVLIB
- SAVOBJ
- SAVPGMPRD
- SAVSAVEDTA
- SAVSYS
- SBMDBJOB
- SIGNOFF
- STRDBRDR
- STRJRNP

オーバーライドは、ルーチン終了ステップまたはジョブ終了処理の一部としてオープンされたシステム・ファイルには適用されません。たとえば、オーバーライドはジョブ・ログ・ファイルには指定できません。場合によっては、システム・ファイル内の何かをオーバーライドする必要がある場合に、オーバーライド・コマンド以外のコマンドを使用して変更できる可能性があります。たとえば、ジョブ・ログ用の出力待ち行列を変更する場合、ジョブ変更 (CHGJOB) コマンド上で **OUTQ** パラメーターを使用してジョブの出力待ち行列の名前を指定すると、サインオフの前に出力待ち行列を変更できます。ジョブ・ログ用のプリンター・ファイルに、出力待ち行列の値として ***JOB** が入っている場合、出力待ち行列はジョブ用に指定されたものとなります。

以下のコマンド上の **SRCFILE** および **SRCMBR** パラメーターは、オーバーライドによる影響を受けません。

- CRTCMD
- CRTICFF
- CRTDSPF

- CRTLF
- CRTPF
- CRTPRTF
- CRTSRCPF
- CRTTBL
- CRTXXXPGM (すべてのプログラム作成コマンド。これらのコマンドもオーバーライドを使用して、コンパイル済みプログラムによりオープンされるファイルを判別します。詳しい説明については、238ページの『プログラムのコンパイル時のオーバーライドの適用』を参照してください。)

OPNQRYF コマンドは、TOFILE、MBR、SEQONLY、LVLCHK、および INHWRT オーバーライド・パラメーターの影響を受けます。

次のコマンドはオーバーライドを許可しますが、MBR の *ALL への変更は許可しません。

- CPYFRMPCD
- CPYTOPCD

以下のコマンドは、コマンドが使用するディスプレイ・ファイルへのオーバーライドの適用を許可しません。コマンドが使用するプリンター・ファイルへのオーバーライドでは、ファイル・タイプまたはファイル名を変更すべきではありません。これらのコマンドが使用するプリンター・ファイルに適用される可能性がある変更にはさまざまな制約事項がありますが、起こりうるすべての仕様の組み合わせで受け入れ可能なレポートがシステムで作成されるという保証はありません。

DMPOBJ および DMPSYSOBJ

前述の制限に加えて、これらのコマンドは、それらがダンプするファイルへのオーバーライドを許可しません。

DSPXXXXXX

すべての表示コマンド。ファイルに関する情報を表示する表示コマンドは、そのファイルへのオーバーライドを許可しません。

DSPIGCDCT および EDTIGCDCT

メッセージ・ファイルはオーバーライドできます。

GO

PRTXXXXXX

すべての印刷コマンド。

QRYDTA

TRCXXX

すべてのトレース・コマンド。

WRKXXXXXX

すべての処理コマンド。

第 8 章 ディスプレイ・ファイルに関するメッセージとエラーの処理

この章では以下の点について扱います。

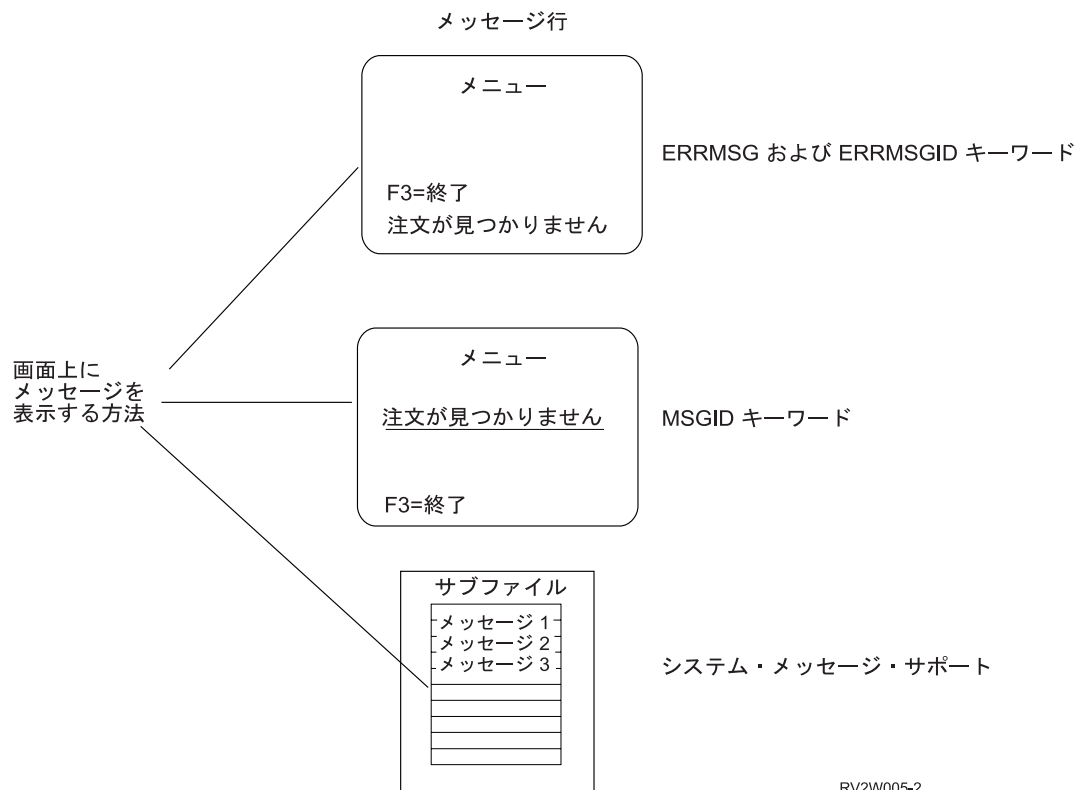
- 独自のメッセージの作成および表示
- システムから送信されるエラー・メッセージの分析

独自のメッセージの作成および表示

i5/OS で独自のメッセージを作成して表示できます。これらのメッセージで、処理エラーが生じたこと、誤った入力データが入力されたこと、または単にシステムが長い要求を処理している間キーボードが一時的にロックされていることを示すことができます。

ディスプレイ・ファイルに関する以下のメッセージ処理機能を指定できます。

- メッセージ行にメッセージを表示する
- サブファイル制御レコードが書き込まれる際にメッセージ行にメッセージを表示する
- メッセージ行を定義する
- ディスプレイ上のフィールドにメッセージを表示する
- プログラム・メッセージ待ち行列中のメッセージを表示する



RV2W005-2

また、以下の操作も行えます。

- システム提供のエラー・サブファイルを使用してメッセージ行にエラー・メッセージを表示する
- システムに、永続的な入出力エラーを受け取ろうとしているジョブを自動的に処理させる

DDS キーワードについて

以下では、DDS キーワードを使用してメッセージの定義と表示を行っています。特定の DDS キーワードについて詳しくは、iSeries Information Center の『DDS』のトピックを参照してください。

メッセージ行にメッセージを表示する

ERRMSG キーワードか ERRMSGID キーワードを使用して、メッセージがメッセージ行に表示されるように指定できます。定数として定義されているメッセージの場合は、ERRMSG キーワードを使用します。この種のメッセージに関するメッセージ・ヘルプはサポートされていません。メッセージ・ファイル中に定義されているメッセージの場合は、ERRMSGID キーワードを使用します。これらのキーワードを使用すると、キーボードはロックするので、ユーザーはディスプレイからメッセージを消去して続行するには取り消しキーを押さなければなりません。

ERRMSG を使用する際には、メッセージを表示する対象のレコードがディスプレイ上に既存でなければなりません。既存でない場合は、ERRMSG 機能は実行されません。

- ERRMSG キーワードを使用してメッセージを表示する際には、そのメッセージはディスプレイのメッセージ行に書き込まれます。普通このメッセージ行は、サーバーとクライアントの接続に使用するソフトウェアに依拠して、画面の下部の行になります。次にユーザーは取り消しキーを押して、ディスプレイからメッセージを消去し、キーボードをアンロックして、入力を続行します。ERRMSG キーワード上の右側にメッセージのテキストを入力します。有効な ERRMSG キーワードのあるレコードを書き込むと、このメッセージが表示されます。通常は、オプション標識を使用して ERRMSG キーワードを有効にします。アプリケーション・プログラムがオプション標識をオンにすると、DDS のこのオプション標識が指定されているキーワードが有効になります。この場合アプリケーション・プログラムは、メッセージを表示する必要が生じるまで、ERRMSG キーワードのオプション標識をオフのままにします。

ERRMSGID キーワードと SFLMSGID キーワードには、メッセージ・データに関するオプション・パラメーター (msg-data) があります。このパラメーターを使用すると、メッセージ・データ (置換テキスト) を含むプログラム-システム間フィールドを定義できます。メッセージ・データ・パラメーターの働きについて詳しくは、iSeries Information Center の『制御言語 (CL)』トピックの『プログラム・メッセージ送信 (SNDPGMMMSG) コマンド』を参照してください。

注: ERRMSG または ERRMSGID キーワードを使用する際には、CRTDSPF または CHGDSPF コマンドに RSTDSP(*YES) を指定する必要があります。指定しない場合は、ディスプレイ・ファイルが中断されると、データは失われます。

サブファイル制御レコードが書き込まれる際にメッセージ行にメッセージを表示する

SFLMSG キーワードか SFLMSGID キーワードを使用して、サブファイル制御レコードが書き込まれる際にメッセージ行にメッセージを表示するように指定できます。メッセージが定数の場合は、SFLMSG キーワードを使用します。メッセージがメッセージ・ファイル中に定義されている場合は、SFLMSGID キーワードを使用します。これらのキーワードに関する制約事項は、ERRMSG および ERRMSGID と同じです。

メッセージ・フィールドを使用してメッセージ行にメッセージを表示する

メッセージ・フィールドを指定できます (位置 38 の M)。この出力フィールドの値は、メッセージ行に表示されます。このフィールドの値は、アプリケーション・プログラムによって出力バッファ中に指定されます。このフィールドの長さは、24 x 80 画面にメッセージが表示される場合は 78 桁を超えることはできず、27 x 132 画面にメッセージが表示される場合は 130 桁を超えることはできません。メッセージ行に関するメッセージ・ヘルプと置換変数はサポートされていません。

メッセージ行にメッセージを表示する場合の優先順位

メッセージ行に表示されるメッセージは、以下の優先順位 (最高は 1) によって決められます。

1. ERRMSG
2. ERRMSGID
3. SFLMSG
4. SFLMSGID
5. DDS 中の位置 38 が M

さらに、MSGLOC キーワードを使用して、メッセージが表示される行を変更できます。指定しないと、メッセージ行はディスプレイ上の最後の行になります。MSGLOC キーワードを使用すると、新しいメッセージ行は、ユーザー定義のメッセージに加えて、妥当性検査エラーおよび無効なキーの場合に表示されるメッセージにも適用されます。

注:

1. MSGLOC キーワードを指定しないと、27 行まで表示できるディスプレイ装置がローカル・ディスプレイ装置コントローラーかリモート 5294 または 5394 コントローラーに接続されている場合に、デフォルト値は以下のようになります。
 - 27 x 132 画面サイズの場合は行 28
 - 24 x 80 画面サイズの場合は行 25
2. デフォルトが使用されているか MSGLOC キーワードに行 25 が指定されているために、24 x 80 モード用に行 25 が指定されている場合に、ディスプレイがメッセージを行 25 に表示できなければ、メッセージは実際には行 24 に表示されます。
3. メッセージ・ヘルプに関する通常のシステム・ディスプレイから、ユーザーは全般ヘルプにアクセスでき (ヘルプ・キーを再度押す)、F10 を使用してジョブ・ログ中のすべてのメッセージを表示できます。ジョブ・リクエスター・ディスプレイ装置以外のディスプレイ装置または複数のディスプレイ・ファイルに関連したディスプレイ装置で使用されるメッセージ・ヘルプ・ディスプレイは、これらの機能を提供しません。

ディスプレイ上のフィールドにメッセージを表示する

MSGID キーワードを使用して、ディスプレイ上のフィールドにメッセージを表示するように指定できます。メッセージが MSGID フィールドより長い場合は、そのメッセージは切り捨てられます。メッセージが MSGID フィールドより短い場合は、そのメッセージはブランクで埋め込まれます。

MSGID キーワードが指定されるフィールドは通常の出力可能フィールドなので、MSGID キーワードを使用すると、キーボードはロックされません。MSGID キーワードに関するメッセージ・ヘルプと置換変数はサポートされていません。

プログラム・メッセージ待ち行列中のメッセージを表示する

SFLMSGRCDD、SFLMSGKEY、および SFLPGMQ キーワードを使用して、プログラム・メッセージ待ち行列に含まれるメッセージを表示するよう指定できます。メッセージに関するサブファイル・サポートを使用すると、一度に複数のメッセージを表示でき、メッセージが表示されている間キーボードはロックされません。プログラム・メッセージ待ち行列中のメッセージが指定されるので、メッセージ・ヘルプと置換テキストの両方ともサポートされます。詳しくは、116 ページの『サブファイルからのエラー・メッセージの表示』を参照してください。メッセージ待ち行列とメッセージの送信に関する情報は、iSeries Information Center の『CL』トピックの『Messages』セクションも参照してください。

サブファイルを使用したエラー・メッセージの表示

ERRSFL (エラー・サブファイル) キーワードを使用して、システム提供のエラー・サブファイルを使って ERRMSG、ERRMSGID、SFLMSG、および SFLMSGID キーワードに関連したエラー・メッセージをメッセージ行に表示するように指示できます。

ERRMSG キーワードと ERRMSGID キーワード、または SFLMSG キーワードと SFLMSGID キーワードに追加して ERRSFL キーワードを使用し、ユーザーがエラー・メッセージのサブファイル全体をロールできるようにすることができます。一度に 1 つのエラー・メッセージが表示されます。ユーザーのプログラムは、エラーのフィールドに関する該当するメッセージを送信するオプション標識をオンに設定して、フィールドの妥当性検査を処理します。システムは、エラー・サブファイル中のエラーのフィールドに関連したメッセージの挿入と、エラー・メッセージの表示を処理します。

ERRSFL キーワードを使用して、システムが妥当性検査を処理する際にエラー・メッセージをエラー・サブファイル中に挿入することもできます。以下の理由で入力データの妥当性検査に失敗すると、エラー・メッセージがエラー・サブファイルに挿入されます。

- DDS キーワード CHECK(M10 M11 VN VNE)、COMP、RANGE、および VALUES
- 浮動小数点操作
- 1 つ以上の小数点以下の桁数があるフィールドに、複数の小数点が入力されている
- 1 つ以上の小数点以下の桁数があるフィールドに入力されている小数点以下の桁数が多すぎるか少ない

エラー・サブファイル全体をページングして、メッセージを表示できます。一度に表示されるエラー・メッセージは 1 つのみです。エラー・サブファイルを表示しても、キーボードはロックされません。エラーのフィールドを訂正する前に取り消しキーを押す必要はありません。

メッセージ行になるよう定義されている行を含むレコード様式が現在表示されている場合は、ERRSFL キーワードは無視されます。

妥当性検査と ERRMSG/ERRMSGID または SFLMSG/SFLMSGID が両方とも画面上で同時に使用されている場合、その結果のエラーはエラー・サブファイル中にすべて同時に示されません。書き込み操作が行われる際に、ERRMSG、ERRMSGID、SFLMSG、および SFLMSGID からのメッセージがエラー・サブファイル中に示されます。データを入力し、妥当性検査エラーが起こると、その後エラー・サブファイルが消去され、この妥当性検査エラーのみ示されます。

ERRSFL キーワードの使用法の例を以下に示します。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                                     MSGLOC(24)
A                                     ERRSFL
A          R RCD1
A          FIELD1          5A B 2 3
A 10                                     ERRMSGID(MSG0001 MSGF1 10 &MDTA1)
A 11                                     ERRMSG('Error Msg 1' 11)
A          FIELD2          10A B 3 3
A 20                                     ERRMSG('Error Msg 2' 20)
A 21                                     ERRMSGID(MSG0002 MSGF1 21)
A 22                                     ERRMSGID(MSG0003 MSGF1 22 &MDTA3)
A          FIELD3          2A B 4 3
A 30                                     ERRMSGID(MSG0004 MSGF1 30)
A          MDTA1          78A P
A          MDTA3          78A P
A
A
A
A
A
A

```

図 98. ERRSFL キーワードに関するサンプル DDS ソース

この例では、RCD1 が現在ディスプレイ上にあると想定しています。入力データの妥当性検査の際に、ご使用のプログラムが複数のエラーを検出し、オプション標識 11、21、および 30 をオンに設定します。以後の出力操作で、以下のようになります。

- FIELD1、FIELD2、および FIELD3 が反転表示される。
- カーソルが位置 2、3 (FIELD1 の先頭) に置かれる。
- キーボードはロックされない。
- エラー・サブファイルが行 24 に表示される。このサブファイルには 3 つのレコード Error Msg 1、MSG2 メッセージ、および MSG4 メッセージが含まれます。ユーザーはメッセージ全体をページングできます。また、カーソルをメッセージ行に置いてヘルプ・キーを押すと、MSG2 または MSG4 に関するメッセージ・ヘルプを表示できます。ユーザーが MSG2 または MSG4 に関するメッセージ・ヘルプを表示しようとする際に、カーソルがメッセージ行に置かれていないと、メッセージ CPF9897 が表示され、ヘルプ情報を使用できないことが示されます。
- Error Msg 1 に関するメッセージ行でヘルプ・キーを押すと、メッセージ CPF9897 に関するメッセージ・ヘルプが表示される。このメッセージ・ヘルプは、メッセージ・ヘルプを使用できないことを説明します。
- 以後の RCD1 のディスプレイでは、以前のエラー・メッセージのエラー・サブファイルは消去される。

ディスプレイ・ファイル中で SFLMSGID キーワードを定義する DDS の例を以下に示します。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A                               MSGLOC(24)
A                               ERRSFL
A           R SFLRCD1           SFL
A           SFL1               5A B 2 3
A           R CTRLRCD1         SFLCTL(SFLRCD1)
A                               SFLPAG(1)
A                               SFLSIZ(15)
A                               SFLDSP SFLDSPCTL
A 10                             SFLMSGID(SFL0001 USERMSGF 10)
A 11                             SFLMSG('Error Msg 1' 11)
A           R SFLRCD2           SFL
A           SFL2               7A 0 4 3DFT('FIELD 2')
A           R CTRLRCD2         SFLCTL(SFLRCD2)
A                               SFLPAG(1)
A                               SFLSIZ(15)
A                               SFLDSP SFLDSPCTL
A 20                             SFLMSG('Error Msg 2' 20)
A 21                             SFLMSGID(SFL0002 USERMSGF 21)
A
A

```

図 99. SFLMSGID キーワードに関するサンプル DDS ソース

ディスプレイ上に CTRLRCD1 および CTRLRCD2 があると想定します。入力データの妥当性検査の際に、ご使用のプログラムが複数のエラーを検出し、オプション標識 11 および 21 をオンに設定します。以後の出力操作で、以下ようになります。

- カーソルが位置 2、3 (SFLRCD1 の先頭) に置かれる。
- キーボードはロックされない。
- エラー・サブファイルが行 24 に表示される。サブファイルには 2 つのレコード Error Msg 1 および SFL0002 のメッセージが含まれます。ユーザーはメッセージ全体をページングできます。また、カーソルをメッセージ行に置いてヘルプ・キーを押すと、SFL0002 に関するメッセージ・ヘルプを表示できます。Error Msg 1 に関するメッセージ行でヘルプ・キーを押すと、メッセージ CPF9897 に関するメッセージ・ヘルプが表示され、ヘルプ情報を使用できないことが説明されます。
- 以後の CTRLRCD1 および CTRLRCD2 のディスプレイでは、以前のエラー・メッセージのエラー・サブファイルは消去される。

SFLEND キーワードは、ERRSFL サブファイル制御レコードに指定します。ユーザーがファイルの上部または下部を超えてロールしようとする時、エラーが出され、キーボードがロックされます。

以下の考慮事項は ERRSFL キーワードに適用されます。ただし、ERRMSG キーワードと ERRMSGID キーワード、および SFLMSG キーワードと SFLMSGID キーワードと併用する場合は除きます。

- エラー・サブファイルが表示される (つまり、エラー状態のフィールドが 1 つ以上ある) 場合、ロール・キーを押すと、その時点のカーソル位置や、他のサブファイルが現在表示されているかどうかにかかわらず、エラー・サブファイルがロールされる。
- 妥当性検査からのエラーを処理する際に、有効なコマンド・キー、ロール・キー、または Enter キーを押してエラーが起きるたびに、エラー・サブファイルが作成される。
- 有効なコマンド・キーまたは Enter キーを押すと、最初のエラー状態のフィールドに関するメッセージが常に表示される。
- エラー・サブファイルがロールされ、ロール・キーを押して表示されたメッセージに対応するフィールドが有効な場合は、エラー・サブファイルと最初のエラー状態のフィールドのメッセージが表示される。

- エラー・サブファイルがロールされ、ロール・キーを押して表示されたメッセージに対応するフィールドが無効な場合は、エラー・サブファイルが次のエラー状態のフィールドにロールされる。
- エラー・サブファイル全体をロールする際には、表示されるエラー・メッセージに対応するフィールドにカーソルが置かれる。
- エラー状態のフィールドがない場合にロール・キーを押すと、サブファイルに対するロールが試行され、ロール・エラー・メッセージが表示される (ロールできる他の領域が最初に検出されないことが前提になっています)。サブファイルのロールの詳細については、95 ページの『第 4 章 サブファイルを使用したレコードのグループの表示』を参照してください。
- プログラムがレコードを書き込み、ユーザーが無効なデータを入力した後でロール・キーを押すと、エラー・サブファイルが作成され、最初のエラー状態のフィールドに関するエラー・メッセージが表示される。

メッセージに関するアラームの鳴動

活動状態の ERRMSG、ERRMSGID、SFLMSG、または SFLMSGID キーワードが検出された際や、妥当性検査メッセージが表示された際にアラームを鳴らすには、MSGALARM キーワードを指定します。このキーワードはファイル・レベルかレコード・レベルで使用できます。アラーム音は短時間のみ鳴ります。

MSGALARM キーワードは、サブファイル制御レコードを含む、ディスプレイ・ファイル中のどのレコードでも使用できます。また、MSGALARM キーワードを ERRSFL キーワードと併用することもできます。

注: 同じレコード様式で MSGALARM キーワードと ALARM キーワードを指定して活動状態にすると、アラームは 1 回のみ鳴ります。

SFLMSG キーワードか SFLMSGID キーワードが活動状態の場合に、MSGALARM キーワードと ALARM キーワードを両方とも指定すると、アラームが鳴ります。メッセージが表示されない場合に、アラームの鳴動を担当するのは、MSGALARM キーワードではなく ALARM キーワードです。

ディスプレイ装置での永続的な入出力エラーの自動処理

装置回復処置ジョブ属性を使用すると、ディスプレイ装置上の永続的な入出力エラーを自動的に処理できます。ジョブ変更 (CHGJOB) コマンドの装置回復処置 (DEVRCYACN) パラメーターは、永続的な入出力エラー (ディスプレイ装置の電源がオフなど) を受け取ろうとしているジョブに関するシステム・アクションを指示します。以下のパラメーター値を使用できます。

***DSCMSG** 入出力エラーが起これば、システムはジョブ切り離し (DSCJOB) コマンドを実行してジョブを中断します。ユーザーは、同じディスプレイに再びサインオンするまで切り離されたままになります。入出力操作の直後にジョブが再開されます。ディスプレイ・データ管理機能はメジャー/マイナー戻りコードを 83E1 に設定し、CPF509F メッセージをプログラム・メッセージ待ち行列に送信します。これらのことは、ディスプレイが消去されており、再び使用可能になっていることを示します。

***DSCENDRQS** このオプションも入出力エラーが起これば DSCJOB コマンドを実行しますが、ジョブが再開される際に、システムは要求終了 (ENDRQS) コマンドを実行して、最新の要求処理プログラムに制御を与えます。

***ENDJOB** 入出力エラーが起これば、ジョブは自動的に終了し、優先順位が下げられて、システム・リソースが節約されます。

*ENDJOBNO LIST

この場合も、ジョブは自動的に終了し、優先順位が下げられて、システム・リソースが節約されます。しかし、このジョブに関するジョブ・ログは作成されません。

入出力エラーが起きた際に (特に、同時に活動状態になっている複数のジョブに関する通信回線に障害が起きた際に)、装置回復処置ジョブ属性を使用すると、システム・リソースを節約できます。さらに、装置回復処置ジョブ属性により、入出力エラーのコーディングのジョブが容易になります。ディスプレイ装置が回復されると、*DSCMSG 値と *DSCENDRQS 値を使用するとユーザー・アプリケーションはエラーが起きた時点から続行できます。*DSCMSG の場合、入出力操作の後でメジャー/マイナー戻りコードが 83E1 であれば、プログラムはディスプレイがブランクであることを想定する必要があります。プログラムは分岐して、ディスプレイに対する最初の入出力操作が行われた時点に戻らなければなりません。*DSCENDRQS の場合、ジョブに再び接続する際に要求処理プログラムは制御を受け取ります。この操作は、システム要求メニューのオプション 2 を選択することと似ています。エラー状態の装置に対する入出力操作が実行されないと、*MSG または *DSCMSG に関するエラー回復は実行されません。他のすべての値の場合は、エラーが起こると直ちに回復が行われます。

装置回復処置ジョブ属性は、*REQUESTER ディスプレイ装置専用で、バッチ・ジョブ、パススルー・ジョブ、またはワークステーション機能ジョブには適用されません。

システムから送信されるエラー・メッセージの分析

以下に、アプリケーション・プログラムの操作中に検出される可能性のあるエラー条件と、プログラム自体の内部で行える、これらの条件への対処を試行するサービス提供について説明します。iSeries Information Center の『CL』トピックの『Debugging CL programs and procedures』セクションには、デバッグ機能を使用して、アプリケーション・プログラムで検出された予期しないエラーを解決する方法が説明されています。また、『トラブルシューティング』トピックには、システム・エラーとハードウェア障害の分析と報告に使用できるプログラムについて説明されています。

ファイルを開く際、プログラム装置が獲得されるか解放される際、ファイルに対する入出力操作中、およびファイルを閉じる際に、エラーが検出されることがあります。該当する場合には、システムは最大で再試行の限度まで、失敗した操作の実行を自動的に再試行します。再試行が成功した場合は、オペレーター・アクションとプログラム・アクションのどちらも必要ありません。以下のいずれかまたはすべての方法で、プログラムの処理に影響する可能性のあるエラーを報告できます。

- ファイルを使用しているプログラムのプログラム・メッセージ待ち行列に通知、状況、診断、またはエスケープ・メッセージを送信する。ジョブに関するメッセージ・ロギング・レベルの設定に応じて、これらのメッセージをジョブ・ログ中に表示することもできます。
- オペレーター・メッセージ待ち行列 (QSYSOPR) またはヒストリー・メッセージ待ち行列 (QHST) に通知、状況、診断、またはエスケープ・メッセージを送信する。
- 高水準言語でファイル状況コードを戻すことができる。
- ディスプレイ・ファイルの入出力フィードバック域にメジャー/マイナー戻りコードが戻される。
- エラーに関する情報をシステム・エラー・ログに保管して、問題分析プログラムや解決プログラムで使うことができる。
- ネットワーク内の別のシステムのオペレーターに警報メッセージを送信できる。
- 通常のプログラムのフローに割り込んでエラー処理サブルーチンに制御を転送するか、または他の言語操作を行うことができる。実行時エラーを処理する方法に関する追加情報については、該当する高水準言語の資料を参照してください。

エラーの回復を試行しようとしているプログラムには、上記のうち一部のみ有効です。

プログラミングされたエラー回復を行えないファイル・エラーもあります。永続 (つまり、修正アクションを行うまでファイル、装置、またはプログラムが作動できない) と見なされるエラーもあります。このアクションには、装置をオフに変更してから再びオンに変更してリセットすることや、装置構成またはアプリケーション・プログラム中のエラーを訂正することが含まれる可能性があります。特定のメッセージと戻りコードを使用して、ユーザーまたはアプリケーション・プログラムに、エラー以外の情報から成る条件を通知します。たとえば、通信回線の状況の変更や、予期しない条件の場合に行われるシステム・アクションがあります。多くの場合、アプリケーション・プログラムがエラー条件に関するテストを行い、ユーザーの介入なしにそのプログラムが続行できるようにする事前に計画された回復アクションを行うことができます。

メッセージおよびメッセージ・モニターの説明

表示されたメッセージは、新しいアプリケーションをテストしている人物に関する基本的な情報源になります。普通メッセージには、ファイル状況コード、標識、またはメジャー/マイナー戻りコードより具体的な情報が含まれます。制御言語を使用すると、メッセージをモニターして、制御言語プログラムがメッセージを傍受して修正アクションを行えるようにすることができます。メッセージ・タイプとメッセージのモニターについて詳しくは、iSeries Information Center の『CL』トピックの『Messages』セクションを参照してください。ほとんどの高水準言語では、ファイル状況コードやメジャー/マイナー戻りコード (後述) の方が情報源として便利です。

プログラムによる関連メッセージのグループのモニターを容易にするために、カテゴリ単位でメッセージ番号が割り当てられています。ファイル・エラー・メッセージの場合は、以下のメッセージ番号の範囲が割り当てられています。

表 24. システム・メッセージ番号の範囲

メッセージ ID	操作	メッセージ・タイプ
CPF4001-40FF	オープン	診断および状況
CPF4101-43FF	オープン	ファイルが使用不能になるエスケープ
CPF4401-44FF	クローズ	診断および状況
CPF4501-46FF	クローズ	ファイルが使用不能になるエスケープ
CPF4701-48FF	入出力、獲得、および解放	通知と、取り消し、状況、およびファイルや装置が使用不能にならないエスケープのデフォルト応答
CPF4901-49FF	入出力、獲得、および解放	通知と、無視のデフォルト応答
CPF5001-50FF	入出力、獲得、および解放	通知と、取り消しのデフォルト応答
CPF5101-53FF	入出力、獲得、および解放	ファイルまたは装置が使用不能になるエスケープ
CPF5501-56FF	入出力、獲得、および解放	ファイルまたは装置が使用不能になるエスケープ

一部の状況メッセージ (CPF4018 など) では、その前に追加情報を提供する診断メッセージが付随します。診断メッセージは、ジョブに関するメッセージ・ロギング・レベルに応じて、ジョブ・ログ中に保持できません。制御言語プログラムが CPF4018、CPF5041、または同種のメッセージをモニターする場合、付随する診断メッセージをプログラム・メッセージ待ち行列から取り出せます。

エスケープ・メッセージが発行され、このメッセージがモニターされないエラーが起こると、ご使用のプログラムは終了し、メッセージがオペレーター・メッセージ待ち行列に表示されます。状況メッセージもモニターできますが、モニターされない場合でも、プログラムは続行します。CL 以外のほとんどの高水準言語は、検出されそうなすべてのファイル・エラーをモニターし、いくつかの標準的な回復を提供します。エラーの重大度に応じて、高水準言語は単にプログラムを終了して独自のメッセージを発行します。代わりに、エラー回復ルーチンをコーディングして、特定のアプリケーション中で予期されるエラーを処理することもできます。

これらのエラー処理ルーチン中では、普通はファイル状況コードかメジャー/マイナー戻りコードを調べて、エラーの原因を判別する必要があります。ご使用の言語に関する資料には、ファイル状況コードやメジャー/マイナー戻りコードにアクセスする方法について説明されています。この言語の資料には、言語ごとに定義されているファイル状況コードについても説明されています。

メジャー/マイナー戻りコードの説明

メジャー/マイナー戻りコードは、ディスプレイ・ファイルに関するエラーと特定の状況情報の報告に使用されます。普通これらのコードは 4 文字で示されます。最初の 2 文字はメジャー・コードを参照し、次の 2 文字はマイナー・コードを参照します。メジャー・コードは一般タイプのエラーを示し、マイナー・コードは詳細情報を提供します。ゼロ以外のマイナー・コードには、組み合わせられているメジャー・コードにかかわらず、同じまたは似た意味があります。

アプリケーション・プログラムは、個々の入出力操作の後で戻りコードをテストできます。メジャー戻りコードが 00 の場合は、操作が正常に完了し、読み取り操作か書き込み操作のどちらを次に実行する必要があるかを示す状況情報がマイナー戻りコードに含まれます。メジャー戻りコードが 04 以上の場合は、エラーが起きたことを示します。プログラムは、プログラミングされた回復が試行される特定のエラーをテストできます。アプリケーション・プログラムは、メジャー・コードとマイナー・コードを 1 つの単位として比較することにより特定の条件をテストしたり、メジャー・コードを単独でテストすることにより条件のクラスを識別したりできます。

ほとんどのメジャー/マイナー戻りコードには、標準的な回復アクションが似ている複数のメッセージ番号のうち 1 つが付随します。ファイル状況コードは、個別の言語によって定義され、メジャー/マイナー戻りコードに基づいて設定できます。

次の表は、メジャー戻りコードを定義しています。『付録 D. ディスプレイ・ファイルの戻りコード』には、ディスプレイ・ファイルに関して使用される場合のメジャーおよびマイナー戻りコードの特定の定義と、その個々に関連したメッセージ番号が記載されています。

表 25. メジャー戻りコードの定義

コード	定義
00	ご使用のプログラムによって要求された操作は正常に完了しました。マイナー・コードには、方向転換などの状態情報が組み込まれます。
02	入力操作が正常に完了しましたが、ジョブは終了します (制御されます)。マイナー・コードに状態情報が組み込まれます。
03	入力操作が成功しましたが、データが受信されませんでした。マイナー・コードに状態情報が組み込まれます。
04	データの読み取りの待機中に出力操作が試行されたので、エラーが発生しました。
08	装置がすでに獲得されているか、またはセッションがすでに確立されているために、獲得操作が失敗しました。
11	装置またはセッションが送信勧誘されなかったので、送信勧誘プログラム装置からの読み取り操作が失敗しました。
34	入力例外が発生しました。データ長またはレコード様式がプログラムにとって受け入れ可能ではありませんでした。
80	回復できない永続的なシステム・エラーまたはファイル・エラーが発生しました。問題を訂正するには、プログラマーのアクションが必要です。
81	入出力操作中に、回復できない永続的な装置エラーまたはセッション・エラーが発生しました。
82	オープンまたは獲得操作中に、装置エラーまたはセッション・エラーが発生しました。回復が可能かもしれませんが。

表 25. メジャー戻りコードの定義 (続き)

コード	定義
83	入出力操作中に、装置エラーまたはセッション・エラーが発生しました。回復が可能かもしれません。

エラーからの回復

以下に、メジャー戻りコードのグループごとに該当するエラー回復アクションについて説明します。

正常完了

メジャー/マイナー戻りコード 0000 は、ご使用のプログラムによって要求された操作が正常に完了したことを示します。ほとんどの場合、メッセージは発行されません。場合によっては、診断メッセージが使用されて、システムが処理できたものの特定の条件下ではエラーと見なされる可能性のある異常な条件がユーザーに通知されることもあります。例えば、無効なパラメーターが無視された可能性があることや、特定のデフォルトのアクションが行われたことなどが含まれます。

例外での完了

アプリケーション・プログラムからの特定の応答が該当する条件に、複数のより具体的なメジャー戻りコードが割り当てられています。

メジャー戻りコード 02 は、要求された入力操作が正常に完了しましたが、ジョブは終了する (制御される) ことを示します。アプリケーション・プログラムは可能な限り迅速に処理を完了する必要があります。通常の方法でプログラム時間を終了できるように、制御付き取り消しが使用されます。ENDJOB コマンドで指定された時間内にご使用のプログラムが終了しない場合は、システムによって追加の通知なしにジョブを終了します。

メジャー戻りコード 03 は、入力操作が成功しましたが、データが転送されなかったことを示します。アプリケーションによっては、エラー条件になる場合もあれば、ユーザーがデータを入力せずにファンクション・キーを押した時点で予想される場合もあります。また、すべてのデータが処理されているので、アプリケーション・プログラムが完了処理に進む必要があることも示す場合もあります。いずれの場合も、プログラム中の入力バッファの内容を無視する必要があります。

メジャー/マイナー・コード 0309 は、データが受け取られず、しかもジョブが終了される (制御される) ことを示すのに使用されます。メジャー/マイナー・コード 0310 は、指定された待機時間が終了しているためにデータがないことを示します。02 または 03 メジャー・コードに付随する他のマイナー戻りコードは、00 メジャー・コードの場合と同じで、通信状況と次に実行される操作を示します。

メジャー戻りコード 04 は、出力例外が起きたことを示します。特に、受信待ち状態のデータがあった場合は、ご使用のプログラムはデータの送信を試行しました。この例外は、以前の正常完了時にマイナー戻りコードが適切に処理されなかった結果生じたと考えられます。ご使用のプログラムは、単に着信データを受け取ってから書き込み操作を繰り返して、回復できます。

メジャー戻りコード 34 は、入力例外が起きたことを示します。受信されたデータが長すぎたか、またはレコード様式と互換性がありませんでした。マイナー戻りコードは、受信されたデータの誤りは何か、およびデータが切り捨てられたのかそれともリジェクトされたのかを示します。ご使用のプログラムが例外を処理して続行できると考えられます。データがリジェクトされた場合は、別のレコード様式を指定してそのデータを読み取れる可能性があります。

このグループに含まれる他の 2 つの戻りコード 0800 および 1100 は、両方とも普通はアプリケーション・プログラミング・エラーの結果ですが、依然として回復可能です。0800 は、装置がすでに獲得されて

いるか、またはセッションがすでに確立されているために、獲得操作が失敗したことを示します。1100 は、送信勧誘された装置がないのに、プログラムが送信勧誘装置からの読み取りを試行したことを示します。どちらの場合も、無効な要求は無視され、プログラムは続行できます。

02 メジャー・コードと共に発行されるメッセージや、03 メジャー・コードに付随するほとんどのマイナー・コードと共に発行されるメッセージはありませんが、このグループ中の他の例外には CPF4701-CPF47FF または CPF5001-CPF50FF の範囲内のメッセージが付随します。

永続的なシステム・エラーおよびファイル・エラー

メジャー戻りコード 80 は、ファイルに影響を与える重要なエラーを示します。ファイルを再び使用しようとする前に、アプリケーション・プログラムはファイルを閉じてから再び開かなければなりません。エラーの原因になっている問題が検出されて訂正されなければ回復の見込みはありません。共用ファイルを閉じてから再び開いてこのファイル中のエラー条件をリセットするには、オープン・データ・パスを共用しているすべてのプログラムがこのファイルを閉じなければなりません。プログラム・スタック中の以前のプログラムに戻って、これらのプログラムごとに共用ファイルを閉じる必要が生じることがあります。付随するメッセージのテキストを参照して、特定のエラーに該当するアクションを判別する必要があります。

このグループ中で、特別な興味を引くマイナー戻りコードが複数あります。メジャー/マイナー・コード 8081 は、APAR が必要になると考えられる重要なシステム・エラーを示します。このメジャー/マイナー戻りコードと共に送信されるメッセージで、問題分析 (ANZPRB) コマンドを実行して詳細な情報を入手するよう指示される場合があります。

メジャー/マイナー・コード 80EB は、非互換オプションまたは無効なオプションが、ディスプレイ・ファイル中に指定されたか、またはオープン操作に関するパラメーターとして指定されたことを示します。ほとんどの場合、ファイルを閉じ、プログラムを終了し、オーバーライド・コマンドを使用して無効なパラメーターを訂正して、再びプログラムを実行できます。オーバーライド・コマンドは、このコマンドが発行されたジョブのみに影響を与えます。このコマンドを使用すると変更内容を容易にテストできますが、変更内容を永続させるには、結局は適切な場合にはディスプレイ・ファイルに変更を加えるか再作成することもできます。

入出力操作時の永続的な装置エラーまたはセッション・エラー

メジャー戻りコード 81 は、装置またはセッションに影響を与える重要なエラーを示します。このエラーには、装置、通信回線、または通信コントローラーに影響を与えるハードウェア障害が含まれます。また、予期しない装置の切断か電源オフによるエラー、および装置によって検出されシステムに報告が戻された異常条件も含まれます。マイナー戻りコードおよび付随するメッセージは両方とも、問題の原因に関するさらに具体的な情報を提供します。

ファイル・タイプに応じて、プログラムはファイルを閉じてから再び開くか、装置を解放してから再び獲得するか、またはセッションを再び獲得しなければなりません。共用ファイルを閉じてから再び開いてこのファイル中のエラー条件をリセットするには、オープン・データ・パスを共用しているすべてのプログラムがこのファイルを閉じなければなりません。装置をオフに変更してから再びオンに変更してリセットするようにメッセージで指示される場合もあります。エラーの原因になっている問題が検出されて訂正されなければ、障害のある装置をプログラムが使用できる見込みはありませんが、代替装置を使用できる場合はプログラム内での回復が有効な可能性があります。

このグループ中のマイナー戻りコードの中には、82 メジャー戻りコードに関するマイナー戻りコードと同じものもあります。装置または回線の障害はいつでも生じる可能性があります。81 メジャー・コードは入出力操作時に生じます。このコードは、ご使用のプログラムが装置またはセッションとのリンクをすでに

確立していることを意味します。したがって、一部のデータは転送済みの可能性があります。再びプログラムが開始される際に、データ転送は先頭から開始されます。データが重複する結果になる可能性があります。

81 メジャー・コードに付随するメッセージ番号は、入出力操作かクローズ操作を示す範囲内である可能性があります。クローズ操作時に装置に障害が起きた場合、ファイルを閉じるのに固有のアクションではなく、単にデータの最終ブロックの送信に失敗した結果として起きた可能性があります。クローズ操作時にエラーが起きた場合、ファイルが一部のみ閉じたままになる結果になる可能性があります。エラー回復プログラムはクローズの失敗に対して、2 度目のクローズ操作で応答する必要があります。2 度目のクローズは、エラーにかかわらず常に完了します。

オープン操作または獲得操作時の装置エラーまたはセッション・エラー

メジャー戻りコード 82 は、オープン操作または獲得操作中に、装置エラーまたはセッション・エラーが発生したことを示します。マイナー戻りコードおよび付随するメッセージは両方とも、問題の原因に関するさらに具体的な情報を提供します。

このグループ中のマイナー戻りコードの中には、81 メジャー戻りコードに関するマイナー戻りコードと同じものもあります。装置または回線の障害はいつでも生じる可能性があります。82 メジャー・コードは、ご使用のプログラムが初めて使用しようとした際に装置またはセッションを使用できなかったことを示します。つまり、転送されたデータはありません。この問題は、構成エラーかインストール・エラーの結果として生じた可能性があります。

マイナー戻りコードに応じて、ご使用のプログラムがエラーから回復するか、または一定の期間待った後に失敗した操作を再試行することが該当する場合があります。試行する回数はプログラム側で指定しなければなりません。代わりに、代替またはバックアップの装置またはセッションを使用することもできます。

82 メジャー・コードに付随するメッセージ番号は、オープン操作か獲得操作を示す範囲内である可能性があります。オープン操作の場合は、エラーから回復するには、部分的に開かれたファイルを閉じてから再び開く必要があります。獲得操作の場合は、解放操作を行った後で獲得を再試行する必要が生じる可能性があります。どちらの場合も、システムがエラーから回復できるのに十分な長さのファイル待ち時間を指定する必要があります。

入出力操作時の回復可能な装置エラーまたはセッション・エラー

メジャー戻りコード 83 は、装置へのデータの送信時か装置からのデータの受信時にエラーが起きたことを示します。アプリケーション・プログラムによる回復が可能です。マイナー戻りコードおよび付随するメッセージは両方とも、問題の原因に関するさらに具体的な情報を提供します。

このグループ中のほとんどのエラーは、無効なコマンドかデータを装置に送信した結果か、または有効なデータを誤った時刻かそのデータを処理できない装置に送信した結果生じます。アプリケーション・プログラムは、失敗した操作かデータ項目をスキップして次に進むか、または該当するデフォルトに置換して、回復できます。アプリケーションに論理エラーが存在する可能性があります。

第 9 章 ディスプレイ・ファイルを使用したメニューの作成およびアクセス

この章では、ユーザー定義のメニューについて説明し、それらの作成方法、およびシステム・メニューと合わせた使用方法について解説します。

ユーザー定義のメニューのうち、以下の上 2 つのタイプは、本章の情報を使用して作成できます。

ディスプレイ・ファイル・メニュー

メニュー・フォーマットを示すデータ記述仕様 (DDS) によって定義された表示を使用するメニュー。メニュー機能は、各メニュー・オプションを実行するために使用されるコマンドを含むメッセージ・ファイルによって制御されます。

プログラム・メニュー

メニュー・フォーマットを示し、メニュー・オプションを実行するために必要な機能を提供するために、高水準言語プログラムを使用するメニュー。

UIM メニュー

ユーザー・インターフェース管理機能 (UIM) パネル・グループ定義言語によって定義されたメニュー・オブジェクトを持つメニュー。UIM を使用したユーザー独自のメニューの定義について詳しくは、308 ページの『UIM を使用したメニュー・オブジェクトの定義』を参照してください。

CL コマンドについて

本章の例のいくつかは CL コマンドを使用します。特定の CL コマンドについて詳しくは、iSeries Information Center の『CL』のトピックを参照してください。

システム・メニューとユーザー定義メニューの実行

*MENU オブジェクト・タイプから成るメニューは、メニュー表示へ (GO) コマンドを使用して実行されます。GO コマンドでは、特定のメニューまたは総称メニュー名のいずれかを指定できます。総称名を指定する場合、「Work with Menus」画面が表示されます。これは、ユーザーが使用可能なすべてのメニューを表示します。このリストから、実行するメニューを選択できます。

GO コマンドを実行後、メニューに戻る

GO コマンドの戻り点 (RTNPNT) パラメーターを使用すると、GO コマンドによって指定されたメニューの実行後、コマンドが入力されたメニューに戻るかどうかを指定できます。

直前のメニューの判別

メニューはその実行前に、内部メニュー・スタックに置かれます。メニューに対してスタックが使用不可の場合には、それが作成されます。メニューに対して取り消しキーが押されると、スタック内の直前のメニューが表示されます。各メニュー・スタックの深さは 10 エレメント (メニュー) です。11 番目のメニューがメニュー・スタックに置かれると、最初の (一番古い) メニューが、スタックから除去されます。このメニューには、取り消しキーを使用して戻ることはできません。

メニューでの取り消しキーおよび終了キーの使用

以下の例では、メニュー・オプションと GO コマンドの両方を使用して、一連のメニューが表示されます。

1. i5/OS Main Menu で、コマンド行に GO PROGRAM RTNPNT(*YES) と入力します。ここでの RTNPNT(*YES) という指定は、戻り点として Main Menu が使用されることを意味します。

MAIN	I5/OS メインメニュー	システム : XXXXXXXX
------	---------------	-----------------

次の1つを選択してください。

1. ユーザー・タスク
2. オフィス・タスク
3. 汎用システム・タスク
4. ファイル、ライブラリー、およびフォルダー
5. プログラミング
6. 通信
7. システムの定義または変更
8. 問題処理
9. メニューの表示
10. 情報援助オプション
11. ISERIES ACCESS のタスク

90. サインオフ

選択項目またはコマンド
====> GO PROGRAM RTNPNT(*YES) _____

F3= 終了 F4=プロンプト F9=コマンドの複写 F12= 取り消し F13= 情報援助
F23= 初期メニューの設定

2. 次に、Programming メニューのコマンド行で GO PROBLEM RTNPNT(*NO) と入力し Enter キーを押すと、Problem Handling メニューが表示されます。Programming メニューは戻り点として設定されません。

PROGRAM	プログラミング	システム : XXXXXXXX
---------	---------	-----------------

次の中から1つを選んでください。

1. プログラマー・メニュー
2. プログラミング開発管理機能 (PDM)
3. ユーティリティ
4. プログラミング言語のデバッグ
5. SQL 事前コンパイラー
6. 質問・回答 (Q&A)

8. 画面イメージのコピー

9. システム共通プロダクト/適用業務の実行 (CSP/AE)

50. システム/3 6 プログラミング

70. 関連コマンド

選択項目またはコマンド
====> GO PROBLEM RTNPNT(*NO) _____

F3= 終了 F4= プロンプト F9= コマンドの複写 F12= 取り消し
F13= 情報援助 F16= システム・メインメニュー

PROBLEM	問題処理	システム : XXXXXXXX
次の中から1つを選んでください。		
1. 質問・回答 (Q&A) 2. 問題の処理 3. ネットワーク問題処理 4. システム操作員メッセージの表示 5. 活動記録ログの表示 6. システム・サービス・ツール 60. 問題処理オプションの続き 70. 関連コマンド		
選択項目またはコマンド ==> _____		
F3= 終了	F4= プロンプト	F9= コマンドの複写
F13= 情報援助	F16= システム・メインメニュー	F12= 取り消し

3. この例では、Problem Handling メニューの表示中に取り消しキーを押すと、ユーザーは Programming メニューに戻ります。Programming メニューはメニュー・スタックで先行する表示だからです。Problem Handling メニューの表示中に F3 (Exit) を押すと、ユーザーは Main Menu に戻ります。Main Menu は、GO コマンドが RTNPNT(*YES) を指定して入力された最新の表示だからです。

RTNPNT パラメーターのデフォルト値は *YES なので、GO PROBLEM は GO PROBLEM RTNPNT(*YES) と同じ効果を持つことに注意してください。

メニューのヘルプを表示中に F3 (Exit) または F12 (Cancel) のいずれかを押すと、ユーザーはそのメニュー自体に戻ります。

サインオン時に表示されるメニューの選択

あるメニューを初期メニュー、つまりサインオン時に表示される最初のメニューとして指定するには、メニューが表示されたときに F23 を押します。F23 は、ご使用のユーザー・プロファイル内に初期メニュー属性を設定します。ご使用の初期プログラムが他のメニューを要求するのではない限り、F23 を押すことによって選択されたメニューが、次のサインオン時に初期メニューとして表示されます。

注: このキーを使用できるのは、LMTCPB(*NO) がご使用のユーザー・プロファイル内で指定される場合だけです。LMTCPB(*NO) は、初期メニューの変更を許可します。

初期メニューおよび限定ユーザーについては、*iSeries* 機密保護解説書を参照してください。

ユーザー独自のディスプレイ・ファイル・メニューの定義

ディスプレイ・ファイル・メニューは、以下の 3 つの部分から構成されます。

- メニューがディスプレイ装置で表示される仕方を定義する、DDS 定義のディスプレイ・ファイル。
- メニュー・オプションのいずれかが選択されたときに取られる動作を定義するメッセージ・ファイル。
- メニューを含むメニュー・オブジェクト (*MENU)

ヘルプは、以下のものについて、ディスプレイ・ファイル・メニュー上で使用可能です。

- コマンド** コマンド・ヘルプは、コマンド行にコマンドを入力した後、ヘルプ・キーを押すことによって、表示されます。
- 機能キー** 機能キーのヘルプは、機能域にカーソルを移動させ、Help キーを押すことによって、表示されます。
- メッセージ** メッセージのヘルプは、メッセージにカーソルを移動させ、ヘルプ・キーを押すことによって、表示されます。

メニュー・オプションのヘルプも DDS を使用して提供できます。ヘルプ情報についてさらに詳しくは、265 ページの『メニュー用のヘルプ・フォーマットの命名』を参照してください。

DDS キーワードについて

本章で使用される特定の DDS キーワードについて詳しくは、iSeries Information Center の『DDS』のトピックに記載されています。

メニューに関する DDS およびディスプレイ・ファイル考慮事項

メニューの定義で使用される DDS については、いくつかの制約事項があります。

表 26. ディスプレイ・ファイル・メニューの制約事項

影響される領域	制約事項
画面サイズ	ディスプレイ・ファイル内の、メニュー・フォーマットおよびそのメニューに関連したヘルプ・レコード様式は、DDS 内で 24 行 x 80 列の画面サイズ、DPSIZ(*DS3) を指定しなければなりません。
ディスプレイ・ファイル名とレコード様式名	メニュー・レコード様式名は、ディスプレイ・ファイル名と同じでなければなりません。
ヘルプ・フォーマット名	ヘルプ・フォーマット名は、#Hxyy という規則に従わなければなりません。この規則は、265 ページの『メニュー用のヘルプ・フォーマットの命名』で説明されています。ヘルプ・レコード様式は、150 項目に限定されます。 注: Hxyy の前の記号は番号記号です。
標識域	別個の標識域 (INDARA キーワード) が、表示用の DDS で宣言されなければなりません。
サブファイルの使用	サブファイルは使用できません。
許可される行	画面の最初の 21 行だけに記述すべきです。記述は、ヘルプ画面の最初の 21 行だけにすべきです。行 22 と 23 は、長いコマンド行で使用され、行 24 にはシステム・メッセージが表示されます。(行 23 は短いコマンド行で使用され、行 24 はメッセージに使用されます。) 機能キーの凡例が要求されるときには、以下の行を使用できます。 <ul style="list-style-type: none"> 機能キーの凡例がない、長いコマンド行の場合は、行 1 から 21 機能キーの凡例のある、長いコマンド行の場合は、行 1 から 19 機能キーの凡例のない、短いコマンド行の場合は、行 1 から 22 機能キーの凡例のある、短いコマンド行の場合は、行 1 から 20 機能キーの凡例がなく、コマンド行がない場合は、行 1 から 21 機能キーの凡例があり、コマンド行がない場合は、行 1 から 19
キーボード・ロック	LOCK レコード・レベル・キーワードを使用して、画面が表示される前にキーボードがアンロックされないようにしなければなりません。

表 26. ディスプレイ・ファイル・メニューの制約事項 (続き)

影響される領域	制約事項
ページ送り	ロール可能 (ALWROL) レコード・レベル・キーワードは指定できません。
書き込み操作の据え置き	DFRWRT(*NO) は、ディスプレイ・ファイル・メニューの作成時に CRTDSPF コマンドで 必須 です。 DFRWRT(*NO) は、メニューの実行時にメニュー・フォーマットが表示されることを確実にします。

以下の提案も使用して、DDS を使用したメニューの定義に役立てることができます。

表 27. ディスプレイ・ファイル・メニューのための提案

影響される領域	オプション
画面の復元	RSTDSP(*NO) は CRTDSPF コマンドではオプションですが、指定される場合、システム・パフォーマンスをかなり向上させることがあります。
メニューおよびディスプレイ装置別名	メニュー名およびディスプレイ装置名に別名を使用できます (System/36™ 環境の場合)。 CRTMNU コマンドの発行時に、DDS で使用したものと異なる名前を指定できます。このようにして、ディスプレイ装置 ID の名前がメニューで表示されます。メニューに異なる名前を指定することにより、GO コマンドを使用してそのメニューを検出できます。

メッセージ・ファイル内のメニュー・アクションの記述

メッセージ・ファイルは、メニュー・オプションの選択時に取られる動作を記述するために使用します。取られる動作を制御するコマンドは、通常、メッセージ内に置かれます。いくつかのケースでは、コマンド・ストリングが長すぎて、メッセージ内に収まらないことがあります。そのような場合には、メッセージにそのメッセージのメッセージ ID が含まれます。その場合、コマンドはメッセージ・ヘルプ内に置かれます。

ディスプレイ・ファイル・メニュー内で使用されるメッセージは、メッセージ接頭語 **USR** を使用して命名されなければなりません。メッセージ ID の残りの 4 桁は、メニュー・オプション番号に対応します。

たとえば、指定されたディスプレイ・ファイル・メニュー上のオプション 3 が以下の場合、

3. Personnel Menu

コマンド行で 3 と入力されるときに取られる動作を記述するメッセージは、ID **USR0003** を使用します。メッセージには、'GO PERSMENU RTNPNT(*NO)' のような任意のコマンドを含めることができます。

メッセージについての詳細は、247 ページの『独自のメッセージの作成および表示』を参照してください。

メニュー用のヘルプ・フォーマットの命名

ディスプレイ・ファイル・メニューのヘルプ・フォーマットは、以下の形式を使用して命名されます。

#Hxxyy

ここで、

Hxxyy の前の記号は番号記号です。

xx は、ヘルプ情報が適用される最初のメニュー・オプションの番号です。

yy は、ヘルプ情報が適用される最後のメニュー・オプションの番号です。

たとえば、#H0103 を使用して、メニュー・オプション 1 から 3 に適用されるヘルプを命名できます。
#H0202 は、メニュー・オプション 2 に適用されるヘルプを命名します。

#H0000 は、メニューの全般ヘルプを命名します。

ヘルプ・フォーマットは、DDS 内で任意の順序で記述できます。システムはメニューの実行時に、ヘルプ・フォーマットを昇順でソートします。

複数のヘルプ・フォーマット名が同じオプションに適用される場合、最初のヘルプ・フォーマット (システムによってソートされた順序で最初のもの) だけが、そのオプションのヘルプの要求時に表示されます。たとえば、以下のヘルプ・フォーマットが指定された場合、

```
#H0000  
#H0101  
#H0205  
#H0306  
#H0707
```

コマンド行で 3 と入力して (オプション 3 を選択) ヘルプ・キーを押すと、#H0205 を指定されたヘルプが表示されます。この例では、#H0306 は Page Down キーを使用して表示できますが、オプション 6 のヘルプを要求することによってのみ、直接に表示されます。

同様に、カーソルをコマンド行に移動させ、コマンド行にオプションを入力せずにヘルプ・キーを押すことによって、メニューの全般ヘルプ (#H0000) が表示されます。カーソル移動キーを使用して、メニュー・オプションの他のヘルプ・フォーマットを参照できます。

以下の例は、5 つのオプションを持つメニューを表示します。ヘルプ・フォーマット名、およびそれらが適用されるメニュー・オプションの名前が、例の後に示されています。

```
PERSMENU                Personnel Menu
```

```
Select one of the following:
```

1. Departments menu
2. Education menu
3. Benefits menu
4. Job openings
5. Job applicants

```
Selection or Command
```

```
====>
```

ヘルプ名	内容
#H0000	Personnel Menu (PERSMENU) の一般ヘルプ
#H0101	オプション 1、Departments menu のヘルプ
#H0202	オプション 2、Education menu のヘルプ
#H0303	オプション 3、Benefits menu のヘルプ

#H0405 オプション 4 および 5、Job openings および Job applicants のヘルプ

メニュー・オプションのヘルプ情報は、コマンド行にオプション番号を入力してヘルプ・キーを押すことにより、または Help (メニューの全般ヘルプ) を押した後 Page Down キーを使用し、そのオプションのヘルプのページまで進むことによって、表示できます。

ディスプレイ・ファイル・メニューの作成

以下のステップにより、ディスプレイ・ファイル (*DSPF) メニューを作成できます。

1. メニューおよびメニュー・ヘルプ情報を記述する
2. ディスプレイ・ファイルを作成する
3. メッセージ・ファイルを作成する
4. メッセージをメッセージ・ファイルへ追加する
5. メニューを作成する
6. メニューを実行する

以下のセクションでは、それぞれのステップを詳しく説明します。

メニューおよびメニュー・ヘルプ情報を記述する

DDS は、メニューの外観およびヘルプの両方を記述するために使用します。

以下の DDS は、PERSMENU というサンプル表示用のものです。最初のレコード様式 (PERSMENU) を使用して、メニューが表示される時の外観を記述します。行 5 から 9 にわたって、位置 7 から始まる 5 つのメニュー・オプションがリストされます。

さらに 5 つのレコード様式、#H0000、#H0101、#H0202、#H0303、および #H0405 が表示されます。これらの各々を使用して、メニューおよびそのオプションのヘルプ情報を提供します。前述のシステム規則に従い (265 ページの『メニュー用のヘルプ・フォーマットの命名』を参照)、#H0000 を使用してメニューの全般ヘルプを提供します。他のレコード様式は、メニュー・オプションのヘルプを提供します。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A* PERSONNEL MENU (PERSMENU) SPECIFICATION
A*
A          PRINT DSPTSIZ(*DS3)
A          INDARA
A*
A          R PERSMENU          LOCK
A          1 02'PERSMENU'
A          1 29'PERSONNEL MENU'
A          3 02'SELECT ONE OF THE FOLLOWING:'
A          5 07'1. DEPARTMENTS MENU'
A          6 07'2. EDUCATION MENU'
A          7 07'3. BENEFITS MENU'
A          8 07'4. JOB OPENINGS'
A          9 07'5. JOB APPLICANTS'
A          21 02'SELECTION OR COMMAND'
A*
A          R #H0000          LOCK
A          1 02'GENERAL'
A          2 26'HELP FOR THE PERSONNEL MENU
A          3 02'THIS IS THE GENERAL HELP'
A
A          3 27'FOR THE PERSONNEL MENU.'
A*
A          R #H0101          LOCK
A          1 02'HELP1'
A          1 26'HELP FOR OPTION 1'
A          3 02'THIS IS THE HELP'
A          3 19'FOR OPTION 1.'
A*
A          R #H0202          LOCK
A          1 02'HELP2'
A          1 26'HELP FOR OPTION 2'
A          3 02'THIS IS THE HELP'
A          3 19'FOR OPTION 2.'
A*
A          R #H0303          LOCK
A          1 02'HELP3'
A          1 26'HELP FOR OPTION 3'
A          3 02'THIS IS THE HELP'
A          3 19'FOR OPTION 3.'
A*
A          R #H0405          LOCK
A          1 02'HELP45'
A          1 26'HELP FOR OPTIONS 4 AND 5'
A          3 02'THIS IS THE HELP'
A          3 19'FOR OPTIONS 4 AND 5.'
A*

```

図 100. PERSMENU というサンプル・メニュー用の DDS ソース

ディスプレイ・ファイルを作成する

メニュー用のディスプレイ・ファイルを作成するには、以下のコマンドを入力します。

```

CRTDSPF FILE(PERLIB/PERSMENU) +
        SRCFILE(USERLIB/SFPERS) +
        DFRWRT(*NO) +
        RSTDSP(*NO)

```

ライブラリー PERLIB 内のソース・ファイル SFPERS 内にある DDS ソース・メンバー PERSMENU を使用して、PERSMENU メニュー用のディスプレイ・ファイルを作成します。

メッセージ・ファイルを作成する

以下のメッセージ・ファイル作成 (CRTMSGF) コマンドを入力して、PERSMENU という名 (レコード様式と同じ名前) のメッセージ・ファイルを作成します。

```
CRTMSGF MSGF(PERLIB/PERSMENU) +
      TEXT('Message file of commands for menu PERSMENU.')
```

メッセージ・ファイルは、各種メニュー・オプションを選択したときに取られる動作を記述するメッセージを含めるために使用されます。

メッセージをメッセージ・ファイルへ追加する

メッセージ記述追加 (ADDMSGD) コマンドを使用して、メッセージをメッセージ・ファイルに追加します。メッセージは、各種メニュー・オプションが選択されたときに取られるさまざまな動作を記述します。

それぞれの ADDMSGD コマンドの MSGID パラメーターは、USRxxxx という形式です。xxxx はメニュー・オプション番号です。MSG パラメーターには、メニュー・オプションを実行するコマンドが含まれます。

以下の例では、最初の 3 つのメッセージの MSG パラメーターには、メニューを実行するコマンド (GO コマンドを使用) を含め、後の 2 つのメッセージには、それらのメニュー・オプションのいずれかがメニューから選択されたときにプログラムを呼び出すコマンド (CALL コマンドを使用) を含めます。

```
ADDMSGD MSGID(USR0001) MSGF(PERLIB/PERSMENU) +
      MSG('GO DEPTMENU RTNPNT(*NO)')
ADDMSGD MSGID(USR0002) MSGF(PERLIB/PERSMENU) +
      MSG('GO EDUCMENU RTNPNT(*NO)')
ADDMSGD MSGID(USR0003) MSGF(PERLIB/PERSMENU) +
      MSG('GO BENEMENU RTNPNT(*NO)')
ADDMSGD MSGID(USR0004) MSGF(PERLIB/PERSMENU) +
      MSG('CALL JOBOPEN')
ADDMSGD MSGID(USR0005) MSGF(PERLIB/PERSMENU) +
      MSG('CALL JOBAPPS')
```

メニュー・オブジェクトを作成する

以下のメニュー作成 (CRTMNU) コマンドを入力して、メニュー・オブジェクトを作成します。

```
CRTMNU MENU(PERLIB/PERSMENU) TYPE(*DSPF) DSPF(*MENU) +
      MSGF(*MENU) DSPKEY(*NO) CMDLIN(*LONG) +
      TEXT('Personnel menu')
```

DSPKEY パラメーターは、メニューが表示される時、メニュー下部に機能キーの凡例を表示するかどうかを指定します。*NO パラメーターは、メニュー下部に機能キーの凡例を表示しないことを指定します。*YES パラメーターは、メニュー下部に機能キーの凡例を表示することを指定します。特定の機能キーだけを表示するオプションはありません。

CMDLIN パラメーターは、表示されるコマンド行の長さを指定します。*LONG は、153 バイト長のコマンド行を指定します。*SHORT は、73 バイト長のコマンド行を指定します。*NONE は、コマンド行の代わりに 4 バイトのオプション行を指定します。

注: DSPF と MSGF の両方のパラメーターに対して、値 *MENU が指定されたことに注意してください。

*MENU は、ディスプレイ・ファイルまたはメッセージ・ファイルの名前が、メニューの名前と同じであることを指定します。ディスプレイ・ファイル名とメッセージ・ファイル名は、メニューと同じ名前を持つ必要はありません。

メニューを実行する

これで、新規ディスプレイ・ファイル・メニューを、GO コマンドを使用して実行できるようになりました。

GO PERSMENU

```
PERSMENU                Personnel Menu

Select one of the following:

    1. Departments menu
    2. Education menu
    3. Benefits menu
    4. Job openings
    5. Job applicants

Selection or Command
====> _____
_____
```

ユーザー独自のプログラム・メニューの定義

プログラム・メニューは、プログラムを呼び出す GO コマンドを使用することにより、ディスプレイ・ファイル・メニューおよびシステム・メニューに代替を提供します。ユーザーはプログラムを介して、プログラム・メニューの表示および機能を完全に制御できます。

プログラム・メニューのパラメーターの引き渡し

メニューの実行時に、以下の 3 つのパラメーターがプログラムに渡されます。

- メニュー・オブジェクトの 10 文字の名前
- メニュー・オブジェクトを含むライブラリーの 10 文字の名前
- 2 文字のバイナリー戻りコード

3 つのパラメーターはすべて、プログラム内で変数として宣言しなければなりません。戻りコードは、IBM 提供のメニューの機能キーをシミュレートしており、プログラム・メニューを終了する仕方を決定します。

4 つの可能な値は、以下のとおりです。

バイナリー戻りコード	16 進数等価	説明
0	0000	プログラムの再呼び出し (メニューの表示)
-1	FFFF	出口機能の要求
-2	FFFE	直前の機能を要求
-4	FFFC	ホーム機能を要求 (初期メニューの表示)

注: 上記の 16 進値表示は、2 進数をサポートしない高水準言語 (CL など) に使用されます。制御言語プログラムは 2 バイトの文字変数を使用しなければなりません。

プログラム・メニューの作成

この例は、プログラム (*PGM) メニューの作成方法を示します。以下のステップを示し、詳しく説明します。

1. メニューを記述する
2. ディスプレイ・ファイルを作成する
3. ソースを入力し、メニュー機能を制御する制御言語プログラムを作成する
4. メニューを作成する
5. メニューを実行する

メニューを記述する

プログラム・メニューは、メニューの外観を記述するデータ記述仕様 (DDS) を使用して設計されます。以下の DDS は、PGMMENU というサンプル表示用です。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
A          DSPSIZ(*DS3)
A          PRINT
A          CA03(01)
A          CA12(02)
A          HOME(03)
A          INDARA
A          R MENUFMT          BLINK OVERLAY
A          DSPMNUN          10A 0 1 2
A          1 72 TIME
A          1 31'EXAMPLE PROGRAM MENU'
A          DSPATR(HI)
A          2 2'SELECT ONE OF THE FOLLOWING:'
A          3 5'1.' DSPATR(HI)
A          3 10'DISPLAY LIBRARY LIST (DSPLIBL)'
A          4 5'2.' DSPATR(HI)
A          4 10'WORK WITH ACTIVE JOBS (WRKACTJOB)'
A          5 5'3.' DSPATR(HI)
A          5 10'WORK WITH YOUR JOB (WRKJOB)'
A          6 5'90.' DSPATR(HI)
A          6 10'RETURN'
A          23 2'OPTION:'
A          OPTION          3 I 23 12DSPATR(PC)
```

図 101. プログラム・メニュー例用の DDS ソース

ディスプレイ・ファイルを作成する

ディスプレイ・ファイルは、ディスプレイ・ファイル作成 (CRTDSPF) コマンドを使用して作成します。

```
CRTDSPF FILE(QGPL/PGMDDS) SRCFILE(USERLIB/SOURCE1)
```

SOURCE1 という DDS ソース・ファイルを使用して、PGMMENU メニュー用のディスプレイ・ファイルを作成します。

ソースを入力し、制御言語プログラムを作成する

以下の制御言語プログラムを使用して、メニュー機能を制御します。(CL ソースの入力については詳しくは、iSeries Information Center の『CL』のトピックの CL プログラミングのセクションを参照してください。) システムによってサポートされる任意の高水準言語を使用して、メニューを制御することに注意してください。

```

PGM      PARM(&MENUNAME &MENULIB &ACTION) /* Begin the program and +
        indicate that 3 parameters will be passed to it when +
        called. The parameters include 1) The menu name, 2) +
        The menu library name 3) The action desired by this +
        program on return. */
DCL      VAR(&MENUNAME) TYPE(*CHAR) LEN(10) /* Menu name */
DCL      VAR(&MENULIB) TYPE(*CHAR) LEN(10) /* Menu library name */
DCL      VAR(&ACTION) TYPE(*CHAR) LEN(2) /* Action variable */
DCLF     FILE(QGPL/PGMDDS) RCDFMT(MENUFMT) /* Display file */
CHGVAR   VAR(&DSPMNUN) VALUE(&MENUNAME) /* Set the menu name on the +
        menu */
SNDRCVF  DEV(*FILE) RCDFMT(MENUFMT) /* Display the menu at the +
        display station */
CHGVAR   VAR(&ACTION) VALUE(X'0000') /* Indicate the menu should +
        be displayed again */
/*****
/* Handle function keys */
/*****
IF      COND(&IN01 *EQ '1') THEN(CHGVAR VAR(&ACTION) +
        VALUE(X'FFFF')) /* If F3 was pressed, set action +
        to EXIT */
IF      COND(&IN02 *EQ '1') THEN(CHGVAR VAR(&ACTION) +
        VALUE(X'FFFE')) /* If F12 was pressed, set action +
        to PREVIOUS */
IF      COND(&IN03 *EQ '1') THEN(CHGVAR VAR(&ACTION) +
        VALUE(X'FFFC')) /* If HOME was pressed, set action +
        to HOME */
/*****
/* Handle menu options */
/*****
IF      COND(&OPTION *EQ '1') THEN(DSPLIBL) /* If the menu user +
        has selected option 1, then display the library list */
IF      COND(&OPTION *EQ '2') THEN(WRKACTJOB) /* If the menu user +
        has selected option 2, then display the active jobs */
IF      COND(&OPTION *EQ '3') THEN(WRKJOB) /* If the menu user +
        has selected option 3, then display the current job. */
IF      COND(&OPTION *EQ '90') THEN(CHGVAR VAR(&ACTION) +
        VALUE(X'FFFE')) /* If the menu user has selected option +
        90, then set the action to previous. */
ENDPGM   /* End the program */

```

CL プログラム作成 (CRTCLPGM) コマンドを使用して、制御言語プログラムを作成できます。

```
CRTCLPGM PGM(QGPL/PGMCL) SRCFILE(USERLIB/SOURCE1)
```

メニューを作成する

メニュー作成 (CRTMNU) コマンドを使用して、プログラム・メニューを作成します。

```
CRTMNU MENU(QGPL/PGMMENU) TYPE(*PGM) PGM(QGPL/PGMCL)
```

メニューを実行する

これで、新規ディスプレイ・ファイル・メニューを、GO コマンドを使用して実行できるようになりました。

```
GO PGMMENU
```



```

PGMMENU                                EXAMPLE PROGRAM MENU                                09:02:51
SELECT ONE OF THE FOLLOWING:
  1.  DISPLAY LIBRARY LIST  (DSPLIBL)
  2.  WORK WITH ACTIVE JOBS (WRKACTJOB)
  3.  WORK WITH YOUR JOB   (WRKJOB)
  90. RETURN

```

```
OPTION:  __
```

前のメニューに戻らずに、プログラム・メニューから出る

TYPE(*PGM) メニューをコード化して 1 つのプログラム・メニューから別のメニューへ進む場合、RTNPNT(*NO) パラメーターを使用するときには、元のメニューを出るタイミングをメニュー・ドライバーに知らせる必要があります。

プログラム・メニューは、F3 が押されたことを、メニュー・ドライバーに通信する必要があります。プログラム・メニューが、RTNPNT(*NO) を指定してプログラム MENU2 に進むようコード化される場合には、プログラム間で情報を伝達し、またメニュー・ドライバーに情報を伝達して、行うべき処理を知らせることができなければなりません。プログラム間での通信には、データ・キューを使用できます。データ・キューについて詳しくは、iSeries Information Center の『CL』のトピックの CL プログラミングのセクションを参照してください。以下の例は、RTNPNT(*NO) パラメーターを使用して 1 つのメニューから別のメニューへ進む場合に、プログラム・メニュー間での通信方法を示します。

プログラム 1

```

PGM
DCL VAR(&FLDLLEN) TYPE(*DEC) LEN(5 0) VALUE (10)
DCL VAR(&FIELD) TYPE(*CHAR) LEN(10)
DCL VAR(&WAIT) TYPE(*DEC) LEN(5 0) VALUE(0) /* Don't wait */
.
/* Handle the function keys on the menu */
IF COND(&IN03 *EQ '1') + /* F3=Exit pressed */
  THEN(DO)
  /* Delete the Data Queue */
  DLTDTAQ DTAQ(QGPL/DTAQ1)
  /* Create the Data Queue */
  CRTDTAQ DTAQ(QGPL/DTAQ1) MAXLEN(10) +
  TEXT('Test Data Queue')
  GOTO SNDDQ /* Send info to data queue*/
  ENDDO
.
/* Handle the options on the menu */
IF COND(&OPTION *EQ 1) +
  THEN(GO MENU(*LIBL/MENU2) RTNPNT(*NO))
.
/* Returned from menu 2 after F3 or F12 pressed */

```

```

CALL PGM(QRCVDTAQ) PARM(QGPL DTAQ1 &FLDLLEN &FIELD &WAIT)
IF COND((&FLDLLEN *NE 0) *AND (&FIELD *EQ 'EXIT '))
    THEN(GOTO SNDDQ)
ELSE GOTO END
.
.
SNDDQ:DO
    /* Change the variable so the menu driver will know to */
    /* exit and send information to the data queue to */
    /* communicate to next program what happened */
    CHGVAR VAR(&ACTION) VALUE(X'FFFF')
    CHGVAR VAR(&FLDLLEN) VALUE(4)
    CHGVAR VAR(&FIELD) VALUE('EXIT ')
    CALL PGM(QSNDDTAQ) PARM(QGPL DTAQ1 *FLDLLEN &FIELD)
    GOTO END
ENDDO
END: ENDPGM

```

プログラム 2

```

PGM
DCL VAR(&FLDLLEN) TYPE(*DEC) LEN(5 0) VALUE (10)
DCL VAR(&FIELD) TYPE(*CHAR) LEN(10)
DCL VAR(&WAIT) TYPE(*DEC) LEN(5 0) VALUE(0) /* Don't wait */
.
.
/* Handle the function keys on the menu */
IF COND(&IN03 *EQ '1') + /* F3=Exit pressed */
    THEN(DO)
        /* Delete the Data Queue */
        DLTDTAQ DTAQ(QGPL/DTAQ1)
        /* Create the Data Queue */
        CRTDTAQ DTAQ(QGPL/DTAQ1) MAXLEN(10) +
            TEXT('Test Data Queue')
        GOTO SNDDQ /* Send info to data queue*/
    ENDDO
.
.
/* Handle the options on the menu */
IF COND(&OPTION *EQ 1) +
    THEN(GO MENU(*LIBL/MENU2) RTNPNT(*NO))
.
.
/* Returned from menu 2 after F3 or F12 pressed */
CALL PGM(QRCVDTAQ) PARM(QGPL DTAQ1 &FLDLLEN &FIELD &WAIT)
IF COND((&FLDLLEN *NE 0) *AND (&FIELD *EQ 'EXIT '))
    THEN(GOTO SNDDQ)
ELSE GOTO END
.
.
SNDDQ:DO
    /* Change the variable so the menu driver will know to */
    /* exit and send information to the data queue to */
    /* communicate to next program what happened */
    CHGVAR VAR(&ACTION) VALUE(X'FFFF')
    CHGVAR VAR(&FLDLLEN) VALUE(4)
    CHGVAR VAR(&FIELD) VALUE('EXIT ')
    CALL PGM(QSNDDTAQ) PARM(QGPL DTAQ1 *FLDLLEN &FIELD)
    GOTO END
ENDDO
END: ENDPGM

```

メニュー名の競合の回避

高水準システム・メニューはフルワードで命名されるので、ユーザー定義のメニュー名と競合する可能性がいくらかあります。たとえば、ユーザーが CRTMNU コマンドを使用して MAIN という名のメニューを作成する場合には、システム供給のメニュー MAIN と競合する可能性があります。GO コマンドを使用してそのメニューを呼び出すと (GO MAIN)、おそらくシステム・メニュー MAIN が表示されます。なぜなら、それが QSYS ライブラリー内にあるからです。将来のリリースに、前のリリースでユーザーが作成したメニューと同じ名前の新規システム・メニューが含まれる場合にも、競合が生じる可能性があります。

メニューの命名

システム・コマンド・メニューとの命名の競合を回避するためには、メニュー名が CMD で始まらないようにすべきです。システム・コマンド・メニュー名は、形式 CMDxxxxxxx を使用します (xxxxxxx は、CL コマンド名で使用されるサブジェクトまたは動詞)。

ライブラリー・リスト内の高位ライブラリーへのメニューの配置

他のシステム・メニューからのシステム・メニューへの分岐はすべて、ライブラリー・リスト検索順序を使用します。ライブラリー・リスト検索順序の使用により、システム・メニューのマルチリンガル・サポート機能が許可されます。また、ユーザー独自のバージョンを、検索順序でライブラリー QSYS より高位のライブラリーに置くことにより、システム・メニューをオーバーライドできます。

メニューを含むライブラリーの指定

実行したいメニューを含むライブラリーを指定することにより、メニュー名の競合を回避できます。たとえば、GO MENU(*USRLIBL/menu-name) を使用して、ユーザー定義のメニューを呼び出し、また GO MENU(*LIBL/menu-name) を使用して、システム・メニューを呼び出すことができます。

一般メニュー仕様の使用

対話式に作業しているとき、GO コマンドで一般メニュー仕様を使用して、メニュー名の競合を回避できます。前述の例 (『メニュー名の競合の回避』を参照) では、MAIN という名のシステム・メニューとユーザー定義のメニューとの間に、潜在的な競合が存在しました。GO MAIN* を指定すると、システムはユーザーのライブラリー・リスト内にある、MAIN で始まるすべてのメニュー名のリストを表示し、ユーザーが希望のメニューを選択できるようになります。

コマンド複製後のコマンド・デフォルトの変更

複製オブジェクト作成 (CRTDUPOBJ) コマンドを使用して、ユーザー定義のメニューを呼び出すための GO コマンドの複製を作成できます。以下の例は、GOUSR という GO コマンドの複製を作成します。
CRTDUPOBJ OBJ(GO) FROMLIB(QSYS) OBJTYPE(*CMD) NEWOBJ(GOUSR) TOLIB(QGPL)

次に、コマンド・デフォルト変更 (CHGCMDDFT) コマンドを使用して、*LIBL ではなくライブラリー *USRLIBL を使用するよう、MENU パラメーターのコマンド・デフォルトを変更します。

```
CHGCMDDFT CMD(QGPL/GOUSR) NEWDFT('MENU(*USRLIBL/*N)')
```

MENU パラメーターのデフォルトを *USRLIBL に変更すると、GOUSR コマンドはシステム・メニューを迂回して、ライブラリー・リスト内のユーザー部分にあるメニューだけを検索します。

メニュー属性の表示

メニュー属性表示 (DSPMNUA) コマンドを使用して、メニュー・オブジェクトの属性を表示できます。これには以下のものが含まれています。

- メニュー・タイプ (*UIM、ディスプレイ・ファイル、またはプログラム・メニュー)
- ディスプレイ・ファイル (*DSPF) メニューによって使用されるディスプレイ・ファイルおよびメッセージ・ファイル
- プログラム (*PGM) メニューによって使用されるプログラム
- メニューに関連した現行 (CURLIB) ライブラリーおよびプロダクト (PRDLIB) ライブラリー
- メニュー・オブジェクトを記述するテキスト

メニュー属性の変更

メニュー変更 (CHGMNU) コマンドを使用して、オブジェクトを再作成する必要なく、メニュー・オブジェクトの属性を変更できます。変更できる属性は、以下のとおりです。

- すべてのメニュー・タイプの現行 (CURLIB) ライブラリーおよびプロダクト (PRDLIB) ライブラリー
- すべてのメニュー・タイプの記述テキスト
- ディスプレイ・ファイル (*DSPF) メニューによって使用されるディスプレイ・ファイルおよびメッセージ・ファイル
- プログラム (*PGM) メニューによって使用されるプログラム

メニューの削除

メニュー削除 (DLTMNU) コマンドを使用して、ライブラリーからメニュー・オブジェクトを削除できます。DLTMNU コマンドを使用して、メニュー・オブジェクトだけを削除できます。参照されるディスプレイ・ファイルおよびメッセージ・ファイル (ディスプレイ・ファイル・メニューによって使用される) およびプログラム (プログラム・メニューによって使用される) も削除できます。

DLTMNU は一般 コマンドです。たとえば、OELIB という名のライブラリー内のすべてのメニューは、コマンド `DLTMNU OELIB/*ALL` を使用して削除できます。

別の例として、OELIB 内の、名前が ACC で始まるメニューだけを削除する場合は、コマンド `DLTMNU OELIB/ACC*` を使用できます。

第 10 章 ユーザー定義のデータ・ストリームの使用

5250 ディスプレイ・データ・ストリームの制御および処理をシステムに行わせる代わりに、ユーザーがそれを制御および処理できます。そうするには、ディスプレイ・ファイルの DDS 内で USRDFN キーワードを使用しなければなりません。USRDFN キーワードを含むレコード様式の名前を使用する通常の実出力操作を使用して、データがディスプレイ装置に送信されます。出力操作後に入力データを受信したときに、ユーザーのプログラムは、ディスプレイ装置から何を受け取ったかを入力レコード域によってを判別しなければなりません。

注: このサポートを使用するには注意が必要です。なぜなら、エラー条件によっては明白な i5/OS 誤動作が生じることがあるからです。

USRDFN キーワードをレコード・レベルで使用する場合、様式にフィールドは含まれません。したがって、ファイルのバッファ長はデフォルトは、通常レコードの最長または 100 (いずれか大きい方) です。ユーザー定義のデータ・ストリームがこのデフォルトのバッファ長よりも長い場合には、以下のステップを実行して、より大きいバッファ長を取得しなければなりません。

- 外部記述ファイルを定義し、このファイル、およびそのファイル内のレコード様式を使用するプログラムを作成します。様式では、それに対する応答標識を定義すべきではありません。これには、すべてのレコード様式に広がる、ファイル・レベルの標識が含まれます。様式内のフィールドはすべて、*both* (入出力) のフィールド使用を指定して定義する必要があります。
- 2 番目のファイルを作成し、LVLCHK(*NO) を指定します。2 番目のファイルには、以下の 2 つのレコード様式が必要です。
 - 最初のファイル内の様式と同じ名前を持ち、USRDFN キーワードを含む様式。
 - その中に 1 つのフィールドを持つ様式。このフィールドの長さは、画面に送られる最長のユーザー定義のデータ・ストリームと同じ長さでなければなりません。
- プログラムの実行時、最初のファイルは 2 番目のディスプレイ・ファイルでオーバーライドされます。

5250 ディスプレイ・データ・ストリームについて

5250 ディスプレイ・データ・ストリームのコーディングについては、以下の資料を参照してください。

- 「*IBM 5250 Information Display System Functions Reference*」
- 「*IBM 5494 遠隔制御装置 機能参照 マニュアル*」

ディスプレイ装置の違いについて

ユーザー定義のデータ・ストリームを使用している場合、ローカル・ディスプレイ装置 (ディスプレイ装置コントローラーに接続されたもの) とリモート・ディスプレイ装置 (リモート・ディスプレイ装置コントローラーに接続されたもの) との間の、以下の違いを知っておくべきです。

- ローカル・ディスプレイ装置の場合、ヘッダー開始順序長が 11 を超えると、否定応答 16 進数 1005012B が送られます。リモート・ディスプレイ装置の場合、否定応答は送られません。

- ローカル・ディスプレイ装置の場合、それぞれの入力フィールド定義に対して、4 つを超えるフィールド制御ワードを受け入れてはなりません。4 つを超えると、否定応答 16 進数 10050130 が送られます。リモート・ディスプレイ装置の場合、否定応答は送られません。
- ローカル・ディスプレイ装置の場合、自己検査フィールドは、符号付き数字フィールドに 33 バイト、他のすべてのフィールドに 32 バイトの長さを取ることが可能です。長さが超過すると、自己検査フィールドを出ようとしたときに、状況応答 16 進数 00000287 がユーザーに送られます。リモート・ディスプレイ装置の場合、すべてのフィールドは 33 バイトを取ることができます。
- リモート・ディスプレイ装置の場合、Request Maintenance Statistics コマンドの 2 つの形式、つまり、(1) エラー・ログ域をリセットする要求、および (2) エラー・ログ域をリセットしない要求がサポートされます。ローカル・ディスプレイ装置の場合、エラー・ログ域は常時リセットされます。

ユーザー定義のデータ・ストリームの制限について

ユーザー定義のレコード様式の実出力レコード域には、以下のものが指定された順序で含まれなければなりません。

- ディスプレイ装置固有情報 (システムによって必要とされる)。これにより、ディスプレイ装置固有データ・ストリームがシステムによってディスプレイ装置に送られます。
- ディスプレイ装置固有データ・ストリーム。これは、ディスプレイ装置に送られます。

USRDFN キーワードは、レコード様式レベルで指定され、標識およびフィールド関連のすべての機能の使用など、そのレコード様式の他の大半の機能を除外します。ただし、ディスプレイ・ファイルには、USRDFN キーワードを含まない他のレコード様式を含めることができ、プログラムはそれらのレコード様式を任意の順序で使用できます。そのようなディスプレイ・ファイルを使用する場合、以下の事柄に注意する必要があります。

- システム書き込みルーチンは、ユーザー定義の要求を認識すると、ディスプレイに対する前の要求をすべて廃棄します。ユーザー定義の要求の完了時に、システムは、単一のレコード様式が画面上にあり、これが USRDFN キーワードを含むレコード様式であると想定します。すべての削除、すべてのリセット、およびキーボードのアンロックは、ユーザーの責任です。次の入出力要求は、別のユーザー定義の要求であっても、通常のフィールド・レベルの要求であってもかまいません。
- ユーザー定義の要求後の通常のフィールド・レベル要求は、以下のように処理されます。
 - OVERLAY キーワードが指定されていない場合は、要求の実行前に、表示が消去されます。
 - OVERLAY キーワードが指定されている場合には、表示の必要な部分だけ (行全体) が消去されます。
 - 5250 様式バッファがリセットされます。つまり、すべての入力フィールドが出力専用フィールドに変更されます。

システムは、このレコード様式だけが画面上に存在すると想定します。前の要求はすべて廃棄されます。

- ヘルプ仕様は、ユーザー定義のレコード様式で許可されます。
- USRDFN キーワードがレコード様式で指定されると、そのレコード様式に対してフィールドは定義できません。有効なキーワードは、以下のものだけです。

ファイル・レベル	レコード様式レベル
KEEP	ALTNAME
OPENPRT	HELP
PASSRCD	HLPCLR

ファイル・レベル	レコード様式レベル
PRINT	HLPRTN INVITE PRINT TEXT

- ユーザー定義のデータ・ストリームは、CFnn キーおよび CAnn キー、および画面のメッセージ行の位置を変更できます。ただし、ディスプレイ装置サポートは、それらが最後の設定時と同じであると想定します。
- ユーザー定義のデータ・ストリームを含むすべてのディスプレイ・ファイルは、入出力ファイルとしてオープンする必要があります。これは、データ・ストリーム内の読み取りおよび書き込みコマンドが、読み取りおよび書き込み要求に依存していないためです。

出力データ・ストリームは、エスケープ文字 16 進数 04 で開始し、クリア単位 16 進数 40 がその後続くか、または表示コマンド 16 進数 11 に書き込む必要があります。

出力バッファには、適切な回線制御を送受信するために必要な情報が含まれていなければなりません。5250 ディスプレイ装置用のバッファ様式は、以下のとおりです。

バイト	内容
0-1	送信データ長 (16 進数)。出力データ・ストリーム長を定義します。
2-3	受信データ長 (16 進数)。最大入力データ長を定義します。
4	要求された機能
	16 進数 51 送信 (WP モード)
	16 進数 53 送受信 (WP モード)
	16 進数 61 送信 (3270 データ・ストリーム)
	16 進数 63 送受信 (3270 データ・ストリーム)
	16 進数 71 送信
	16 進数 73 送受信
5-n	出力データ・ストリーム

- 3270 モデル 4 ディスプレイ装置は、43 行 x 80 列の画面サイズをサポートします。3270 モデル 4 ディスプレイ装置を使用する場合、機能コード 61 および 63 を使用して、43 行すべてに書き込むことができます。
- 機能コード 61 または 63 を使用する 3270 モデル 4 ディスプレイ装置の場合、システムは行 43 に状況メッセージを表示します。状況メッセージが表示される時、行 43 にあるデータは、状況メッセージによって置き換えられます。
- 機能コード 61 または 63 を使用しており、システム要求キーとして定義されたキーを押した場合、「システム要求」メニューが表示される代わりに、キーがアプリケーション・プログラムに戻されません。機能コード 61 または 63 の使用時にシステム要求機能にアクセスする必要がある場合、「システム要求」メニューを表示するアテンション・キーを定義できます。機能コード 61 および 63 を使用するアプリケーションを開始する前に、以下を指定します。

```
SETATNPGM SET(*ON) PGM(QSYS/QWSSYSRQ)
```

アテンション・キーが押されると、「システム要求」メニューが表示されます。

このサポートは、読み書き (待機なし) 操作の後以外は、読み取り操作の使用を禁止します。読み取りコマンドを送信し、受信データ長を指定する読み書き操作は、読み取り要求で通常実行される操作を実行しま

す。読み書き機能は、以下のいずれか 1 つを行うことによって実行できます。

操作	ILE RPG	ILE COBOL	制御言語
読み書き (待機あり)	EXFMT		SNDRCVF WAIT(*YES)
	INVITE、READ を指定した	INVITE、READ を指定した	SNDRCVF WAIT(*NO)
読み書き (待機なし)	WRITE	WRITE	

読み書き (待機あり) 操作と読み書き (待機なし) 操作はいずれも、同じレコード様式に対する書き込み操作と読み取り操作の組み合わせです。読み書き (待機あり) 操作の使用時は、ディスプレイ装置から入力を受け取るまで、制御がプログラムに戻りません。読み書き (待機なし) 操作を使用する場合は、ディスプレイ装置に入力要求を送信し、入力が届くのを待機せずにプログラムに戻ります。これによりプログラムは、1 つ以上の装置からの入力を要求でき、装置が応答するのを待機せずに処理を継続できます。それぞれの入出力要求について、機能バイトと送信項目数が、送信された実際のデータ・ストリームを実際に反映するようにならなければなりません。受信長は、ディスプレイ装置によって戻されるすべてのデータを収容できるだけの十分な長さでなければなりません。これらのいずれかが条件に合っていない場合には、予期しないまたは受諾不能な動作が生じる可能性があります。たとえば、機能バイトがディスプレイ装置に対する送信を示し、データ・ストリームが変更読み取りコマンドを指定する場合、システムはデータ・ストリームをディスプレイ装置に送信し、読み取りは実行されません。

入力操作では、プログラムの入力バッファに、ディスプレイ装置から受け取ったデータが入ります。たとえば、変更読み取りの完了時、入力バッファには援助機能識別番号 (AID) バイトと、カーソル・アドレス (変更された各フィールドが後に続く) が含まれています。変更された各フィールドの前には、バッファ・アドレス順序および画面上のフィールド・ロケーションが置かれます。

すべての AID バイトは、ユーザー定義のデータ・ストリームのディスプレイ装置サポートによって受け入れられます。Print キーが押されると、システムは印刷機能の実行を試行します。

すべての書き込み操作および読み書き操作は、レコード様式名を指定しなければなりません。読み取り操作が USRDFN キーワードを含むレコード様式の名前を指定し、かつディスプレイ装置に対して適切な読み取り操作の 1 つではない場合には、例外が発行されます。

5250 ディスプレイ装置に対する各要求には、複数のコマンドを含めることができます。各コマンドは、以下に示された適切なシステム操作を使用して要求しなければなりません。

操作	出力バッファ 送信項目数 (バ イト 0-1)	出力バッファ 受信項目数 (バ イト 2-3)	出力バッファ要求機能 (バイト 4)	出力バッファ・コマンド (バ イト 5-n)
書き込み	nn	0	71 (送信)	クリア単位 クリア・フォーマット・ テーブル 表示への書き込み 書き込みエラー行 復元 ロール コピー

操作	出力バッファ 送信項目数 (バ イト 0-1)	出力バッファ 受信項目数 (バ イト 2-3)	出力バッファ要求機能 (バイト 4)	出力バッファ・コマンド (バ イト 5-n)
読み書き	nn	nn	73 (送受信)	クリア単位 クリア・フォーマット・ テーブル 表示への書き込み 書き込みエラー行 復元 ロール 読み取り入力フィールド 読み取り MDT フィールド 即時読み取り 保管

第 11 章 プログラム間でのデータの受け渡し

システム入出力操作を使用して、ジョブの同じ経路指定ステップ内の、または異なる経路指定ステップにあるプログラム間でデータを受け渡すことができます。

ジョブ内の同じ経路指定ステップ内でのデータの受け渡し

同じ経路指定ステップ内のプログラム間でデータを受け渡すために、プログラム間でファイルを共用するか、または KEEP および ASSUME DDS キーワードを使用して、プログラム間でファイルを共用できます。プログラムは同じファイルを開かなければならず、そのファイルは共用ファイルとして指定されなければなりません。(92 ページの『同一ジョブ内でのディスプレイ・ファイルの共用』を参照してください。) プログラムはファイルを共用するので、76 ページの『ディスプレイからの入力の読み取り』および 82 ページの『システムがディスプレイから入力を読み取る方法についての理解』で説明されたように読み取り操作が実行されます。

また、KEEP および ASSUME キーワードを使用して、プログラム間でデータを受け渡すこともできます。データは最初のプログラムによって画面に書き込まれ、2 番目のプログラムは画面からそのデータを使用します。

KEEP キーワードを使用して、プログラムの終了後に検討するため画面上のデータを保持したり、プログラム間でデータを受け渡したりできます。

通常、ファイルをクローズすると、現行画面は消去されます。しかし、KEEP キーワードを使用することによって、これを制御できます。ファイルのクローズ時に画面上にあるレコードに、KEEP キーワードが指定されていれば、システムはそのようなレコードの最初のもの名を保管して、データの受け渡しをサポートします。このキーワードだけでは、受け渡されるデータの処理をサポートできません。次のプログラムによって開かれるディスプレイ・ファイル内に ASSUME キーワードを使用しなければなりません。

ASSUME キーワードを使用すると、ディスプレイ・ファイルが開かれてから、特定のレコード様式名 (または他の任意のレコード様式) に対する先行の書き込み操作がない場合に、その特定のレコード様式名に対する読み取り操作が有効となります。以下は、ASSUME キーワードが指定された場合の典型的な例を示しています。

1. プログラム PGM1 が、ディスプレイ・ファイル DSPFIL1 内にあるレコード様式 PGM1ANY に対する書き込み操作を発行し、プログラム PGM2 を呼び出します。PGM1ANY は LOCK キーワードを指定します。
2. PGM2 が DSPFIL2 を開き、レコード様式 PGM2ANY に対する読み取り操作を発行します。ASSUME キーワードが指定されます。入力データが画面から読み取られ、レコード様式 PGM2ANY によって処理されます。

ASSUME キーワードを指定されたレコード様式のレコードは、画面上で互いにオーバーレイすることはできません。さらに、すべてのレコードは、表示される少なくとも 1 つのフィールドを持っていないけません。

システムが画面から想定レコードを読み取ると、変更データ・タグ (MDT) がオンのフィールドだけが戻されます。(これは、最後の出力操作でフィールドに対して DSPATR(MDT) キーワードが指定されたか、またはユーザーがフィールドに入力したことを前提としています。) ASSUME キーワードが有効な場合、プログラムに戻されるデータは次のとおりです。

- MDT をオンに設定された入力専用フィールドおよび出力/入力フィールドの場合、変更データが戻されます。
- 他の入力可能フィールドの場合は、文字フィールドにはブランクが戻され、数値フィールドにはゼロが戻されます。

さらに、受け取ったフィールドのうち、画面上の行と位置が想定レコードのフィールドと一致するものだけが処理されます。データは、フィールドが画面に書き込まれる方法に関係なく、想定レコード内のフィールド記述を使用して処理されます。

5250 ディスプレイ装置の場合、ファイルを開いた後の最初の書き込み操作または読み書き操作は、CA および CFnn キー仕様をディスプレイ装置に送ります。ASSUME キーワードの使用時に、オープン後の書き込み操作はないので、CFnnkey 仕様は最後のアプリケーションによって残されたままになります。このファイル内の CA および CFnnkey 仕様は、ファイルに対する最初の書き込み操作が終わるまでは、使用されません。

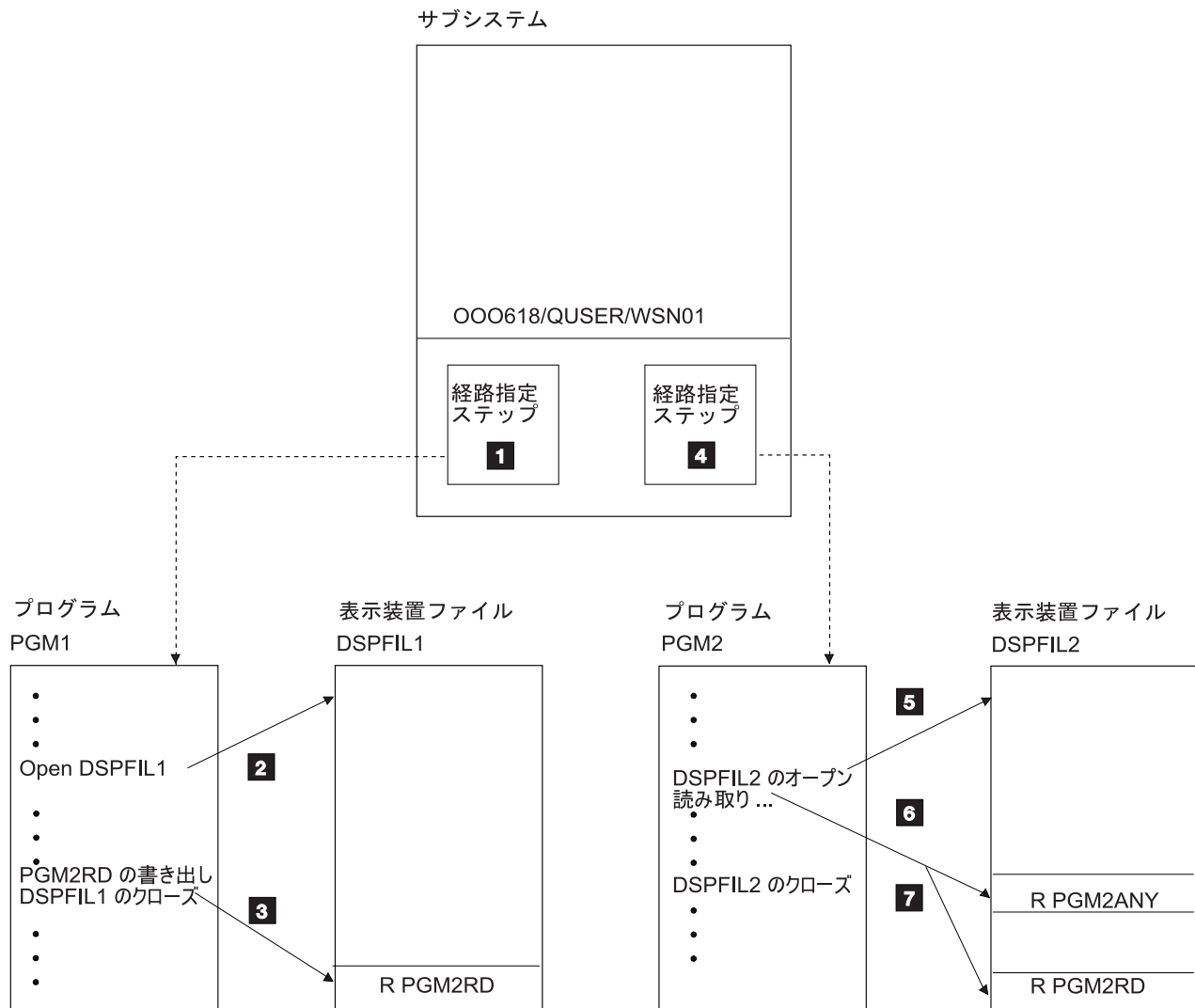
ファイルを開き、最初の書き込み操作が発行された後、OVERLAY キーワードが指定されていない場合には、画面が消去されます。OVERLAY キーワードが指定されている場合、画面上の入力可能フィールドすべては、出力専用フィールドになります。最初の書き込み操作後、想定レコードはプログラムによって読み取ることはできません。

画面上の想定レコードから受け取っていないフィールドは、次のようにして、ユーザー・プログラムに戻されます。

フィールド	戻される初期設定値
DFT キーワードが使用されているフィールド	DFT キーワードで指定した値
文字フィールド (DFT キーワードなし)	ブランク
数値フィールド (DFT キーワードなし)	ゼロ
隠しフィールド	ゼロ

ジョブ内の経路指定ステップ内でのデータの受け渡し

以下の例は、異なる経路指定ステップ内のプログラム間での表示データの受け渡しに使用されるステップを示しています。KEEP、ASSUME、および PASSRCD キーワードの使用に注意してください。プログラム PGM1 および PGM2 は、異なる経路指定項目を使用するサブシステム SBS によって開始されるユーザー・プログラムです。PGM1 および PGM2 は、異なる経路指定ステップで実行されますが、どちらもジョブ 000618/QUSER/WSN01 内に含まれます。



RV2W044-2

- 1** ユーザーがサインオンし、SBS が経路指定データを基にして、経路指定ステップを開始します。経路指定ステップ内の最初のプログラムは PGM1 です。
- 2** PGM1 がディスプレイ・ファイル DSPFIL1 を開きます。
- 3** PGM1 がユーザーと対話し、終了前に以下を発行します。
 - レコード様式名 PGM2RD を持つ DSPFIL1 に対する書き込み操作。KEEP キーワードがレコード様式 PGM2RD 内で指定されます。
 - DSPFIL1 に対するクローズ操作。WSN01 に表示される情報は消去されません。なぜなら、KEEP キーワードがレコード様式 PGM2RD に指定されているからです。
 次いで、経路指定データを指定する RRTJOB コマンドを発行することにより、PGM1 が終了します。結果として SBS が PGM2 を呼び出します。
- 4** SBS が RRTJOB コマンドで PGM1 によって提供されたデータを基に、新規経路指定ステップを開始します。
- 5** PGM2 がディスプレイ・ファイル DSPFIL2 を開きます。
- 6** PGM2 がレコード様式名 PGM2ANY を持つ、または持たない DSPFIL2 に対する読み取り操作を

実行します。レコード様式名が指定されない場合、システムはレコード様式 PGM2RD を使用してデータの処理を試行します。KEEP キーワードが指定されていたからです (ステップ 5)。そうするには、レコード様式 PGM2RD が DSPFIL2 内に存在しなければなりません、それは DSPFIL1 内のレコード様式 PGM2RD と同じである必要はありません。PGM2 の必要とするフィールドだけが同じでなければなりません。(PGM2ANY 内にフィールド記述が存在しないフィールドが、ディスプレイ装置から戻される場合、そのフィールドは無視されます。フィールド記述は妥当性検査を必要とし、受け取ったデータが妥当性検査に不合格だった場合には、PGM2 は読み取り操作を再発行し、現行ユーザーがデータを訂正できるようにします。) レコード様式が指定されない場合、またはレコード様式 PGM2RD が DSPFIL2 に存在しないかまたは ASSUME キーワードが指定されていない場合、KEEP キーワードを介してディスプレイ装置に渡されるデータは、処理できません。(ASSUME キーワードは、渡されたデータを処理するために使用されるレコード様式に対して指定しなければなりません。)

7 PGM2 がデータを処理し、DSPFIL2 に対してクローズ操作を発行します。その結果、画面が消去されます。

PGM2 が終了すると、その経路指定ステップは終了し、画面は SBS の制御に戻り、サインオン画面が表示されます。

第 12 章 ディスプレイ・ファイル、ICF ファイル、およびデータ待ち行列からの入力待ち

データ待ち行列は、ディスプレイ・ファイル、ICF ファイル、およびデータ待ち行列のデータを同時に (どの組み合わせでもよい) 待機するプログラムで使用できます。DTAQ パラメーターを以下のコマンドに指定すると、

- ディスプレイ・ファイル作成 (CRTDSPF) コマンド
- ディスプレイ・ファイル変更 (CHGDSPF) コマンド
- ディスプレイ・ファイル・オーバーライド (OVRDSPF) コマンド
- ICF ファイル作成 (CRTICFF) コマンド
- ICF ファイル変更 (CHGICFF) コマンド
- ICF ファイル・オーバーライド (OVRICFF) コマンド

以下のいずれかが生じた場合に、項目が配置されるデータ待ち行列を指示することができます。

- 送信勧誘ディスプレイ装置で、使用可能なコマンド・キーまたは Enter キーを押した場合。
- データが送信勧誘 ICF セッションから使用可能になった場合。

IBM 提供の QSNDDTAQ プログラムを使用することによって、システム上で実行中のジョブは、DTAQ パラメーターで指定されたのと同じデータ待ち行列に項目を配置することもできます。

アプリケーション・プログラムは、IBM 提供の QRCVDTAQ プログラムを使用して、データ待ち行列に配置された各項目を受け取った後、配置がディスプレイ・ファイル、ICF ファイル、または QSNDDTAQ プログラムのいずれによって行われたかに基づいてその項目を処理します。ディスプレイ・ファイルの場合、アプリケーションは読み取りまたは送信勧誘された装置からの読み取り操作を実行して、データを受信します。QRCVDTAQ 関数と構文、および 1 つ以上のファイルや 1 つのデータ待ち行列を待機する例について詳しくは、iSeries Information Center にある『CL』トピックの CL プログラミング・セクションを参照してください。

データ待ち行列に配置されるディスプレイ・ファイル項目および ICF ファイル項目は、長さが 80 文字であり、288 ページの表 28 で説明されているフィールド属性が含まれています。このため、上にリストされているコマンドを使用して指定されるデータ待ち行列は、少なくとも 80 文字の長さでなければなりません。

QSNDDTAQ を使用してジョブによりデータ待ち行列に配置された項目は、ユーザーによって定義されません。

表 28. ディスプレイ・ファイルおよび ICF ファイル入力フィールド属性

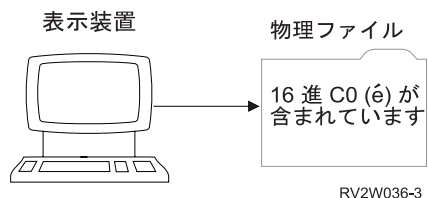
位置	データ・タイプ	意味
1 から 10	文字	<p>データ待ち行列に項目を配置したファイルのタイプ。このフィールドには以下の 2 つの値のいずれかが含まれます。</p> <p>*ICFF (ICF ファイル)</p> <p>*DSPF (ディスプレイ・ファイル)</p> <p>データ待ち行列からデータを受信するジョブが、ディスプレイ・ファイルまたは ICF ファイルを 1 つしかオープンしていない場合、どのタイプの項目がデータ待ち行列から受信されたかを判別するために必要なフィールドは、このフィールドだけです。</p>
11 から 12	2 進	<p>ファイルの固有 ID。ID の値は、ファイルのオープン・フィードバック域にある値と同じです。このフィールドは、同じ名前の複数のファイルがデータ待ち行列に項目を配置する場合にのみ、データ待ち行列から項目を受信するプログラムによって使用されるべきです。</p>
13 から 22	文字	<p>ディスプレイ・ファイルまたは ICF ファイルの名前。これは、すべてのオーバーライドが処理された後、実際にオープンされたファイルの名前で、ファイルのオープン・フィードバック域にあるファイル名と同じです。このフィールドは、複数のディスプレイ・ファイルまたは ICF ファイルがデータ待ち行列に項目を配置する場合にのみ、データ待ち行列から項目を受信するプログラムによって使用されるべきです。</p>
23 から 32	文字	<p>ファイルが入っているライブラリー。これは、すべてのオーバーライドが処理された後のライブラリーの名前で、ファイルのオープン・フィードバック域にあるライブラリー名と同じです。このフィールドは、複数のディスプレイ・ファイルまたは ICF ファイルがデータ待ち行列に項目を配置する場合にのみ、データ待ち行列から項目を受信するプログラムによって使用されるべきです。</p>
33 から 42	文字	<p>すべてのオーバーライドが処理された後のプログラム装置名。この名前は、オープン・フィードバック域のプログラム装置定義リストにあるのと同じ名前です。ファイル・タイプ *DSPF の場合、これは、コマンドまたは Enter キーが押されたディスプレイ装置の名前です。ファイル・タイプ *ICFF の場合、これは、データが使用可能なプログラム装置の名前です。このフィールドは、データ待ち行列に項目を配置したファイルに、データ待ち行列項目を受信する前に送信勧誘された複数の装置またはセッションが含まれる場合にのみ、データ待ち行列から項目を受信するプログラムによって使用されるべきです。</p>
43 から 80	文字	<p>予約済み</p>

第 13 章 代替文字セットおよびコード・ページの使用

多国語環境では、ある国別文字セット内のデータを別の国別文字セットをサポートするディスプレイ装置で表示または入力しなければならない場合があります。これは、アクセント付き文字や発音区別符号 (セディーユの付いた c、波形記号の付いた n、およびウムラウトの付いた u など) の付いた他の文字に特に当てはまります。このセクションでは、これらの文字を**拡張英字**と呼びます。

たとえば、システム上の物理ファイルに基本フランス語コード・ページのデータが含まれており、é (揚音アクセントの付いた e) が含まれているとします。基本フランス語文字セットでは、この文字は 16 進数の C0 です。このデータは、その文字を処理できるディスプレイ装置で入力されたか、通信回線を通して別のシステムからこのシステムに送信された可能性があります。

注: 一部のディスプレイ装置では、拡張英字はキーボードで入力できます (Cmd を押してから、右側に 2 つの発音区別符号が付いているキー、次にその文字を表す 2 つの 16 進数字を押します)。

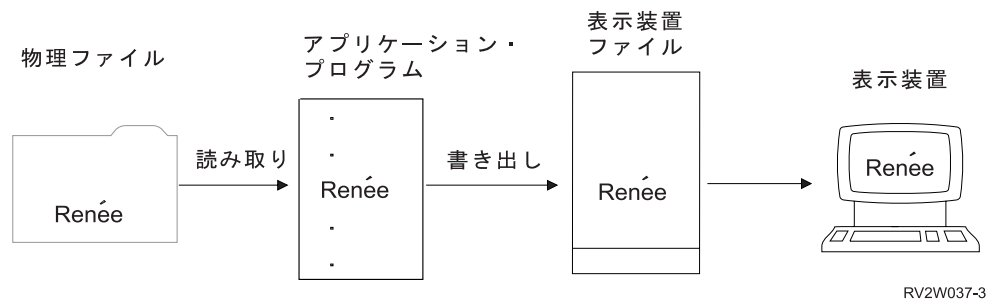


16 進数 C0 は、それを é として認識しないディスプレイ装置に送信された場合、é として表示されません。たとえば、米国コード・ページを認識するディスプレイ装置は、16 進数 C0 を { (左中括弧) として認識します。é は米国コード・ページのコード・ポイント 51 にあります。é を正しく表示するには、米国の基本的なディスプレイ装置での表示用に、16 進数 C0 を 16 進数 51 に変換しなければなりません。

システムは、IBM 提供の変換テーブルを使用して、ほとんどの拡張英字を変換できますが、すべての拡張英字を変換できるわけではありません。ある拡張英字にそれと明確に相当する適切な文字がない場合、システムはその文字をハイフン (-) に変換します。

システムに装置で通常表示されない文字がある

以下の出力/入力フィールドの事例は、文字変換がディスプレイ装置で行われる方法を示しています。



物理ファイル内のレコードに値 Renée を持つフィールドが含まれると想定します。アプリケーション・プログラムは、その物理ファイルからレコードを読み取り、そのデータを含むレコードをディスプレイ・ファイルに書き込みます。文字変換を実行する 1 つの方法は、ディスプレイ・ファイル用の DDS で、Renée

が表示される出力/入力フィールドに文字 ID (CHRID) キーワードを指定することです。¹ このキーワードは、必要に応じてシステムに文字変換を実行するよう要求します。文字変換が必要になるのは、ディスプレイ・ファイルの CHRID パラメーター値が、ディスプレイ装置の CHRID 値と異なる場合です。この場合、ディスプレイ・ファイルに基本フランス語コード・ページおよび文字セットである CHRID(288 297) が含まれているのに対し、装置記述には基本米国コード・ページおよび文字セットである CHRID(101 37) が含まれているため、文字変換が必要になります。

データを表示する際、システムは 16 進数 C0 を 16 進数 51 に変換し、Renée が画面上に表示されます。変換が行われなければ、Ren{e が画面に表示されます。

入力時に、以下のいずれかが行われます。

- 変更データ・タグ (MDT) がフィールドに対してオンになっている場合 (これは、ユーザーがフィールドに入力している場合、または DSPATR(MDT) キーワードがフィールドに対して有効な場合に起こります)、フィールドの内容は、装置 CHRID からファイルの表示 (DSPF) コマンドで指定された CHRID パラメーター値に変換され、プログラムに戻されます。
- 変更データ・タグ (MDT) がフィールドに対してオンになっていない場合、保管されたフィールドの内容は (元の変換されていないデータと共に) プログラムに戻されます。

ファイル・ベースで文字変換を行う別の方法は、ディスプレイ・ファイル作成 (CRTDSPF)、ディスプレイ・ファイル変更 (CHGDSPF)、またはディスプレイ・ファイル・オーバーライド (OVRDSPF) コマンドの CHRID パラメーターで *JOBCCSID 値を指定することです。*JOBCCSID についての詳細情報は、『フィールド用の文字変換を指定する』にあります。

装置がシステム上で表示されない文字を渡す

ユーザーが、通常はシステム上に表示されないデータを入力できる場合 (たとえば、モントリオールにあるフランス語のディスプレイ装置のユーザーが、拡張英字を表示できないディスプレイ装置を持つトロントのシステムにリモート回線を使用してダイヤルアップした場合など)、データがシステムに送信されても何の変更も行われません。プログラムは、フランス語のディスプレイ装置からデータを読み取り、それを物理ファイルに書き込むことができます。しかし、トロントのシステム上で別のユーザーが物理ファイルからデータを表示する場合、システムは、拡張英字を表示できないディスプレイ装置用にデータの変換を試行します。

フィールド用の文字変換を指定する

CHRID キーワードがフィールドで指定されている場合、文字変換が行われるのは、ディスプレイ・ファイルの CHRID パラメーター値が装置記述の CHRID 値と異なる場合のみです。

1. ディスプレイ・ファイルの CHRID パラメーターは、CRTDSPF、CHGDSPF、または OVRDSPF コマンド上の CHRID パラメーターによって設定されます。CHRID パラメーターは、以下の値のいずれかを持つことができます。

*DEVD (デフォルト): アプリケーションが現在実行中の装置記述で指定された CHRID パラメーターを使用します。文字変換は行われません。DDS で指定された CHRID キーワードはすべて無視されます。

1. CRTDSPF、CHGDSPF、または OVRDSPF コマンド上で、CHRID パラメーターによって *JOBCCSID が指定されている場合、DDS キーワード CHRID は無視されます。詳しくは、『フィールド用の文字変換を指定する』の *JOBCCSID に関する説明を参照してください。

*SYSVAL: アプリケーションが現在実行中の、システム値 QCHRID 中で指定された文字セットおよびコード・ページを使用します。変換を必要とするディスプレイ・ファイル・フィールド用の DDS 中に CHRID キーワードを指定します。定数は変換されません。

*JOBCCSID: ジョブが指定した文字セットおよびコード・ページを使用します。これにより、ジョブ CCSID またはディスプレイ・ファイル CCSID が装置記述 CHRID と一致しない場合、指定されたフィールドおよび指定されていない (固定) フィールドを自動的に変換することができます。ジョブ CCSID が装置記述 CHRID と一致しない場合、指定されたフィールドは、出力時にジョブ CCSID から装置記述 CHRID に変換され、入力時にはそれと逆のことが起こります。ディスプレイ・ファイル CCSID が装置記述 CHRID と一致しない場合、ディスプレイ・ファイル内のデータは、出力時にディスプレイ・ファイル CCSID から装置記述 CHRID に変換されます。*JOBCCSID を指定すると、DDS CHRID キーワードは無視されます。変換したくない指定および未指定のフィールドに対して、コード化文字セット ID 変換なし (NOCCSID) キーワードを使用します。

文字セットおよびコード・ページ: 指定された文字セットおよびコード・ページを使用します。有効な値のリストについては、292 ページの表 29 を参照してください。

2. 装置記述の CHRID 値は、装置記述の作成 (表示装置)(CRTDEV DSP) または装置記述の変更 (表示装置)(CHGDEV DSP) コマンドの CHRID パラメーターによって設定されます。このパラメーターは、以下の値のいずれかを持つことができます。

*SYSVAL: システム値 QCHRID 中で指定された文字セットおよびコード・ページを使用します。

*KBDTYPE: キーボード言語プロンプト (KBDTYPE) で指定された文字セットおよびコード・ページを使用します。この値は、キーボード言語 ID に対応します。

文字セットおよびコード・ページ: 指定された文字セットおよびコード・ページを使用します。有効な値のリストについては、292 ページの表 29 を参照してください。

物理ファイルからの UCS-2 レベル 1 のデータを処理する場合、このデータを画面に表示する前に、EBCDIC に変換する必要があります。この変換を実行するには、DDS CCSID キーワードをファイル・レベル、レコード・レベル、またはフィールド・レベルのいずれかに設定し、出力時に UCS-2 データを、キーワードによって指定された UCS-2 CCSID 値から装置の CHRID に変換できるようにします。入力時に、データは装置の CHRID から UCS-2 値に変換されます。詳しくは、iSeries Information Center の『DDS』トピック内の『UCS-2 Level 1 Considerations for DDS』を参照してください。

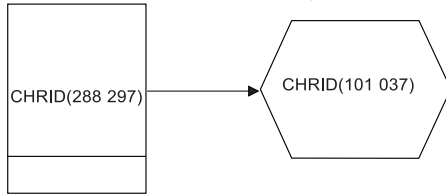
データを表示するたびに行われる別の変換は、出力時に 16 進数 『3F』 文字を 16 進数 『1F』 文字に変換し、入力時にその逆に変換するというものです。Character Data Representation Architecture (CDRA) は、16 進数 『3F』 文字を置換文字として指定します。この文字は、5250 データ・ストリーム仕様のフィールド属性定義でもあります。*JOBCCSID 変換が活動状態であるか非活動状態であるかに関わりなく、出力における 16 進数 『3F』 文字の 『1F』 への変換は、すべてのフィールドについて行われます。フィールド・レベルで変換を防止するには、NOCCSID キーワードを使用してください。

文字変換が必要な場合、システムはライブラリー QUSRSYS 内の変換テーブルを使用してデータを変換します。使用される変換テーブルの名前は、以下の例で示されているように、テーブルを必要とするソースおよびターゲット CCSID の一部から取り出されます。

出力時:

表示装置ファイル

装置記述

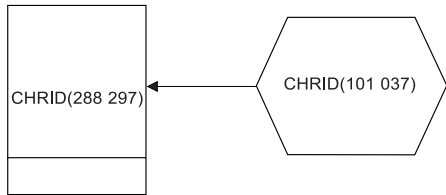


使用される変換テーブル: Q297101037

入力時:

表示装置ファイル

装置記述



使用される変換テーブル: Q037288297

RV2W038-3

IBM は、文字セットおよびコード・ページの最も頻繁に使用される組み合わせ間での変換を処理するためのいくつかのテーブルを、ライブラリー QUSRSYS 内に提供しています。文字の変換方法を調整する必要がある場合には、テーブル作成 (CRTTBL) コマンドを使用してライブラリー QUSRSYS 内にユーザー独自のテーブルを作成し、変換を行うことができます。どの変換テーブルが使用できるかを判別するには、QUSRSYS ライブラリー内でオブジェクト・タイプ *TBL を検索します。

ご使用のディスプレイ用の文字 ID (CHRID) 値の判別

指定される CHRID 値は、ディスプレイ装置の属性を基にしているべきです。下記の表は、それぞれのディスプレイ装置のキーボード・タイプに適切な CHRID 値を示します。一部のディスプレイ装置では、KBDTYPE パラメーターを指定する必要はありませんが、等価キーボードの KBDTYPE 値を使用して、ディスプレイ装置の CHRID 値を判別できます。

表 29. CHRID 値

言語/国または地域	キーボード・タイプ (KBDTYPE)	限定 CHRID	完全 CHRID
国際および US ASCII	INB	103 038	697 500
各国間	AGI BLI CAI DMI FAI FNI FQI ICI INI ITI JEI NEI NWI PRI SFI SGI SPI SSI SWI UKI USI		697 500
アラビア語	CLB		235 420
オーストリア/ドイツ	AGB	265 273	697 273
ベルギー各国間	BLI		697 500
カナダ/フランス語圏	CAB	277 260	341 260
キリル	CYB		960 880
デンマーク/ノルウェー	DMB NWB	281 277	697 277

表 29. CHRID 値 (続き)

言語/国または地域	キーボード・タイプ (KBDTYPE)	限定 CHRID	完全 CHRID
フィンランド/スウェーデン	FNB SWB	285 278	697 278
フランス	FAB FQB	288 297	697 297
ギリシャ	GKB		218 423
ヘブライ語	NCB		941 424
アイスランド	ICB		697 871
イタリア	ITB	293 280	697 280
日本/英語	JEB	297 281	697 281
日本/漢字	JKB (Personal System/55、5295 および 3477-J ディスプレイ装置 用)		332 290
日本/カタカナ	KAB (5251、5291、5292、および 3180 カタカナ・ディスプレイ装 置用)		332 290
韓国語	KOB		933 833
ラテン 2	ROB		959 870
オランダ	NEB		697 037
ポルトガル	PRB	301 037	697 037
中国語 (簡体字)	RCB		936 836
スペイン	SPB	305 284	697 284
スペイン語圏	SSB	309 284	697 284
スイス/フランス語各国間	SFI		697 500
スイス/ドイツ語各国間	SGI		697 500
タイ語	THB		938 838
中国語 (繁体字)	TAB		101 037
トルコ	TKB		965 905
英国/英語	UKB	313 285	697 285
米国/英語	USB	101 037	697 037
旧ユーゴスラビア諸国	YGI		959 870

第 14 章 ディスプレイでのシステム・パフォーマンスの向上

システム・パフォーマンスは、以下を行うことによって向上します。

- 画面の設計時に、送受信されるバイト数を最小限にします。たとえば、アプリケーション表示のフィールド数を削減するには、
 - 多数のフィールドを持つ画面を、複数の画面に分割します。
 - 開発、テスト、または問題報告を支援するために追加されたフィールドの除去を考慮します。
- ディスプレイ装置との間で送受信されるバイト数を削減するシステム機能を利用します。この章では、これらのシステム機能のいくつかを説明します。

ディスプレイ・ファイルの書き込み操作の据え置き

ディスプレイ・ファイル作成 (CRTDSPF) またはディスプレイ・ファイル変更 (CHGDSPF) コマンドの DFRWRT パラメーターによって、必要になるまで、画面へのデータの書き込みを遅らせることができます。書き込み操作の据え置きが有用なのは、ディスプレイ装置に表示される最終画面が、画面に書き込まれるいくつかのレコード様式によって形成される場合です。

ディスプレイ・ファイルで DFRWRT(*YES) を使用すると、システム・パフォーマンスが向上する場合があります。

書き込み据え置きパラメーターについて詳しくは、68 ページの『読み取り要求の実行時までの書き込み操作の据え置き』に記載されています。

ディスプレイ・ファイルの 1 次画面サイズの指定

通常、ディスプレイ・ファイルは 24 x 80 画面 (デフォルトのサイズ) 用にセットアップされます。DSPSIZ キーワードは、ファイルに対して有効な画面サイズを指定し、どのサイズが 1 次および 2 次の画面サイズであるかを示します。(1 次画面サイズは、最初の、またはただ一つの DSPSIZ 値です。) DSPSIZ キーワード上で、画面サイズは *DS3、*DS4、24 80、または 27 132 として指定できます。たとえば、DSPSIZ (24 80) は 24 x 80 の画面サイズを指定します。

1 次画面サイズとして指定された画面サイズは、ディスプレイ・ファイルがそれと共に最も頻繁に使用されるものである必要があります。このような方法で DSPSIZ キーワードをコーディングすることによって、パフォーマンスが向上します。実際の画面サイズが 2 次画面サイズのときには、追加処理が実行されません。

サブファイル・レコードを一度に 1 ページだけ書き込む

複数のページがあるサブファイルの使用時にパフォーマンスを向上させる一つの技法は、サブファイル・レコードを一度に 1 ページだけ書き込み、サブファイルをロールするには i5/OS サポートを使用するという方法です。これを行うには、応答標識と共に DDS 内に ROLLUP キーワードを定義し、また SFLRCDNBR キーワードを使用する必要があります。ご使用のプログラム内で、サブファイルの 1 ページを埋めるのに必要なレコードを書き込み、次いでそのページを表示します。ユーザーがさらに多くのレコー

ドを参照したい場合には、Roll Up キーを押します。次にプログラムは、レコードの別のページをサブファイルに書き込み、その 2 番目のページのレコードの相対レコード番号を SFLRCDNBR フィールドに入れて、レコードを表示します。

サブファイル・レコードの 2 番目のページが現在表示されており、ユーザーが Roll Down キーを押すと、システムによってロールダウンが処理されます。先頭ページの表示中にユーザーが Roll Up キーを押すと、システムもロールアップを処理します。プログラムが通知を受けるのは、ユーザーがサブファイル内の現行レコードを超えてロールアップを試行する場合だけです。その場合、プログラムは追加のロールアップ要求を 2 ページ目の場合と同じ方法で処理します。この技法を使用するときには、ロール・キーを使用するため、サブファイルは複数ページのように見えます。それでも、プログラムは画面に書き込む前に 1 つのサブファイル・ページを埋めるだけなので、優れた応答時間を維持できます。

同じジョブのオープン・データ・パス (ODP) を共有する

i5/OS データ管理機能サポートは、複数のプログラムが、データまたはディスプレイ装置への同じパスを共有できるようにする、ジョブ内の共有のクローズ・レベルを提案します。ディスプレイ・ファイル作成 (CRTDSPF) コマンド、ディスプレイ・ファイル変更 (CHGDSPF) コマンド、およびファイル・オーバーライド・コマンドの SHARE パラメーターに *YES を指定することによって、複数のプログラムがファイル状況、位置、およびストレージ域を共有できます。ジョブが必要とする主ストレージの容量、およびジョブがファイルを開いたり閉じたりするのにかかる時間を削減することにより、パフォーマンスが向上します。

注: SHARE(*YES) を使用してパフォーマンスを向上させるには、ディスプレイ・ファイル内に多数のレコード様式を定義する必要があります。しかし、プログラムがディスプレイ・ファイル内に定義されたすべてのレコード様式に対して割り振られた作業用ストレージを持っていても、プログラムがディスプレイ・ファイル内の必ずしもすべてのレコード様式を使用するとは限りません。このことは結果として、ジョブに関連した PAG ストレージを大きくし、それはパフォーマンスに悪影響を与える可能性があります。

オープン・データ・パスについての詳細は、92 ページの『同一ジョブ内でのディスプレイ・ファイルの共有』に記載されています。

入力フィールドを持つレコードを、順番に画面に送信する

入力フィールドを含むレコードは、画面に表示される順番で、ディスプレイ装置に送信する必要があります。この技法は、入力フィールドを持つレコード様式がランダムに、または他の何らかの順序で送信される場合よりも、より優れたパフォーマンスを提供します。

繰り返し送信されるレコードを重複させ、削除しない

CLRL(*NO) キーワードを使用すると、オーバーラップするレコードが画面に書き込まれるときに、オーバーラップされるレコードが削除されないようにすることができます。このキーワードを使用する場合、表示中の重複レコードは、画面から削除されません。新規レコードがそれらを完全に、または部分的にオーバーレイします。CLRL(*NO) を使用することによってパフォーマンスが向上するのは、画面に繰り返し送信される固定情報およびデータを含む表示がある場合です。固定情報を別個の様式として送信し、データだけを含む様式に対して CLRL(*NO) を使用することにより、画面にレコード様式を送信するために必要な時間を削減できます。

PUTOVR キーワードを使用すると、同じレコード様式に対して後続の書き込み操作または読み書き操作が発行されたときに、OVRDTA または OVRATR キーワードが指定されたフィールドが、画面に送信されません。

また、OVERLAY キーワードを使用すると、書き込まれるレコード様式によって影響を受ける画面部分だけを消去できます。

ERASEINP を使用すると、画面にブランクを送信するよう要求する代わりに、画面にフィールドを消去させることによって、応答時間を改善できます。画面の消去されるフィールドの、次の読み取り操作の変更データ・タグがオンに設定されていない場合には、それらのフィールドに対して入力保管域からデータが戻されます。これは、前にディスプレイ装置から戻ってきたフィールドの、システムによって保管されるデータです。

INZINP キーワードをレコード・レベルで ERASEINP(*ALL) および PUTOVR と一緒に使用すると、消去されたフィールドのデータをディスプレイに送信することなく、入力保管域を初期化できます。

画面の復元

画面の復元 (RSTDSP) パラメーターに *YES を指定すると、画面の中断時に現行画面のイメージが保管されます。ディスプレイ・ファイルが再度活動状態になったときに、保管されたイメージを使用して、画面をその中断前の外観に復元します。

RSTDSP(*YES) パラメーターは、以下のキーワードに対して指定しなければなりません。パラメーターが指定されない場合には、ファイルが中断すると、画面上のデータが失われることがあります。

- CLRL
- OVERLAY
- PUTOVR
- PUTRETAIN
- ERRMSG
- ERRMSGID

これらのキーワードのいずれも使用されない場合には、RSTDSP パラメーターに *NO を指定することによって、パフォーマンスを改善できます。

コマンド機能キーではなくコマンド・アテンション・キーの定義

コマンド・アテンション (CAnn) キーは、データではなく、押したキーの標識だけを戻します。コマンド機能 (CFnn) キーは、すべての入力データを戻します。機能キーの検出だけが必要で、画面からデータを戻す必要がない場合には、コマンド・アテンション・キーとしてキーを定義します。

送信勧誘操作の使用

INVITE キーワードによって指定される送信勧誘操作を使用すると、さまざまなディスプレイ装置およびシステム・イベントにプログラムが対応できるようになります。送信勧誘操作はまた、最後の WRITE および READ 操作間をプログラムが処理できるようにします。

ウィンドウの使用

DDS 記述ウィンドウは、それらが置かれた画面部分だけに影響を与えるので、パフォーマンスを向上させることがあります。

第 3 部 パネル・グループを使用したアプリケーション表示のプログラミング

第 15 章 ユーザー・インターフェース管理機能による生産性の向上	301
ユーザー生産性の増大	301
アプリケーション・プログラマーの生産性の増大	301
データ記述仕様 (DDS) の代わりに UIM を使用する前に考慮すべき事柄	301

第 16 章 ユーザー・インターフェース管理機能の概要	303
UIM の概要	303
UIM のサポート対象	304
パネル・グループとは	304
メニューとは	305
オブジェクトの作成	305
パネル・グループ内のエレメント	305
UIM 言語タグの使用法	306
ダイアログ・コマンドの使用法	306
制御言語 (CL) コマンドの使用法	307
アプリケーション・プログラミング・インターフェース (API) の使用法	307
UIM を使用したメニュー・オブジェクトの定義	308
メニュー・パネルの作成	309
メニュー・パネルに必要なタグ	310
メニューの例のソース	311
UIM を使用したパネル・グループ・オブジェクトの定義	315
リスト・パネルの作成	316
リスト・パネルの必須タグ	318
リスト・パネルの例のソース	319
リスト・パネルのアプリケーション・プログラミング	327
確認リスト・パネルの作成	329
確認リスト・パネルに必要なタグ	330
確認パネルの例のソース	330
自動確認処理	332
確認処理のアプリケーション・プログラミング	333
データ表示パネルの作成	333
データ表示パネルに必要なタグ	336
データ表示パネルの例のソース	337
データ表示パネルのアプリケーション・プログラミング	347
データ入力パネル	350
メニュー・バーのあるパネルの作成	351
メニュー・バーのあるパネルに必要なタグ	353
メニュー・バーのあるパネルの例のソース	354
メニュー・バー・パネルのアプリケーション・プログラミング	363

第 17 章 ユーザー・インターフェース管理機能の詳しい使用方法	365
UIM アプリケーションを開く	365
ダイアログ変数の定義	365
ダイアログ変数の使用上の制約事項	367
ダイアログ変数のエラー・メッセージ	367
ダイアログ変数を使用する表示パネルへのフィールド値の提供	368
変数プール・サービスの使用法	368
ダイアログ変数および特殊値	369
文字セットおよびコード・ページについての考慮事項	369
表示	369
印刷	370
リストの管理	371
リストの定義	372
リストの初期化	372
リストの表示	372
リストの更新	373
不完全リストの処理	373
リストからの項目の除去および挿入	373
リスト画面上のリスト項目の制御	374
リスト画面の対話応答時間の改善	374
処置リストおよび選択リストの使用	375
処置リストの使用	375
選択リストの使用	376
選択文字の使用	377
パネル機能の管理	377
GUI への変換の使用可能化	378
スクロールのサポート	378
スクロール可能域の定義	378
機能キー・スクロールの定義	378
スクロールおよびエラー条件	379
リスト域のスクロール	379
メニュー域のスクロール	379
情報域のスクロール	379
データ項目グループのスクロール	380
テキスト域のスクロール	380
文脈ヘルプの定義	380
コマンド行の制約事項	382
コマンド行の解釈	382
長すぎるコマンドの入力	383
機能キーの定義	383
機能キーの書式設定	383
機能キーおよび VARUPD 値の処理	384
パネルの書式設定の概念	385
パネルの書式設定をいつ行うか	385
パネルの書式設定のアプリケーション制御	386
パネル・フォーマッターの制限	387

EXIT の要求時に複数のパネルをまとめて閉じる	387
リスト・パネルをまとめて閉じる	388
別のパネルへのポップアップ・ウィンドウの追加	389
メニュー・バーの使用方法	390
プルダウン・メニューとポップアップ・ウィンドウとの違い	391
ポップアップ・ウィンドウの使用方法	393
アプリケーション・ウィンドウの定義	393
ウィンドウの追加および除去	393
ウィンドウ内でのコマンド行の使用方法	394
パネルを表示する際の要求処理プログラムとしての UIM	395
印刷の概念	396
印刷ヘッド・パネルの印刷	398
印刷パネルの印刷	398
分離のためブランク行の使用	398
フォントおよび強調表示	398
トレーラーの印刷	398
プロログ域の定義	398
ヘッダー域の定義	399
印刷中のページ排出機能の使用方法	399
プリンター・ファイルの共用およびオーバーライド	399
2 バイト文字セット (DBCS) 印刷の考慮事項	400
UIM に関するよく尋ねられる質問	400

第 15 章 ユーザー・インターフェース管理機能による生産性の向上

ユーザー・インターフェース管理機能 (UIM) を使用すると、ユーザーおよびアプリケーション・プログラマーの生産性を向上できます。

ユーザー生産性の増大

UIM は、パネルの外観を制御し、IBM により開発されたパネルとの整合性を確保します。これは、アプリケーションに一貫性のあるユーザー・インターフェースを提供するので、ユーザーの生産性を高めます。

アプリケーション・プログラマーの生産性の増大

アプリケーション・プログラマーの生産性は、UIM を使用してパネルを作成することによって高まります。アプリケーション・プログラマーは、詳細すべてを指定しなくても、パネルのコンテンツを記述できます。パネル・エレメントの様式は、UIM によって自動的に処理されます。UIM の処理するパネル・エレメントについては、304 ページの『UIM のサポート対象』を参照してください。

UIM 言語タグを使用すると、アプリケーション・プログラマーはオプション番号または機能キーを、特定のコマンドまたはプログラムに「リンク」できます。そうすると、ユーザーがオプションを選択するか、機能キーを押したとき、UIM がコマンドまたはプログラムの実行を自動的に処理します。UIM はまた、複数ページのパネルのスクロールを処理します。「メジャー・コマンド・グループ」メニューは、複数ページ・パネルの例です。

データ記述仕様 (DDS) の代わりに UIM を使用する前に考慮すべき事柄

UIM および DDS はそれぞれ、異なるソース言語を使用して、対話式表示を作成します。対話式アプリケーションの表示機能は、DDS、DDS と UIM の組み合わせ、または UIM のみを使用して作成できます。DDS を使用すると、アプリケーション・プログラマーは表示画面をカスタマイズできます。また、UIM を使用すると、表示画面が自動的にフォーマット設定され、結果として一貫性のある外観となります。さらに、UIM は DDS よりも多くのダイアログ管理機能を実行するので、UIM を使用すれば、アプリケーション・プログラミングの労力が少なくて済みます。

UIM は、以下のタイプの表示画面 (またはパネル) をサポートします。それぞれは、スクロール可能またはスクロール不可のいずれでもかまいません。

- リスト
- メニュー
- 情報
- データ入力項目

これらの表示画面の作成について詳しくは、303 ページの『第 16 章 ユーザー・インターフェース管理機能の概要』を参照してください。

UIM を使用する前に、以下の事柄を考慮してください。

- 新規アプリケーションを作成している場合、UIM は標準の表示画面の作成を単純化します。

- 古いアプリケーションを再書き込みしている場合は、表示画面の再書き込みを行うための労力と、上記の生産性の利点とを比較考量してください。他の理由でアプリケーションの大掛かりな再設計や再書き込みが必要なでない限り、UIM への変換は保証されない場合もあります。
- UIM は 525x 非プログラマブル・ワークステーション用に設計されたものなので、標準表示画面を確保するのであれば、UIM を使用しても、アプリケーションのプログラマブル・ワークステーション向きへの変換は、さほど単純化されません。
- アプリケーション・プログラムがデータベース・ファイルを広範囲に使用する場合、UIM は DDS が行うようにファイル記述を利用することはしません。
- データ・パネルを作成するための UIM の使用は、頻度が低くボリュームも小さい入出力に対して行われるべきです。対話の頻度が高くボリュームが大きい、スクロールできないデータ入力アプリケーションでは、最良のパフォーマンスを得るために、DDS から作成されたディスプレイ・ファイルの使用を考慮すべきです。なぜなら、UIM は頻度が高くボリュームの大きいキーボード入力を、DDS が行うと同じほど迅速には受け入れられないからです。つまり、UIM はキーボード入力を、ユーザーが入力するのと同じ速度で受け入れることはできません。ただし、先行入力機能を持つキーボードは、これを補うことができます。

以下は、UIM を使用する場合と DDS を使用する場合の、それぞれ利点と違いのいくつかをリストしています。

- UIM の利点
 - システムと同じ標準を使用するので、標準を再定義する必要はありません。ユーザー・アプリケーションは、システム・パネルと同じ仕方で作動します。UIM は、ユーザーが希望する表示を基にして、パネルをフォーマットします。
 - リスト処理によって、コマンドの処理、リスト・パネルから簡単にプロンプトを出すことやプログラムを呼び出すこと、またオプションの選択後に UIM が呼び出すプログラムの指定などの機能が提供され、より効率的なリスト項目アクセスまたは更新処理が可能になります。
 - 構造を効率的に処理する言語で、良く作動します。
 - さらに多くのモジュラー・プログラミング技法を提供します (1 つのプログラムが、不完全なリスト出口呼び出しすべてを処理し、すべてのアプリケーションを開くことができる、など)。
 - メニュー・オプションの条件を決定する機能を提供します。
 - ユーザー・プログラムの介入なしに、大容量域のスクロールをフォーマットし処理します (データ、リスト、情報、メニュー、機能キー域など)。
- DDS の利点
 - 画面設計におけるより高い柔軟性の提供
 - 初期フォーマットにおける SDA ヘルプ
 - UIM ヘルプまたはフォルダー内のヘルプを使用する機能
 - GUI ウィンドウを利用する機能
 - サブファイル処理
 - EDTCDE および EDTWRD およびユーザー定義の編集の使用
 - セットアップ時間の短縮による、より小型のアプリケーションに対する高速処理

UIM を使用する場合の追加の考慮事項については、404 ページの『ヘルプにパネル・グループとレコードのいずれかを選択する』を参照してください。

第 16 章 ユーザー・インターフェース管理機能の概要

この章では、IBM が i5/OS パネルの開発に使用するユーザー・インターフェース管理機能 (UIM) についての概要を示します。ここではパネルの作成と管理を簡単にするため、およびパネルに i5/OS パネルの外観を付与するために UIM が提供する機能について説明します。

UIM について詳しくは、365 ページの『第 17 章 ユーザー・インターフェース管理機能の詳しい使用方法』を参照してください。

この章で言及される例の使用方法については、ライブラリー QUSRTOOL 内のソース・ファイル QATTINFO にあるソース・メンバー T0011INF を参照してください。

UIM の概要

i5/OS UIM は、アプリケーション用にパネルおよびダイアログを定義可能にするシステムの一部で、以下のサポートを提供します。

- データおよびパネルを記述するためのタグ・ベースの言語。
- タグ・ベースの言語を使用してパネル・グループ・オブジェクトおよびメニュー・オブジェクトを作成するコンパイラー。
- パネルを表示および印刷するパネル・グループ・オブジェクトとして使用する、アプリケーション・プログラミング・インターフェース (API) のセット。
- UIM は、以下の機能も提供します。
 - 画面管理のためのダイアログ・コマンド。
 - 文脈オンライン・ヘルプ
 - 索引検索
 - ポップアップ・ウィンドウ
 - メニュー・バー
 - CL コマンドを入力するためのコマンド行
 - さまざまなユーザーまたは環境に合わせたパネル内容の調整
 - メニュー・ネットワークでのファースト・パス
 - 2 バイト文字セット (DBCS) 言語
 - 両方向 (BIDI) 言語サポート
 - グラフィカル・ユーザー・インターフェース (GUI) サポート
 - UCS-2 サポート

UIM は、メニュー、情報画面、リスト画面、入力画面などの共通パネル・タイプをサポートします。すべての画面タイプおよびインターフェースに一貫性があれば、ユーザーは新しいアプリケーションにより早く適合できます。

UIM アプリケーションは、UIM の制御下でない他のオープン・ディスプレイ・ファイルと共存し、要求元のディスプレイ装置を共用できます。ただし、UIM パネルと DDS 定義のレコード様式とを同時に画面に表示することはできません。UIM パネルが DDS パネルを置き換えるとき、またはその逆のときには、システムは必要に応じて 1 つのファイルまたはパネル・グループの操作を中断し、画面をリストアします。

UIM のサポート対象

パネルを設計するとき、UIM は以下のような多数のパネル・エレメントに対して正しい配置および様式を提供します。

- パネル名
- パネル・タイトル
- 区切り行
- 列見出し
- 入力フィールド
- コマンド行
- メッセージ行
- 機能キー
- ポップアップ・ウィンドウ
- メニュー・バー
- プルダウン・メニュー

さらに、UIM は以下のサポートも提供します。

- カーソルの正しい配置。

ユーザー・インターフェース・スタイルには、UIM がサポートするカーソルの配置に関していくつかの規則があります。

- カーソル移動に影響されるヘルプ情報

ヘルプ・キーが押されたときのカーソルの位置に応じて、ユーザー・インターフェース・スタイルには、表示されるヘルプ情報のタイプを決める規則があります。

- スクロール
- メニュー・ネットワークでのファースト・パス
- 翻訳を容易にするための、各国語に関する配慮
- ヘルプ情報およびアプリケーション・パネルのためのウィンドウ
- 132 カラムのパネル
- 左から右および右から左 (両方向) の各国語のフォーマット設定
- ハイパーテキスト・リンク
- テキストを左から右および右から左に方向付けることに関する制御
- オンライン・ヘルプ情報の作成機能
- パネルをグラフィカル・ユーザー・インターフェース (GUI) への変換用に使用可能にする機能

パネル・グループとは

パネルは、画面にデータをビジュアルに表示したものです。パネル・グループは、表示形式、印刷形式、またはヘルプ情報のいずれかが集まったオブジェクトです。オブジェクト・タイプのシステム認識 ID は *PNLGRP です。

パネル・グループ (PNLGRP) 言語タグについては、665 ページの『PNLGRP (パネル・グループ)』を参照してください。

メニューとは

メニューは、1 つ以上のオプションのあるパネルの定義を含むオブジェクトです。ユーザーはパネルからオプションを選択して、プログラムまたはコマンドの使用を開始したり、他のメニューに移動することができます。オブジェクト・タイプのシステム認識識別コードは *MENU です。

i5/OS には、メニューを表示するための GO コマンドが備わっています。そのため、メニュー用のユーザー対話を表示および処理するアプリケーション・プログラムは必要ありません。

メニュー作成 (CRTMNU) コマンドを使用して、以下のタイプのメニューを定義できます。

***DSPF** 既存のディスプレイ・ファイル (*DSPF) およびメッセージ・ファイル (*MSGF) を使用して、メニューを表示します。

***PGM** i5/OS は、アプリケーション定義のプログラムを呼び出してメニューを表示します。このプログラムには、メニューをユーザーに表示して、ユーザーが要求したオプションを処理する役割があります。

***UIM** メニュー・オブジェクトは、メニューの記述を含むソース・ファイル内のメンバーを使用して作成されます。ソースは、UIM タグを使用してメニューを記述します。

この章では、*UIM タイプのメニューを作成する方法について説明します。*DSPF または *PGM タイプのメニューを作成する方法については、261 ページの『第 9 章 ディスプレイ・ファイルを使用したメニューの作成およびアクセス』を参照してください。

オブジェクトの作成

UIM は、以下のオブジェクトを作成および変更します。

- パネル・グループ・オブジェクト
- メニュー・オブジェクト
- 検索索引オブジェクト

パネル・グループおよびメニュー・オブジェクトには、パネル定義とオンライン・ヘルプ情報が含まれます。パネル・グループ定義を含むメニュー・オブジェクト (*MENU) は、メニュー作成 (CRTMNU) コマンドで TYPE(*UIM) パラメーターを使用して作成されます。パネル・グループおよびメニュー・オブジェクトは、UIM エレメントの定義を指定するタグ・ベースの言語を使用して作成されます。

UIM は、オンライン・ヘルプ情報から抽出された検索項目を含む検索索引オブジェクトを作成します。検索索引オブジェクトにより、ユーザーは索引検索機能を使用して特定のオンライン情報をより効率的に見つけられるようになります。

パネル・グループ内のエレメント

パネル、メニュー、およびオンライン・ヘルプの定義に使用されるタグ・ベースの言語は、以下を含むシンボリック・エレメントの定義もサポートします。

- 変数クラス
- ダイアログ変数やリストなど、UIM アプリケーション・プログラミング・インターフェース (API) を介してアクセス可能なデータ・エレメント。
- アプリケーション・プログラムによって受け渡されるバッファーを定義する変数レコード
- 特定の処理を実行するために真でなければならない条件式

- 機能キーの定義を含むキー・リスト
- 1 つ以上のプルダウン・メニューの定義を含むメニュー・バー
- データ、情報、リスト、およびメニューを表示する 1 つ以上の領域を含むパネル定義
- オンライン・ヘルプのテキスト・モジュール

UIM 言語タグの使用法

言語タグは、パネルを定義するための使いやすい機能です。言語タグの使用は、アプリケーション・プログラマーが表示方法に一貫性のあるパネルを作成するために役立ちます。言語タグについて詳しくは、511 ページの『付録 A. UIM パネル・グループ定義言語』を参照してください。

ダイアログ・コマンドの使用法

ダイアログ・コマンドは、UIM だけによって認識される特別の機能で、ユーザーが複数のパネル内をナビゲートできるようにします。

ダイアログ・コマンドとそれらが実行する機能のリストを以下に示します。

ACTIONS	カーソル位置をパネルとメニュー・バーとの間で切り替えます。プルダウン・メニューが表示されている場合、それをパネルから除去します。
CALL	機能を実行するアプリケーション・プログラムを呼び出します。
CANCEL	パネル 1 つ分、後退します (直前のパネルに戻ります)。
CHGVIEW	表示する列の定義済みセットを切り替えて、表示されるリストのビューを変更します。
CMD	i5/OS CL コマンド (System/36 環境の OCL コマンド) を処理のためにシステムにサブミットします。
CMDLINE	コマンド行のあるポップアップ・ウィンドウを表示します。
DSPHELP	ヘルプ・テキストのモジュールを表示します。
ENTER	アクションの処理を開始します。パネル入力を処理のためにサブミットします。
EXIT	ユーザーを画面またはメニューのグループから戻します。
EXTHELP	パネルの全般ヘルプ・テキストを表示します。
HELP	カーソルの位置を基にして、パネルのヘルプ情報を表示します。
HELPHELP	ヘルプ機能の使用法に関する情報を表示します。
HELPIDX	アプリケーションの索引検索機能を開始して、ユーザーが検索を要求できるようにします。
HOME	ジョブの初期 (ホーム) メニューを表示します。
KEYSHELP	表示されたパネルに示されている機能キーのヘルプを表示します。
MENU	メニュー項目を選択したり機能キーを押した結果として、後続のメニューを表示します。
MOREKEYS	アクティブな機能キーの追加セットおよびそれらの説明を表示します。すべてのキーを同時に表示できないときに使用されます。
MOVETOP	カーソルで選択された行をスクロール可能な情報域の先頭に移動します。
MSG	メッセージ行にメッセージを表示します。
PAGEDOWN	1 パネルずつ前方にページ送り (スクロール) します。

PAGEUP	1 パネルずつ後方にページ送り (スクロール) します。
PRINT	現行の画面に表示されたすべての情報を印刷します。
PROMPT	コマンド、処置リスト・オプション、または入力フィールドの (入力を求める) プロンプトを出します。
PULLDOWN	メニュー・バーに示された最初の選択項目に対して、プルダウン・メニューを表示します。
RETRIEVE	直前に入力されたコマンドを取得して表示します。
RETURN	制御 (および戻り値) を処理のためにアプリケーションに戻します。

注: アプリケーションに対してリフレッシュ/再表示機能を提供するために、RETURN ダイアログ・コマンドを使用してください。

ダイアログ・コマンドについて詳しくは、『付録 B. UIM ダイアログ・コマンド』を参照してください。

制御言語 (CL) コマンドの使用法

以下の CL コマンドを使用して、パネル・グループ、メニュー、および索引検索に対する作業を行うことができます。

- ADDSCHIDX (検索索引項目追加)
- CHGMNU (メニュー変更)
- CHGSCHIDX (検索索引変更)
- CRTMNU (メニュー作成)
- CRTPNLGRP (パネル・グループ作成)
- CRTSCHIDX (検索索引作成)
- DLTMNU (メニュー削除)
- DLTPNLGRP (パネル・グループ削除)
- DLTSCHIDX (検索索引削除)
- DSPMNUA (メニュー属性表示)
- GO (メニュー表示)
- RMVSCHIDX (検索索引項目除去)
- STRSCHIDX (検索索引開始)
- WRKSCHIDX (検索索引項目処理)

CL コマンドについて詳しくは、iSeries Information Center の『CL』のトピックを参照してください。

アプリケーション・プログラミング・インターフェース (API) の使用法

UIM は、いくつかの API サービスを提供します。UIM へのサービス呼び出しは、以下のものから構成されます。

- アプリケーション・サービス
- 変数プール・サービス
- リスト・サービス
- 表示サービス
- 印刷サービス

アプリケーション・サービスは、以下のものから構成されます。

- Open Display Application (QUIOPNDA)
- Open Print Application (QUIOPNPA)
- Close Application (QUICLOA)

変数プール・サービスは、以下のものから構成されます。

- Get Dialog Variable (QUIGETV)
- Put Dialog Variable (QUIPUTV)

リスト・サービスは、以下のものから構成されます。

- Add List Entry (QUIADDLE)
- Add List Multiple Entries (QUIADDLM)
- Get List Entry (QUIGETLE)
- Get List Multiple Entries (QUIGETLM)
- Update List Entry (QUIUPDLE)
- Remove List Entry (QUIRMVLE)
- Delete List (QUIDLTL)
- Set List Attributes (QUISETLA)
- Retrieve List Attributes (QUIRTVLA)

表示サービスは、以下のものから構成されます。

- Display Panel (QUIDSPP)
- Display Help (QUHDSPH)
- Add Pop-up Window (QUIADDPW)
- Remove Pop-up Window (QUIRMVPW)
- Set Screen Image (QUISETSC)
- Display Long Text (QUILNGTX)

印刷サービスは、以下のものから構成されます。

- Print Panel (QUIPRTP)
- Print Help (QUHPRTH)
- Add Print Application (QUIADDPA)
- Remove Print Application (QUIRMVPA)

API について詳しくは、iSeries Information Center の『**API**』のトピックを参照してください。

UIM を使用したメニュー・オブジェクトの定義

UIM を使用して、*UIM タイプのメニュー・オブジェクトを作成できます。メニュー・オブジェクトには、1 つ以上のオプションを持つメニュー・パネルの定義が含まれます。 *UIM タイプのメニューを作成するには、以下のように行います。

1. ソース物理ファイルにメンバーを作成します。

2. 原始ステートメント入力ユーティリティ (SEU) などのソース・エディターを使用して、メニューを記述する UIM タグ・ソースを入力します。メニューの例は、ライブラリー QUSRTOOL 内のファイル QATTUIM にあるメンバー T0011MN2 内にあります。この例をコピーして、独自のメニューを作成するためのテンプレートとして使用できます。
3. CRTMNU (メニュー作成) コマンドを使用して、タグ・ソースを入力にすることにより、メニュー・オブジェクトを作成します。CRTMNU コマンドを使用する例を、以下に示します。

```
CRTMNU MENU(MYLIB/MYMENU) TYPE(*UIM) SRCFILE(MYLIB/QMNUSRC)
```

ソース・メンバーは、デフォルトで、作成されるメニューの名前となります。

メニュー・パネルの作成

図 102に示したパネルは、メニューの例です。メニュー域のヘルプを作成する方法については、426 ページの『メニュー域のヘルプ』を参照してください。

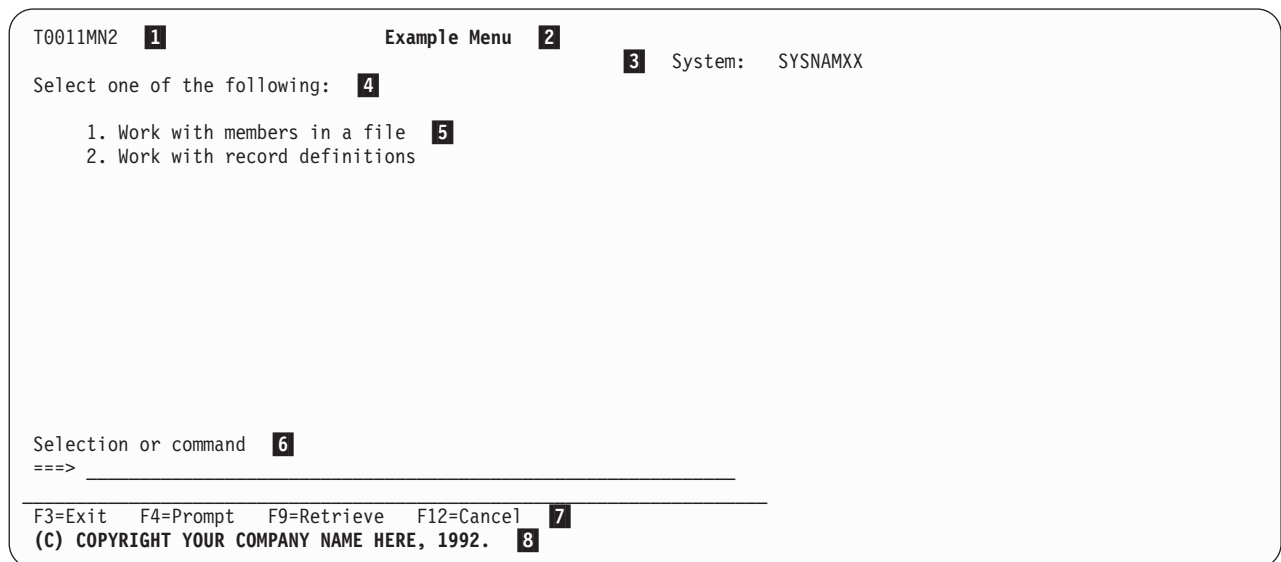


図 102. メニューの例

メニューの例に示されている参照番号は、画面には表示されません。これらは図示のためのものであり、311 ページの『メニューの例のソース』に示されている UIM タグ・ソース内にもあります。これらの参照番号は、メニューに表示されるテキストおよび情報を定義するソースの特定部分を示します。

- 1 パネル ID は、VAR タグを使用して UIM 定義の Z 変数 ZMENU を宣言することにより、そして PANEL タグの PANELID 属性で Z 変数をパネル ID として使用するよう指定することにより、定義されます。
- 2 パネル・タイトルは、PANEL タグのピリオドに続くテキストとして定義されます。
- 3 システム名は、PANEL タグで TOPSEP=SYSNAM 属性が指定されていたために表示されます。
- 4 最上部指示行は、最上部指示 (TOPINST) タグのピリオドに続くテキストとして定義されます。
- 5 メニューの各オプションは、メニュー項目 (MENU) タグを使用して定義されます。オプション番号は OPTION 属性を使用して定義され、オプション番号に続くテキストは MENU タグに続くテキストとして定義されます。

6 コマンド行は、コマンド行 (CMDLINE) タグを使用して定義されます。コマンド行のプロンプト・テキストは、CMDLINE タグに続くテキストとして定義されます。UIM はすべてのコマンド行に矢印 (==>) を自動的に表示します。

オプション行は、CMDLINE タグの代わりにオプション行 (OPTLINE) タグを使用することで定義できます。この場合、ユーザーはオプション番号を入力しますが、システム・コマンドを入力することは許可されません。

7 各機能キーは、キー・リスト項目 (KEYI) タグを使用して定義できます。機能キー名を含むテキストは、KEYI タグのピリオドに続くテキストとして定義されます。

パネルのすべての機能キーは、KEYI タグをキー・リスト (KEYL) および EKEYL タグの間に配置することにより定義されます。KEYL タグの NAME 属性に指定された名前は、PANEL タグの KEYL 属性にも指定されます。

Enter キーなどの、いくつかの機能キーには、KEYI タグに指定されたテキストがありません。この場合、画面にはテキストが表示されません。ただし、すべての機能キーは KEYI タグを使用して定義して、各キーにどの操作が割り当てられているか、および機能キーのヘルプを表示する際にどのヘルプ・モジュールを表示するかを UIM が認識するようにする必要があります。

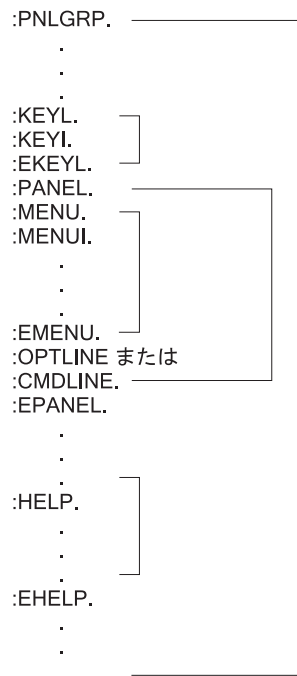
8 著作権メッセージは、著作権 (COPYR) タグのピリオドに続くテキストとして定義されます。著作権メッセージは、ユーザーが初めてメニューを表示するときに一度表示されます。オプションまたはコマンドを処理した後にメニューが再表示される場合は、このメッセージは表示されません。

このメニューの例には 2 つのオプションしか含まれていないので、スクロール可能として定義されません。スクロール可能メニューを定義する必要がある場合、MENU タグで SCROLL=NO を SCROLL=YES に変更してから、追加のメニュー・オプションごとに 1 つのメニュー項目 (MENU I) タグを追加します。さらに、メニュー・オプションごとに新しいヘルプ・モジュールを定義する必要もあります。

メニュー・パネルに必要なタグ

311 ページの図 103は、UIM メニュー・パネルを作成するために必要なタグを示しています。必須タグおよびオプション・タグの例については、311 ページの『メニューの例のソース』を参照してください。

注: 311 ページの図 103にあるタグは、属性が必要です。それらの属性は、示されていません。それらの属性を指定しなければ、311 ページの図 103 の例はコンパイルされません。これらの必須タグに必要な属性について詳しくは、『付録 A. UIM パネル・グループ定義言語』を参照してください。



RBAHG507-0

図 103. メニュー・パネルの必須 UIM タグ

メニューの例のソース

309 ページの図 102で示したメニューのタグ・ソースを以下にリストします。ソース全体は、ライブラリー QUSRTOOL 内のソース・ファイル QATTUIM にあるメンバー T0011MN2 内にあります。

```

.* -----
.*
.* Beginning of menu source.
.*
.* -----
:PNLGRP   DFTMSGF=t0011msgf2
          SUBMSGF=t0011msgf2.
.*
.* -----
.* Copyright statement appears when the menu is initially displayed.
.* -----
:COPYR.  8
(C) COPYRIGHT YOUR COMPANY NAME HERE, 1992.
.*
.* -----
.* UIM Z-variable to be used as the panel identifier
.* -----
:VAR     NAME=ZMENU.  1
.*
.* -----
.* Define keys for the menu
.* -----
:KEYL    NAME=menukeys 7
          HELP=keyl.
:KEYI    KEY=F1
          HELP=helpf1

```

```

ACTION=HELP.
:KEYI    KEY=F3
        HELP=exit
        ACTION='EXIT SET'
        VARUPD=NO.
F3=Exit  7
:KEYI    KEY=F4
        HELP=prompt
        ACTION=PROMPT.
F4=Prompt
:KEYI    KEY=F9
        HELP=retrieve
        ACTION=RETRIEVE.
F9=Retrieve
:KEYI    KEY=F12
        HELP=cancel
        ACTION='CANCEL SET'
        VARUPD=NO.
F12=Cancel
:KEYI    KEY=F24
        HELP=morekeys
        ACTION=MOREKEYS.
F24=More keys
:KEYI    KEY=ENTER
        HELP=enter
        ACTION=ENTER.
:KEYI    KEY=HELP
        HELP=help
        ACTION=HELP.
:KEYI    KEY=HOME
        HELP=home
        ACTION=HOME.
:KEYI    KEY=PAGEDOWN
        HELP=pagedown
        ACTION=PAGEDOWN.
:KEYI    KEY=PAGEUP
        HELP=pageup
        ACTION=PAGEUP.
:KEYI    KEY=PRINT
        HELP=print
        ACTION=PRINT.
:EKEYL.
.*
.* -----
.* Define Example Menu panel
.* -----
:PANEL   NAME=xmpmenu
        HELP='menu/help'
        KEYL=menukeys 7
        ENTER='MSG CPD9817 QCPFMSG'
        PANELID=ZMENU 1
        TOPSEP=SYSNAM. 3
Example Menu 2
.*
.* -----
.* Define the menu area
.* -----
:MENU    DEPTH='*'
        SCROLL=NO
        BOTSEP=SPACE.
:TOPINST.Select one of the following: 4
.*
.* -----

```



```

.* Specify the action to be taken for each option
.* -----
:MENUI   OPTION=1 5
          ACTION='CMD ?t0011cm2'
          HELP='menu/option1'.
Work with members in a file
.*
:MENUI   OPTION=2
          ACTION='CMD ?t0011cm3'
          HELP='menu/option2'.
Work with record definitions
.*
:EMENU.
.*
.* -----
.* Use a command line and allow commands and option numbers
.* -----
:CMDLINE SIZE=LONG. 6
Selection or command
.*
:EPANEL.
.*
.* -----
.* Define help modules for the menu panel
.* -----
:HELP    NAME=key1. Function Keys - Help
:XH3.Function keys
:EHELP.
.*
:HELP    NAME=helpf1.
:PARML.
:PT.F1=Help
:PD.
Provides additional information about using the display or a
specific field on the display.
:EPARML.
:EHELP.
.*
:HELP    NAME=exit.
:PARML.
:PT.F3=Exit
:PD.
Ends the current task and returns to the display from which the
task was started.
:EPARML.
:EHELP.
.*
:HELP    NAME=prompt.
:PARML.
:PT.F4=Prompt
:PD.
Provides assistance in entering or selecting a command.
:EPARML.
:EHELP.
.*
:HELP    NAME=retrieve.
:PARML.
:PT.F9=Retrieve
:PD.
Displays the last command you entered on the command line and
any parameters you included. Pressing this key once, shows the
last command you ran. Pressing this key twice, shows the
command you ran before that and so on.
:EPARML.
:EHELP.
.*

```

```

:HELP      NAME=cancel.
:PARML.
:PT.F12=Cancel
:PD.
Returns to the previous menu or display.
:EPARML.
:EHELP.
.*
:HELP      NAME=morekeys.
:PARML.
:PT.F24=More keys
:PD.
Shows additional function keys.
:EPARML.
:EHELP.
.*
:HELP      NAME=enter.
:PARML.
:PT.Enter
:PD.
Submits information on the display for processing.
:EPARML.
:EHELP.
.*
:HELP      NAME=help.
:PARML.
:PT.Help
:PD.
Provides additional information about using the display.
:EPARML.
:EHELP.
.*
:HELP      NAME=home.
:PARML.
:PT.Home
:PD.
Goes to the menu that was shown after you signed on the system.
This menu is either the initial menu defined in your user
profile or the menu you requested from the Sign-On display.
:EPARML.
:EHELP.
.*
:HELP      NAME=pagedown.
:PARML.
:PT.Page Down (Roll Up)
:PD.
Moves forward to show additional information for this display.
:EPARML.
:EHELP.
.*
:HELP      NAME=pageup.
:PARML.
:PT.Page Up (Roll Down)
:PD.
Moves backward to show additional information for this display.
:EPARML.
:EHELP.
.*
:HELP      NAME=print.
:PARML.
:PT.Print
:PD.
Prints information currently shown on the display.
:EPARML.
:EHELP.
.*
:HELP      NAME='menu/help'.

```

```

Example Menu - Help
:P.
The Example Menu shows an example of a menu created using the UIM.
:XH3.
How to Use a Menu
:P.
To select a menu option, type the option number and press Enter.
:P.
To run a command, type the command and press Enter. For assistance
in selecting a command, press F4 (Prompt) without typing anything.
For assistance in entering a command, type the command and press F4
(Prompt). To see a previous command you entered, press F9
(Retrieve).
:P.
To go to another menu, use the Go to Menu (GO) command. Type GO
followed by the menu ID, then press the Enter key. For example, to
go to the User Tasks (USER) menu, type GO USER and press the Enter
key. The menu ID is shown in the upper left corner of the menu.
For assistance in entering the GO command, type GO and press F4
(Prompt). If you do not know the entire menu name you can use a
generic name. For example, GO US* will show a list of all menus
that start with US.
:EHELP.
.*
:HELP      NAME='menu/option1'.
Option 1 - Help
:XH3.Option 1. Work with members in a file
:P.
Select this option to work with the members in a file. You will be
prompted for the name of the file.
:EHELP.
.*
:HELP      NAME='menu/option2'.
Option 2 - Help
:XH3.Option 2. Work with record definitions
:P.
Select this option to work with record definitions for a file.
You will be prompted for the name of the file.
:EHELP.
.*
.*
.*
.* -----
.* End of menu source
.* -----
:EPNLGRP.

```

UIM を使用したパネル・グループ・オブジェクトの定義

UIM を使用して、パネル・グループ・オブジェクトを作成できます。パネル・グループ・オブジェクトには、ダイアログ変数、リスト、パネル、およびヘルプ・モジュールの定義を含めることができます。パネル・グループ・オブジェクトを作成するには、以下のように行います。

1. ソース物理ファイルにメンバーを作成します。
2. 原始ステートメント入力ユーティリティー (SEU) などのソース・エディターを使用して、パネル・グループを記述する UIM タグ・ソースを入力します。パネル・グループの例は、ライブラリー QUSRTOOL 内のファイル QATTUIM にあるメンバー T0011PN2 内にあります。この例をコピーして、独自のパネル・グループを作成するためのテンプレートとして使用できます。
3. パネル・グループ作成 (CRTPNLGRP) コマンドを使用して、タグ・ソースを入力にすることにより、パネル・グループ・オブジェクトを作成します。CRTPNLGRP コマンドを使用する例を、以下に示します。

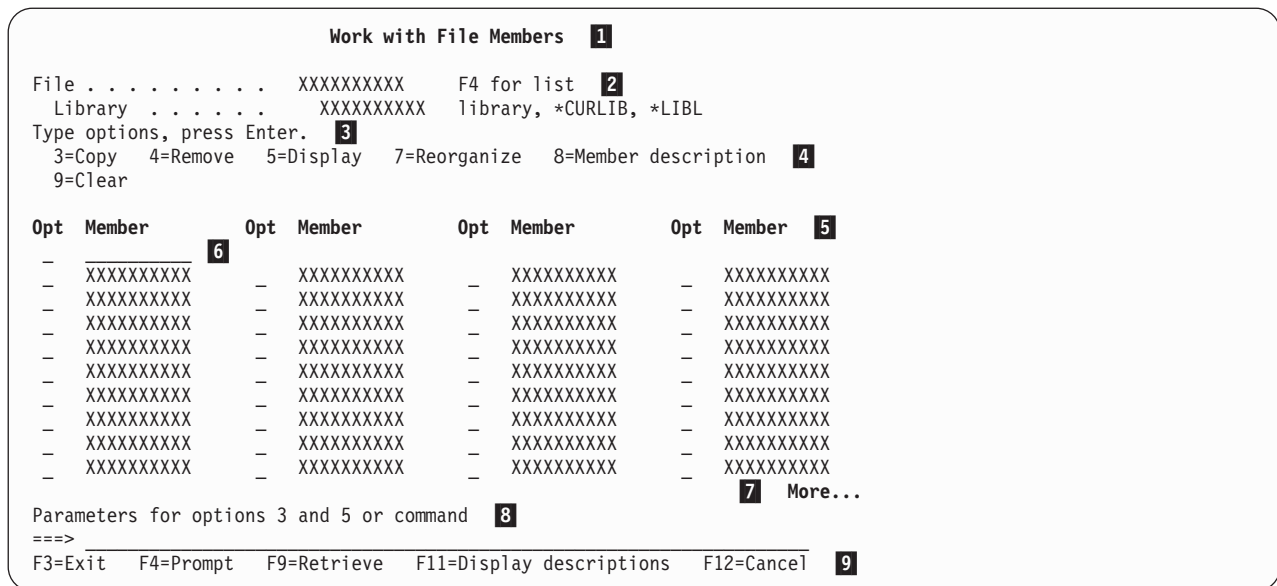


図 105. リストの代替ビューの例

パネルの例に示されている参照番号は、画面には表示されません。これらは図示のためのものであり、319ページの『リスト・パネルの例のソース』に示されている UIM タグ・ソース内にもあります。これらの参照番号は、ソースのどの部分がパネルに表示されるテキストおよび情報を定義しているかを示します。

- 1** パネル・タイトルは、表示パネル (PANEL) タグのピリオドに続くテキストとして定義されます。
- 2** ファイルおよびライブラリー名のテキストは、データ項目 (DATAI) タグのピリオドに続くテキストとして定義されます。ファイル名およびライブラリー名は、ダイアログ変数値です。各ダイアログ変数は、変数定義 (VAR) タグを使用して定義されます。ダイアログ変数の名前は、DATAI タグの VAR 属性に指定されます。各入力フィールドの可能な選択項目テキストは、データ選択項目 (DATAC) タグのピリオドに続くテキストとして定義されます。
2 つの DATAI タグを囲むデータ・グループ (DATAGRP) タグに GRPSEP=QINDENT が指定されているので、ファイル名およびライブラリーは修飾オブジェクト名としてフォーマット設定されます。
ファイル名およびライブラリー名のダイアログ変数は、それぞれ **A** および **B** で VAR タグを使用して定義されます。
- 3** 指示行は、最上部指示 (TOPINST) タグのピリオドに続くテキストとして定義されます。
- 4** 各リスト・アクションは、リスト・アクション (LISTACT) タグを使用して定義されます。等号の付されたオプション番号を含むテキストは、LISTACT タグのピリオドに続くテキストとして定義されます。
- 5** 各列の列見出しは、リスト列 (LISTCOL) タグのピリオドに続くテキストとして定義されます。LISTCOL タグの VAR 属性は、列見出しの下に値が表示されるダイアログ変数の名前を示します。指定されるダイアログ変数名は、VAR タグを使用して定義されている必要があります。ダイアログ変数名は、パネルに示されるリストのリスト定義 (LISTDEF) タグの VARS 属性にも指定されている必要があります。
オプション、メンバー名、メンバー・タイプ、および記述テキストのダイアログ変数は、それぞれ **C**、**D**、**E**、および **F** にある VAR タグを使用して定義されます。

これらの変数の名前は、**G** にある LISTDEF タグの VARS 属性にも指定されます。リスト定義の名前は、**H** にあるリスト域 (LIST) タグの LISTDEF 属性に指定される必要があります。

- 6** 列見出しのすぐ下にある拡張アクション項目は、LIST タグに EXTACT=YES を指定して定義されます。拡張アクション項目に入力フィールドのある列ごとに、LISTCOL タグに EXTACT=YES が指定されている必要があります。

拡張アクション項目により、ユーザーはリスト内の特定の項目までスクロールしなくてもアクションを実行できます。

LISTACT タグで定義される各リスト・オプションは、アクションを実行することが許可されているのは、拡張アクション項目か、拡張アクション項目の下に示されるリスト項目か、またはその両方に対してかを指定します。この指定は、LISTACT タグの ACTFOR 属性を使用して行います。

- 7** LIST タグ上で SCROLL 属性はデフォルトで「YES」になるため、リスト域はスクロール可能として定義されるので、スクロール情報の「続く ...」および「終わり」が表示されます。

- 8** コマンド行は、コマンド行 (CMDLINE) タグを使用して定義されます。コマンド行のプロンプト・テキストは、CMDLINE タグのピリオドに続くテキストとして定義されます。UIM はすべてのコマンド行に矢印 (==>) を自動的に表示します。

コマンド行を使用して、リスト・オプションのパラメーターを入力できます。リスト・オプションを処理する前に、UIM はコマンド行の内容をダイアログ変数に入れます。ダイアログ変数には、以下の使用方法があります。

- LISTACT タグ上で実行するアクションとして指定された CL コマンドに置換される
- LISTACT タグ上で実行するアクションとして指定された出口プログラムによって使用される
- リストに対するアクションを実行するときにアプリケーション・プログラムによって使用される

コマンド行からのパラメーターを入れるために使用されるダイアログ変数は、**I** で VAR タグを使用して定義されています。これは、**J** でリスト域 (LIST) タグの PARMS 属性にも指定されています。UIM はリスト・オプションの処理を開始する前に、コマンド行の内容を PARMS 属性で指定されたダイアログ変数に入れます。

その後 **K** で、そのダイアログ変数は LISTACT タグのコマンド・ストリングで置換変数として使用されます。

- 9** 各機能キーは、キー・リスト (KEYL) タグを使用して定義されます。機能キー名を含むテキストは、KEYI タグのピリオドに続くテキストとして定義されます。

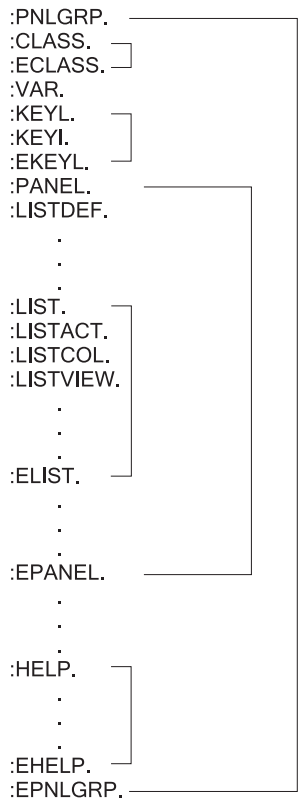
すべての機能キーは、KEYI タグをキー・リスト (KEYL) および EKEYL タグの間に配置することにより定義されます。KEYL タグの NAME 属性に指定された名前は、PANEL タグの KEYL 属性にも指定されます。

Enter などのいくつかの機能キーには、KEYI タグに指定されたテキストがありません。この場合、画面にはテキストが表示されません。ただし、すべての機能キーは KEYI タグを使用して定義して、各キーにどの操作が割り当てられているか、および機能キーのヘルプを表示する際にどのヘルプ・モジュールを使用するかを UIM が認識する必要があります。

リスト・パネルの必須タグ

319 ページの図 106は、UIM リスト・パネルを作成するために必要なタグを示しています。必須タグおよびオプション・タグの例については、319 ページの『リスト・パネルの例のソース』を参照してください。

注: 図 106にあるタグは、属性が必要です。それらの属性は、示されていません。それらの属性を指定しなければ、図 106の例はコンパイルされません。これらの必須タグに必要な属性について詳しくは、『付録 A. UIM パネル・グループ定義言語』を参照してください。



RBAHG508-0

図 106. リスト・パネルの必須タグ

リスト・パネルの例のソース

これは、ライブラリー QUSRTOOL 内のソース・ファイル QATTUIM にあるメンバー T0011PN2 の部分リストです。

```

.* -----
.*
.* Beginning of panel group source.
.*
.* -----
:PNLGRP   DFTMSGF=t0011msgf2
          SUBMSGF=t0011msgf2.
.*
.* The import tag specifies that all help is to be found
.* in panel group T0011HL2 searching the library list.
:IMPORT   NAME='*'
          PNLGRP=t0011h12.
.*
.* -----
.* Define all variable classes
.* -----
.* -----

```

```

.* Option
.* Note: Need WIDTH=1 to preserve column alignment on confirmation panel.
:CLASS    NAME=optc1
          BASETYPE='ACTION'
          WIDTH=1.
:ECLASS.
.* -----
.* Object name
:CLASS    NAME=namec1
          BASETYPE='OBJNAME 10'.
:ECLASS.
.* -----
.* Library name
:CLASS    NAME=libc1
          BASETYPE='OBJNAME 10'.
:TL.
:TI      VALUE='"*LIBL"*.*LIBL
:TI      VALUE='"*CURLIB"*.*CURLIB
:ETL.
:ECLASS.
.* -----
.* File attribute
:CLASS    NAME=attrc1
          BASETYPE='CHAR 10'
          CASE=UPPER.
:ECLASS.
.* -----
.* Descriptive text
:CLASS    NAME=textc1
          BASETYPE='IGC 50'
          SUBST=QUOTED.
:ECLASS.
.* -----
.* Source type
:CLASS    NAME=srctypc1
          BASETYPE='CHAR 10'.
:ECLASS.
:
:
Additional CLASS tags in member T0011PN2 are not shown here
:
.* -----
.* Command line parameters
:CLASS    NAME=parmc1
          BASETYPE='CHAR 255'.
:ECLASS.
.* -----
.* Exit program specification for CALL dialog command
:CLASS    NAME=exitc1
          BASETYPE='CHAR 20'.
:ECLASS.
.* -----
.* View number
:CLASS    NAME=vwnumc1
          BASETYPE='BIN 15'.
:ECLASS.
.* -----
.* Classes for pad variables in variable record definitions.
:CLASS    NAME=pad1c1
          BASETYPE='CHAR 1'.
:ECLASS.
:CLASS    NAME=pad2c1

```



```

        BASETYPE='CHAR 2'.
:ECLASS.
:CLASS   NAME=pad10c1
        BASETYPE='CHAR 10'.
:ECLASS.
:CLASS   NAME=pad13c1
        BASETYPE='CHAR 13'.
:ECLASS.
:CLASS   NAME=pad48c1
        BASETYPE='CHAR 48'.
:ECLASS.
:CLASS   NAME=pad50c1
        BASETYPE='CHAR 50'.
:ECLASS.
.*
.* -----
.* Define all dialog variables
.* -----
.*
.* -----
.* Variables for file and library
.* -----
.*
.* File name
:VAR     NAME=file A
        CLASS=namecl.
.* -----
.* Library name
:VAR     NAME=lib B
        CLASS=libcl.
.* -----
.* attributes
:VAR     NAME=fattr
        CLASS=attrcl.
.* -----
.*
.* -----
.* Variables for list of members
.* -----
.*
.* option for list of members
:VAR     NAME=mopt C
        CLASS=optcl.
.* -----
.* Object name
:VAR     NAME=mbr D
        CLASS=namecl.
.* -----
.* member type
:VAR     NAME=mtype E
        CLASS=attrcl.
.* -----
.* Descriptive text
:VAR     NAME=mtext F
        CLASS=textcl.
        :
        Additional VAR tags in member T0011PN2 are not shown here
        :
.*
.* -----
.* Variable for command line parameters
.* -----
.*
.* Command line parameters
:VAR     NAME=parms I
        CLASS=parmc1.

```

```

.*
.* -----
.* Variables for specifying CALL/exit programs
.* -----
.* -----
.* Program to call for all UIM exits
:VAR      NAME=exitpgm
          CLASS=exitcl.
.*
.* -----
.* Variables for controlling list views
.* -----
.* -----
.* View number for list of members
:VAR      NAME=mbrview
          CLASS=vwnumcl.
          .
          .
          .
Additional VAR tags in member T0011PN2 are not shown here
          .
          .
          .
.*
.* -----
.* Variables for padding in variable record definitions.
.* Padding is needed in variable records so the layout
.* of the record matches a list format returned from
.* an API. The pad variables are used as placeholders for
.* variables not used in the API format or for reserved space
.* in the API format.
.* -----
:VAR      NAME=pad1
          CLASS=pad1cl.
:VAR      NAME=pad2
          CLASS=pad2cl.
:VAR      NAME=pad10
          CLASS=pad10cl.
:VAR      NAME=pad13
          CLASS=pad13cl.
:VAR      NAME=pad48
          CLASS=pad48cl.
:VAR      NAME=pad50
          CLASS=pad50cl.
.*
.* -----
.* Define a variable record for file, library and file attribute
.* -----
:VARRCD   NAME=filelib
          VARS='file lib fattr'
          NOGET='fattr'
          .
.*
.* -----
.* Define a variable record for exit program
.* -----
:VARRCD   NAME=exitprog
          VARS='exitpgm'
          .
.*
.* -----
.* Define a variable record for list of members.
.* The layout of this record is designed to match the
.* List Database File Members API (QUSLMBR) format name MBRL0200.
.* -----
:VARRCD   NAME=mbrl0200
          VARS='mbr mtype pad13 pad13 mtext'
          NOPUT='pad13'
          NOGET='mtype pad13 mtext'
          .

```

```

:
Additional VARRCD tags in member T0011PN2 are not shown here
:
.*
.*
.* -----
.* Define a list of members
.* -----
G
:LISTDEF NAME=mbrlist
        VARS='mopt mbr mtype mtext'
        MSGID=USR0101.
:
Additional LISTDEF tags in member T0011PN2 are not shown here
:
.*
.* -----
.* Define all conditions
.* -----
.* -----
.* Condition for physical files
:COND   NAME=pf
        EXPR='fattr="PF      "'.
.* -----
.* Conditions for views of members list
:COND   NAME=mbrview1
        EXPR='mbrview=0'.
:COND   NAME=mbrview2
        EXPR='mbrview=1'.
:
Additional COND tags in member T0011PN2 are not shown here
:
.*
.* -----
.* Define truth table to specify that mbrview1 and mbrview2
.* are mutually exclusive conditions
.* This will cause UIM to reserve only one line of function
.* keys on the work with members panel
.* -----
:TT      NAME=mbrtt
        CONDS='mbrview1 mbrview2'.
:TTROW   VALUES='  1      0  '.
:TTROW   VALUES='  0      1  '.
:ETT.
:
Additional TT tags in member T0011PN2 are not shown here
:
All MBAR tags in member T0011PN2 are not shown here
:
.*
.*
.* -----
.* Define keys for work with members panel
.* -----
:KEYL    NAME=mbrkeys 9
        HELP=key1.
:KEYI    KEY=F1
        HELP=help1
        ACTION=HELP.
:KEYI    KEY=F3
        HELP=exit
        ACTION='EXIT SET'
        VARUPD=NO.
F3=Exit

```

```

:KEYI      KEY=F4
           HELP=prompt
           ACTION=PROMPT
           PRIORITY=30.
F4=Prompt
:KEYI      KEY=F9
           HELP=retrieve
           ACTION=RETRIEVE
           PRIORITY=35.
F9=Retrieve
:KEYI      KEY=F11
           HELP=mbrviewname
           ACTION=CHGVIEW
           PRIORITY=25
           COND=mbrview1.
F11=Display names only
:KEYI      KEY=F11
           HELP=mbrviewdesc
           ACTION=CHGVIEW
           PRIORITY=25
           COND=mbrview2.
F11=Display descriptions
:KEYI      KEY=F12
           HELP=cancel
           ACTION='CANCEL SET'
           VARUPD=NO.
F12=Cancel
:KEYI      KEY=F24
           HELP=morekeys
           ACTION=MOREKEYS.
F24=More keys
:KEYI      KEY=ENTER
           HELP=enter
           ACTION=ENTER.
:KEYI      KEY=HELP
           HELP=help
           ACTION=HELP.
:KEYI      KEY=PAGEDOWN
           HELP=pagedown
           ACTION=PAGEDOWN.
:KEYI      KEY=PAGEUP
           HELP=pageup
           ACTION=PAGEUP.
:KEYI      KEY=PRINT
           HELP=print
           ACTION=PRINT.
:EKEYL.
.
.
Additional KEYL tags in member T0011PN2 are not shown here
.
.*
.*
.* -----
.* Define Work with Members panel
.* -----
:PANEL    NAME=wrkibr
          HELP='wrkibr/'
          KEYL=mbrkeys  9
          TT=mbrtt
          ENTER='RETURN 500'
          TOPSEP=SPACE.
Work with File Members  1
.*
.* -----
.* Define a data presentation area to display the
.* library/file name whose members are listed.

```

```

.* -----
:DATA    DEPTH=3
         SCROLL=NO
         LAYOUT=1
         BOTSEP=SPACE
         COMPACT
         .
.* -----
.* Divide the layout width into two columns.
.* The first column is for the prompt text with leader dots.
.* The second column is for the variable values.
:DATACOL WIDTH=22.
:DATACOL WIDTH=12.
:DATACOL WIDTH='*'.
.* -----
.* Display qualified file name
:DATAGRP GRPSEP=QINDENT
         HELP='wrkmbbr/filelib'
         COMPACT
         .
:DATAI   VAR=file 2
         USAGE=INOUT
         PROMPT='CALL exitpgm'
         .
File
:DATAC.F4 for list
:DATAI   VAR=lib
         USAGE=INOUT
         .
Library
:DATAC.library, *CURLIB, *LIBL
:EDATAGRP.
.*
:EDATA.
.*
.* -----
.* Define the list area
.* -----
:LIST    DEPTH='*' 7
         LISTDEF=mbrlist H
         ACTOR=UIM
         EXTRACT=YES 6
         MAXHEAD=4
         MAXACTL=3
         VIEW=mbrview
         PARS=parms J
         BOTSEP=SPACE.
:TOPINST.Type options, press Enter. 3
.*
.* -----
.* Specify the action to be taken for each option
.* -----
:LISTACT OPTION=3
         ACTFOR=BOTH 6
         NOCMD=PROMPT
         NOEXT=PROMPT
         HELP='wrkmbbr/cpyf'
         ENTER='CMD CPYF ?*FROMFILE(&lib/&file)'
         ENTER=' ?*FROMMBR(&mbr) &parms' K
         PROMPT='CMD ?CPYF ?*FROMFILE(&lib/&file)'
         PROMPT=' ?*FROMMBR(&mbr) &parms' K
         EXTENTER='CMD ?CPYF ?*FROMFILE(&lib/&file)'
         EXTENTER=' ??FROMMBR(&mbr) &parms' K
         EXTPROMPT='CMD ?CPYF ?*FROMFILE(&lib/&file)'
         EXTPROMPT=' ??FROMMBR(&mbr) &parms'. K
3=Copy 4
.*

```

```

:LISTACT OPTION=4
ACTFOR=LISTE
HELP='wrkmbbr/rmvm'
ENTER='CMD RMVM FILE(&lib/&file) MBR(&mbr)'
PROMPT='CMD ?RMVM ?*FILE(&lib/&file) ?*MBR(&mbr)'
USREXIT='CALL exitpgm'.

4=Remove
.*
:LISTACT OPTION=5
COND=pf
ACTFOR=BOTH
NOEXT=PROMPT
HELP='wrkmbbr/dsppfm'
ENTER='CMD DSPPFM FILE(&lib/&file) MBR(&mbr) &parms'
PROMPT='CMD DSPPFM ?*FILE(&lib/&file) ?*MBR(&mbr) &parms'
EXTPROMPT='CMD DSPPFM ?*FILE(&lib/&file) ??MBR(&mbr) &parms'.

5=Display
.*
:LISTACT OPTION=7
COND=pf
ACTFOR=BOTH
HELP='wrkmbbr/rgzm'
ENTER='CMD RGZPFM FILE(&lib/&file) MBR(&mbr)'
PROMPT='CMD ?RGZPFM ?*FILE(&lib/&file) ?*MBR(&mbr)'.

7=Reorganize
.*
:LISTACT OPTION=8
ACTFOR=BOTH
HELP='wrkmbbr/dspfd'
ENTER='CALL exitpgm'
PROMPT='CALL exitpgm'.

8=Member description
.*
:LISTACT OPTION=9
COND=pf
ACTFOR=LISTE
HELP='wrkmbbr/clrm'
ENTER='CMD CLRPFM FILE(&lib/&file) MBR(&mbr)'
PROMPT='CMD ?CLRPFM ?*FILE(&lib/&file) ?*MBR(&mbr)'.

9=Clear
.*
.*
.* -----
.* Define the columns and headings to display
.* -----
:LISTCOL VAR=mopt
USAGE=INOUT
EXTRACT=YES 6
HELP='wrkmbbr/option'
MAXWIDTH=6.

Opt 5
:LISTCOL VAR=mbr
USAGE=OUT
EXTRACT=YES
HELP='wrkmbbr/mbr'
MAXWIDTH=10.

Member
:LISTCOL VAR=mtype
USAGE=OUT
HELP='wrkmbbr/type'
MAXWIDTH=10.

Type
:LISTCOL VAR=mtxt
USAGE=OUT
HELP='wrkmbbr/text'
MAXWIDTH='*'.

Text

```

```

.*
.* -----
.* Define multiple views for F11 to toggle between
.* -----
:LISTVIEW COLS='mopt mbr mtype mtext'.
:LISTVIEW COLS='mopt mbr' layout=4.
.*
:ELIST.
.*
.* -----
.* Use a command line and allow parameters to be given
.* -----
:CMDLINE SIZE=SHORT. 8
Parameters for options 3 and 5 or command
.*
:EPANEL.
:
Additional PANEL tags in member T0011PN2 are not shown here
:
.*
.* -----
.* End of panel group source
.* -----
:EPNLGRP.

```

リスト・パネルのアプリケーション・プログラミング

316 ページの図 104 に示されたリスト・パネルを表示するアプリケーション・プログラムの例は、ライブラリー QUSRTOOL 内のソース・ファイル QATTSYSC にあるメンバー T0011CP2 に含まれています。これは適切な UIM アプリケーション・プログラミング・インターフェース (API) を呼び出してパネルを表示する、ILE C/C++ プログラムです。

UIM API を使用する RPG アプリケーションの一般的な例は、QUSRTOOL にあります。ライブラリー QUSRTOOL 内のソース・ファイル QATTINFO にあるメンバー T0011INF を参照してください。例の使用方法について詳しくは、455 ページの『第 21 章 IBM i5/OS スタイルの画面の設計』を参照してください。

リスト・パネルの例を表示するプログラムをいずれかの言語で記述するには、そのプログラムが以下を行う必要があります。

1. Open Display Application (QUIOPNDA) API を呼び出して、パネル・グループを開きます。パネル・グループは、パネル・グループ作成 (CRTPNLGRP) コマンドを使用して作成済みでなければなりません。
2. 以下のダイアログ変数の値を含むバッファをセットアップします。

FILE ファイルの名前を示す CHAR 10 変数。

LIB ファイルが存在するライブラリーの名前を示す CHAR 10 変数。

FATTR

ファイルのファイル属性を示す CHAR 10 変数。この変数は、物理ファイルだけに許可されるリスト・オプションに対して条件設定するために使用されます。

3. Put Dialog Variable (QUIPUTV) API を呼び出して、変数レコード FILELIB および直前のステップで初期化されたバッファを使用することにより、ダイアログ変数の内容を変更します。
4. 以下のダイアログ変数の値を含むバッファをセットアップします。

EXITPGM

呼び出されるプログラムを UIM 出口プログラムとして識別する CHAR 20 変数。拡張プログ

ラム・モデル (EPM) 言語では、EXITPGM ダイアログ変数の定義に使用される CLASS タグのこの BASETYPE 属性は、CHAR 130 ダイアログ変数に変更する必要があります。

このダイアログ変数を設定する方法については、699 ページの『付録 B. UIM ダイアログ・コマンド』で CALL ダイアログ・コマンドについての説明を参照してください。

5. Put Dialog Variable (QUIPUTV) API を呼び出して、変数レコード EXITPROG および直前のステップで初期化されたバッファーを使用することにより、ダイアログ変数の内容を変更します。
6. Create User Space (QUSCRTUS) API を使用して、ファイル内のメンバーのリストを受け取るユーザー・スペースを作成します。ファイル内のメンバーは、List Database File Members (QUSLMBR) API を使用して検索します。アプリケーション・プログラミング言語がポインターをサポートする場合、Retrieve Pointer to User Space (QUSPTRUS) API を使用してユーザー・スペースの内容に対するポインターを取得します。これにより、アプリケーション・プログラムはデータを直接取り扱うことが可能になります。あるいは、Retrieve User Space (QUSRTVUS) API を使用して、ユーザー・スペースの内容を取得します。これらの API について詳しくは、iSeries Information Center の『API』のトピックを参照してください。
7. 表示するファイル内のすべてのメンバーに対して、以下を行います。ファイル内のメンバーのリストは、List Database File Members (QUSLMBR) API を使用して検索できます。この API について詳しくは、『API』のトピックで「List Database File Members (QUSLMBR) API」を参照してください。
 - a. 以下のダイアログ変数の値を含むバッファーをセットアップします。

MBR メンバー名を示す CHAR 10 変数。

MTYPE

メンバー・タイプを示す CHAR 10 変数。

PAD13

バッファー内の CHAR 13 予約済みスペース。

PAD13

バッファー内の CHAR 13 予約済みスペース。

MTEXT

メンバーについて説明するテキストを示す CHAR 50 変数。

注: このバッファーのレイアウトは、MBRL0200 フォーマットを使用する QUSLMBR API によって戻される、ユーザー・スペース内の項目のレイアウトと一致するように設計されています。そのため、バッファーをセットアップする代わりに、アプリケーション・プログラムはユーザー・スペースに存在するままのバッファーを渡すことができます。

2 つの PAD13 変数を使用して、QUSLMBR API によって戻される MBRL0200 フォーマットの 2 つの変数を見捨てるように UIM に知らせます。これらの変数は、作成日時、およびソースの最終変更日時です。

- b. Add List Entry (QUIADDLE) API を呼び出して、変数レコード MBRL0200 および直前のステップで初期化されたバッファーを使用することにより、リスト項目を MBRLIST という名前のリストに追加します。
8. Display Panel (QUIDSPP) API を呼び出して、表示パネル WRKMBR を表示します。以下の 1 つが生じたとき、UIM は制御をアプリケーションに戻します。
 - ユーザーがコマンド行にリスト・オプションまたはコマンドを入力しないで Enter キーを押したとき。機能要求パラメーターとして QUIDSPP API に渡されるプログラム変数は、500 に設定されます。これは、WRKMBR パネルを定義する PANEL タグで、ENTER='RETURN 500' が指定されているためです。

- ユーザーが F12 (取り消し) キーを押したとき。機能要求パラメーターとして QUIDSPP API に渡されるプログラム変数は、-8 に設定されます。これは CANCEL ダイアログ・コマンドに定義された値です。
- ユーザーが F3 (終了) キーを押したとき。機能要求パラメーターとして QUIDSPP API に渡されるプログラム変数は、-4 に設定されます。これは CANCEL ダイアログ・コマンドに定義された値です。

9. Close Application (QUICLOA) API を呼び出して、UIM アプリケーションを閉じます。これにより、UIM アプリケーションによって使用されていたシステム・リソースが解放されます。

確認リスト・パネルの作成

破壊的な操作を実行する処置リスト・パネル上のオプションは、その操作を実行する前にユーザーが要求の確認または取り消しを行うことを可能にする確認パネルを使用する必要があります。確認パネルを備える必要のある操作の例としては、316 ページの図 104 のパネルの例に示された「4=除去」および「9=消去」などがあります。

確認パネルの例は、図 107 に示されています。この確認パネルは、316 ページの図 104 に示されたパネルでユーザーがオプション 4 を入力したときに表示されます。

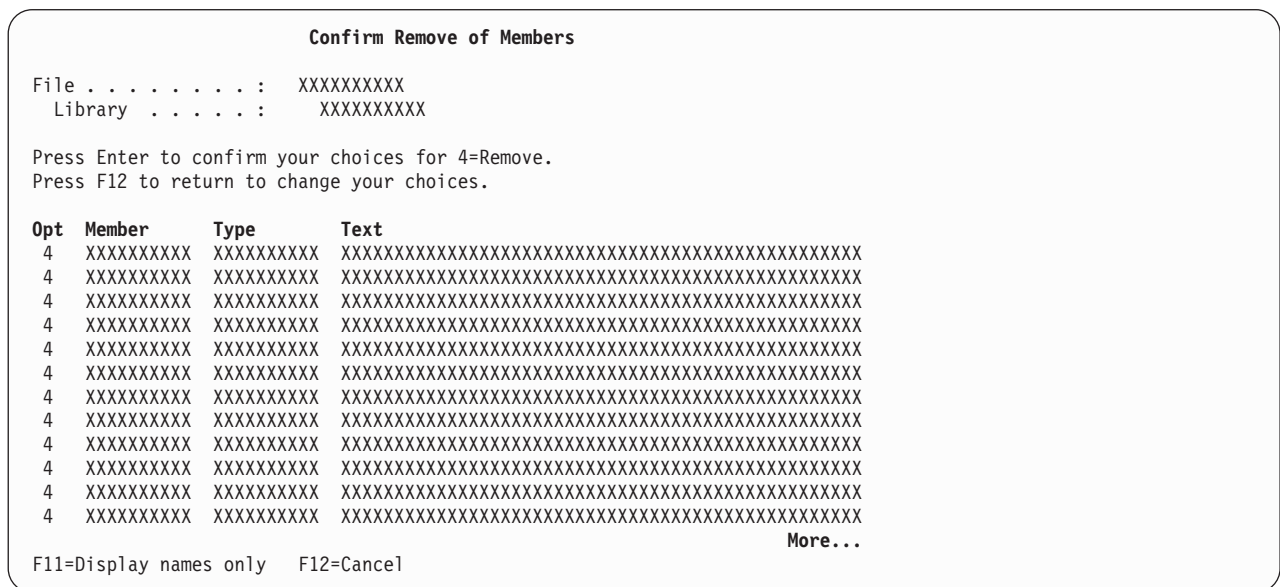


図 107. 確認リスト・パネルの例

確認パネルの外観は処置リスト・パネルと似ていますが、以下の例外があります。

- 指示はユーザーにオプションを確認するかまたは取り消すように伝えます。
- 表示されるリストは出力専用です。ユーザーがいずれかのオプション番号を変更することはできません。
- 確認パネルでは、F3 (終了) キーは許可されません。
- 確認パネルにはコマンド行がありません。

ユーザーが別のビューに変更するための F11 キーを押すと、処置リスト・パネルが再表示されるときにそのビューが変更されます。これは、処置リスト・パネルと確認パネルの両方で、LIST タグの VIEW 属性に同じダイアログ変数が指定されているためです。

確認リスト・パネルに必要なタグ

確認リスト・パネルに必要なタグは、319 ページの図 106 で説明されているリスト・パネルに必要なタグと基本的に同じです。ただし、LISTACT タグは必要ありません。

確認パネルの例のソース

この確認パネルを定義するには、319 ページの『リスト・パネルの例のソース』に示されたソースに追加のソースを追加する必要があります。以下は、ライブラリー QUSRTOOL 内のソース・ファイル QATTUIM にあるメンバー T0011PN2 からの抜粋です。最初に、新規のリスト定義を作成する必要があります。このリスト定義には、確認の対象となるリスト項目のコピーが含まれます。確認リストには、メンバーのリストに含まれるものと同じダイアログ変数が含まれています。以下のリスト定義 (LISTDEF) タグは、**G** にある LISTDEF タグの後に配置する必要があります。

```
. *
. * -----
. * Define a list of members for confirmation panels
. * -----
:LISTDEF  NAME=mbrconf
          VARS='mopt mbr mtype mtext'.
```

次に、確認パネルの機能キーを定義する必要があります。確認パネルでは F3 (終了) を許可しないでください。以下のキー・リスト (KEYL) タグは、319 ページの『リスト・パネルの例のソース』に示されている KEYL タグの後に配置する必要があります。

```
. *
. * -----
. * Define keys for confirm remove of members
. * -----
:KEYL    NAME=mconfkeys
          HELP=keyl.
:KEYI    KEY=F1
          HELP=helpf1
          ACTION=HELP.
:KEYI    KEY=F11
          HELP=altview
          ACTION=CHGVIEW
          COND=mbrview1.
F11=Display names only
:KEYI    KEY=F11
          HELP=altview
          ACTION=CHGVIEW
          COND=mbrview2.
F11=Display descriptions
:KEYI    KEY=F12
          HELP=cancel
          ACTION='CANCEL SET'
          VARUPD=NO.
F12=Cancel
:KEYI    KEY=F24
          HELP=morekeys
          ACTION=MOREKEYS.
F24=More keys
:KEYI    KEY=ENTER
          HELP=enter
          ACTION=ENTER.
:KEYI    KEY=HELP
          HELP=help
```

```

:KEYI      ACTION=HELP.
           KEY=PAGEDOWN
           HELP=pagedown
           ACTION=PAGEDOWN.
:KEYI      KEY=PAGEUP
           HELP=pageup
           ACTION=PAGEUP.
:KEYI      KEY=PRINT
           HELP=print
           ACTION=PRINT.
:EKEYL.

```

確認パネルを定義する必要があります。以下のソースは、319 ページの『リスト・パネルの例のソース』に示されている EPANEL タグの後に配置する必要があります。

```

.*
.* -----
.* Define panel for confirm remove of members from WRKMBR panel
.* -----
:PANEL      NAME=confrvm
           HELP='confrvm/'
           KEYL=mconfkeys
           TT=mbrtt
           ENTER='RETURN 100'
           TOPSEP=space.
Confirm Remove of Members
.*
.* -----
.* Define a data presentation area to display the
.* library/file name whose members are listed.
.* -----
:DATA      DEPTH=3
           SCROLL=NO
           LAYOUT=1
           BOTSEP=SPACE
           COMPACT
           .
.* -----
.* Divide the layout width into two columns.
.* The first column is for the prompt text with leader dots.
.* The second column is for the variable values.
:DATACOL   WIDTH=22.
:DATACOL   WIDTH='*'.
.* -----
.* Display qualified file name
:DATAGRP   GRPSEP=QINDENT
           HELP='wrkmbbr/filelib'
           COMPACT
           .
:DATAI     VAR=file
           USAGE=OUT
           .
File
:DATAI     VAR=lib
           USAGE=OUT
           .
Library
:EDATAGRP.
.*
:EDATA.
.*
.* -----
.* Define the list area
.* -----

```

```

:LIST      DEPTH='*'
           MAXHEAD=4
           LISTDEF=mbrconf
           VIEW=mbrview.

:TOPINST.Press Enter to confirm your choices for 4=Remove.
:TOPINST.Press F12 to return to change your choices.
.*
.* -----
.* Define the columns of the list
.* -----
:LISTCOL  VAR=mopt
           USAGE=OUT
           HELP='confrvmv/option'
           MAXWIDTH=6.

Opt
:LISTCOL  VAR=mbr
           USAGE=OUT
           HELP='wrkmbrr/mbr'
           MAXWIDTH=10.

Member
:LISTCOL  VAR=mtype
           USAGE=OUT
           HELP='wrkmbrr/type'
           MAXWIDTH=10.

Type
:LISTCOL  VAR=mtext
           USAGE=OUT
           HELP='wrkmbrr/text'
           MAXWIDTH='*'.

Text
.*
.* -----
.* Define multiple views for F11 to toggle between
.* -----
:LISTVIEW COLS='mopt mbr mtype mtext'.
:LISTVIEW COLS='mopt mbr' layout=4.
.*
:ELIST.
:EPANEL.

```

自動確認処理

UIM は、確認処理の自動実行をサポートします。このサポートは、処置リストを定義する LIST タグに ACTOR=UIM を指定することにより、UIM がリスト・オプションの処理を制御中であるときに使用可能となります。

UIM がリスト・オプションの確認処理を実行するには、確認対象のオプションを定義するリスト・アクション (LISTACT) タグの CONFIRM 属性で確認パネルの名前を指定します。以下の UIM ソースは、CONFIRM 属性によって確認パネルの名前が指定された、オプション 4 に対する LISTACT タグを示しています。

```

.*
:LISTACT  OPTION=4
           ACTFOR=LISTE
           HELP='wrkmbrr/rmvm'
           CONFIRM=confrvmv
           ENTER='CMD RMVM FILE(&lib/&file) MBR(&mbr)'
           PROMPT='CMD ?RMVM ?*FILE(&lib/&file) ?*MBR(&mbr)'
           USREXIT='CALL exitpgm'.

4=Remove
.*

```

CONFIRM 属性を使用して確認パネルを定義する方法については、614 ページの『LISTACT (リスト・アクション)』を参照してください。

UIM がリスト・オプションを処理して CONFIRM 属性の指定されたオプションを見つけると、UIM は以下を行います。

1. LIST タグの LISTDEF 属性に指定されたリストを確認パネルから削除します。
2. 処置リスト内でオプション番号が確認対象のオプション番号と等しいすべての項目を検索して、それらの項目を確認リストにコピーします。
3. 確認パネルを表示します。
4. ユーザーが Enter キーを押した場合、そのオプション番号がユーザーによって確認されたことを記憶します。確認済みのオプションは、それらが処置リストに出現する順序で処理されます。ユーザーが処置リスト内の追加項目のオプション番号を入力するまで、このオプション番号についてはそれ以上の確認処理は行われません。
5. ユーザーが F12 (取り消し) キーを押すとリスト・オプションの処理が停止して、オプション番号が確認されていない最初の項目を示す処置リスト・パネルが再表示されます。

確認処理のアプリケーション・プログラミング

ACTOR=UIM が処置リスト・パネルの LIST タグに指定されているときは、UIM がすべての確認処理を実行するので、アプリケーション・プログラミングは必要ありません。

ACTOR=CALLER が処置リスト・パネルの LIST タグに指定されているときは、リスト・オプションを処理するアプリケーション・プログラムが、332 ページの『自動確認処理』に示されている UIM 処理に似た確認処理を実行する必要があります。

データ表示パネルの作成

データ表示パネルは、リスト・パネル上のオプションに関連して、ユーザー・データを表示するため、またはユーザー・データの入力を許可するために使用されます。データの入出力がスクロール可能で頻度とボリュームが小さい場合、UIM を使用すると一貫性のあるデータ表示パネルの作成が簡単になります。

以下は「Work with File Members (ファイル・メンバー処理)」画面から「Display Member Description (メンバー記述の表示)」画面を作成するコードを示しています。

334 ページの図 108にあるパネルは、データ表示パネルの例を示しています。このパネルは、316 ページの図 104に示されたパネルでオプション 5 を使用する結果として表示されます。

このパネルには、2 つのデータ表示域が含まれます。最初の領域は、2 つのレイアウト列のある垂直レイアウトを使用します。この領域には、ファイル、ライブラリー、およびメンバー名が示されます。2 番目の領域も垂直レイアウトを使用しますが、レイアウト列は 1 つだけです。この領域には、最初の領域で識別されたメンバーについての詳細情報が示されます。データ表示パネルのヘルプを作成する方法については詳しくは、428 ページの『データ域のヘルプ』を参照してください。

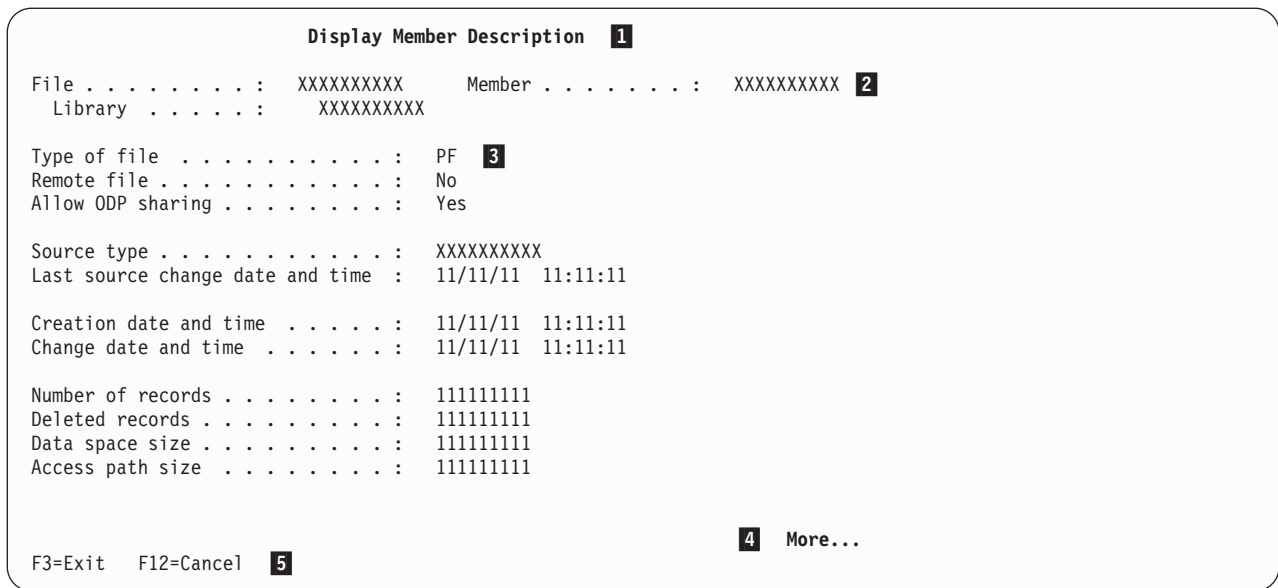


図 108. データ表示パネルの例

2 番目の領域は、画面に一度に収まる以上の項目があるため、スクロール可能域として定義されます。ユーザーが Page Up (前ページ) キーまたは Page Down (次ページ) キーを押すと、UIM はスクロールを自動的に処理します。Page Down 操作が実行されると、2 番目のデータ表示域の内容は次に表示される項目のセットに置き換わります。パネル内の結果が、335 ページの図 109 に示されています。

2 番目の領域がスクロールされる時も、最初の領域の内容は変化しないことに注意してください。スクロール操作は、カーソルの位置に基づいて、パネル内の 1 つの領域だけに適用されます。さらに、最初の領域はスクロール可能として定義されていないので、カーソル位置が領域内にある場合でも、スクロール操作はその領域に適用されません。

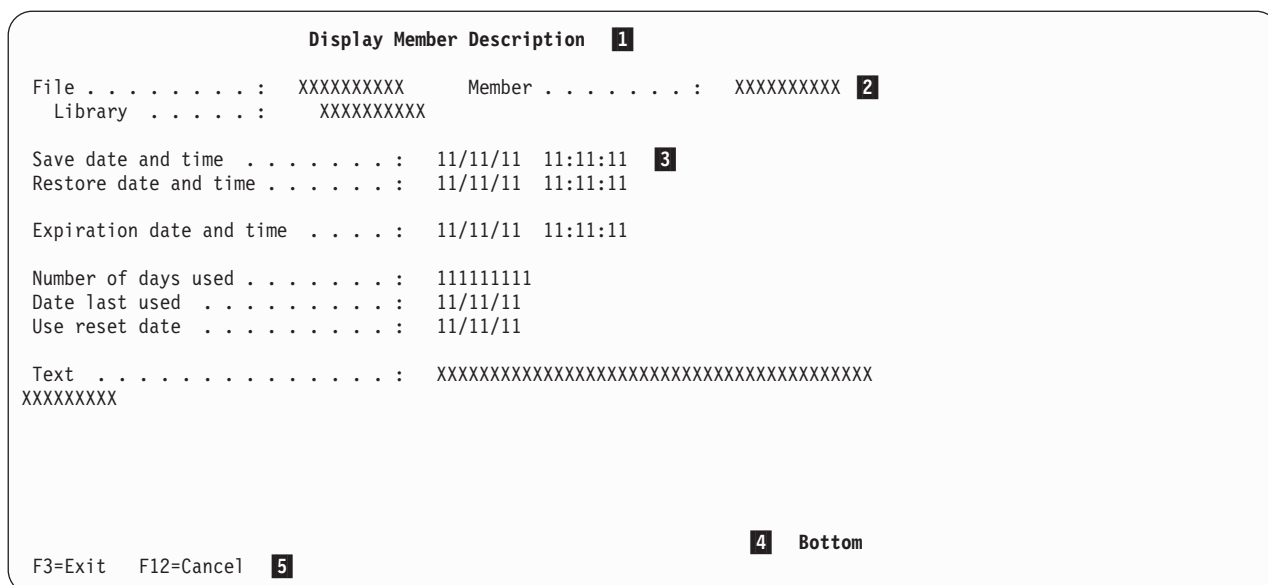


図 109. スクロール後のデータ表示パネルの例

パネルの例に示されている参照番号は、画面には表示されません。これらは図示のためのものであり、337 ページの『データ表示パネルの例のソース』に示されている UIM タグ・ソース内にもあります。これらの参照番号は、ソースのどの部分がパネルに表示されるテキストおよび情報を定義しているかを示します。

1 パネル・タイトルは、表示パネル (PANEL) タグのピリオドに続くテキストとして定義されます。

2 最初のデータ域で、ファイル、ライブラリー、およびメンバー名のテキストは、データ項目 (DATAI) タグのピリオドに続くテキストとして定義されます。ファイル、ライブラリー、およびメンバー名は、ダイアログ変数値です。各ダイアログ変数は、変数定義 (VAR) タグを使用して定義されます。ダイアログ変数の名前は、DATAI タグの VAR 属性に指定されます。

2 つの DATAI タグを囲むデータ・グループ (DATAGRP) タグに GRPSEP=QINDENT が指定されているので、ファイル名およびライブラリーは修飾オブジェクト名としてフォーマット設定されます。

ファイル名、ライブラリー名、およびメンバー名のダイアログ変数は、それぞれ **A**、**B**、および **C** で VAR タグを使用して定義されます。

このデータ表示域は、領域を定義する DATA タグに LAYOUT=2 を指定して定義されます。これは **D** で示されています。

3 2 番目のデータ域で、各項目のテキストは DATAI タグのピリオドに続くテキストとして定義されます。各項目の値は、ダイアログ変数値です。それぞれのダイアログ変数は、VAR タグで定義されます。ダイアログ変数の名前は、DATAI タグの VAR 属性に指定されます。

領域内のいくつかの項目は、日付および時刻の値を示します。これを行うには、DATAI タグの VAR 属性に日付値のダイアログ変数を指定して、データ項目エクステンダー (DATAIX) タグの VAR 属性に時刻値のダイアログ変数を指定します。

日付および時刻のダイアログ変数は、BASETYPE=DATE および BASETYPE=TIME を、変数の定義に使用されるクラス定義 (CLASS) タグに指定して定義します。これは **E** に示されています。

UIM は日付および時刻の変数を、ジョブの日付形式、区切り記号属性、および時刻区切り記号属性に従ってフォーマット設定します。

4 リスト域はスクロール可能として定義されているので、スクロール情報の「続く ...」および「終わり」が表示されます。 SCROLL=YES が DATA タグに指定されます。

5 各機能キーは、キー・リスト項目 (KEYI) タグを使用して定義されます。機能キー名を含むテキストは、KEYI タグのピリオドに続くテキストとして定義されます。

すべての機能キーは、KEYI タグをキー・リスト (KEYL) および EKEYL タグの間に配置することにより定義されます。KEYL タグの NAME 属性に指定された名前は、PANEL タグの KEYL 属性にも指定されます。

Enter などのいくつかの機能キーには、KEYI タグに指定されたテキストがありません。この場合、画面にはテキストが表示されません。ただし、すべての機能キーは KEYI タグを使用して定義して、各キーにどの操作が割り当てられているか、および機能キーのヘルプを表示する際にどのヘルプ・モジュールを使用するかを UIM が認識する必要があります。

データ表示パネルに必要なタグ

図 110は、UIM データ表示パネルを作成するために必要なタグを示しています。必須タグおよびオプション・タグの例については、337 ページの『データ表示パネルの例のソース』を参照してください。

注: 図 110にあるタグは、属性が必要です。それらの属性は、示されていません。それらの属性を指定しなければ、図 110の例はコンパイルされません。これらの必須タグに必要な属性について詳しくは、『付録 A. UIM パネル・グループ定義言語』を参照してください。

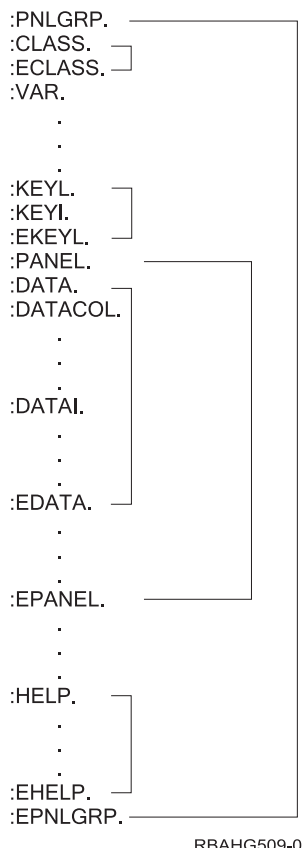


図 110. データ表示パネルの必須 UIM タグ

データ表示パネルの例のソース

これは、ライブラリー QUSRTOOL 内のソース・ファイル QATTUIM にあるメンバー T0011PN2 の部分リストです。

```
. * -----
. *
. * Beginning of panel group source.
. *
. * -----
:PNLGRP   DFTMSGF=t0011msgf2
          SUBMSGF=t0011msgf2.
. *
. * The import tag specifies that all help is to be found
. * in panel group T0011HL2 searching the library list.
:IMPORT   NAME='*'
          PNLGRP=t0011h12.
. *
. * -----
. * Define all variable classes
. * -----
. * -----
. * Option
. * Note: Need WIDTH=1 to preserve column alignment on confirmation panel.
:CLASS    NAME=optc1
          BASETYPE='ACTION'
          WIDTH=1.
:ECLASS.
. * -----
. * Object name
:CLASS    NAME=namec1
          BASETYPE='OBJNAME 10'.
:ECLASS.
. * -----
. * Library name
:CLASS    NAME=libc1
          BASETYPE='OBJNAME 10'.
:TL.
:TI       VALUE='*"LIBL"*LIBL
:TI       VALUE='*"CURLIB"*CURLIB
:ETL.
:ECLASS.
. * -----
. * File attribute
:CLASS    NAME=attrc1
          BASETYPE='CHAR 10'
          CASE=UPPER.
:ECLASS.
. * -----
. * Descriptive text
:CLASS    NAME=textc1
          BASETYPE='IGC 50'
          SUBST=QUOTED.
:ECLASS.
. * -----
. * Source type
:CLASS    NAME=srctypc1
          BASETYPE='CHAR 10'.
```

```

:ECLASS.
.* -----
.* Date
:CLASS    NAME=datec1  E
          BASETYPE='DATE'.

:TL.
:TI VALUE=' '      '.
:TI VALUE='0000000'.
:ETL.
:ECLASS.

.* -----
.* Time
:CLASS    NAME=timec1  E
          BASETYPE='TIME'.

:TL.
:TI VALUE=' '      '.
:TI VALUE='0000000'.
:ETL.
:ECLASS.

.* -----
.* Yes or No flag class
:CLASS    NAME=yesnoc1
          BASETYPE='CHAR 1'
          WIDTH=3.

:TL.
:TI VALUE='0'.No
:TI VALUE='1'.Yes
:ETL.
:ECLASS.

.* -----
.* Type of file
:CLASS    NAME=typfc1
          BASETYPE='CHAR 1'
          WIDTH=2.

:TL.
:TI VALUE='0'.PF
:TI VALUE='1'.LF
:ETL.
:ECLASS.

.* -----
.* Binary 31
:CLASS    NAME=bin31
          BASETYPE='BIN 31'.

:ECLASS.
:
:
Additional CLASS tags in member T0011PN2 are not shown here
:
.* -----
.* Classes for pad variables in variable record definitions.
:CLASS    NAME=pad1c1
          BASETYPE='CHAR 1'.

:ECLASS.
:CLASS    NAME=pad2c1
          BASETYPE='CHAR 2'.

:ECLASS.
:CLASS    NAME=pad10c1
          BASETYPE='CHAR 10'.

:ECLASS.
:CLASS    NAME=pad13c1
          BASETYPE='CHAR 13'.

:ECLASS.
:CLASS    NAME=pad48c1
          BASETYPE='CHAR 48'.

:ECLASS.
:CLASS    NAME=pad50c1

```

```

        BASETYPE='CHAR 50'.
:ECLASS.
.*
.* -----
.* Define all dialog variables
.* -----
.*
.* -----
.* Variables for file and library
.* -----
.*
.* File name
:VAR      NAME=file A
          CLASS=namecl.
.* -----
.* Library name
:VAR      NAME=lib B
          CLASS=libcl.
.* -----
.* attributes
:VAR      NAME=fattr
          CLASS=attrcl.
.* -----
.*
.* -----
.* Variables for list of members
.* -----
.*
.* option for list of members
:VAR      NAME=mopt
          CLASS=optcl.
.* -----
.* Object name
:VAR      NAME=mbr C
          CLASS=namecl.
.* -----
.* member type
:VAR      NAME=mtype
          CLASS=attrcl.
.* -----
.* Descriptive text
:VAR      NAME=mtext
          CLASS=textcl.
.*
.* -----
.* Variables for member description
.* -----
.*
.* Source type
:VAR      NAME=mbrsrc
          CLASS=srctypcl.
.* -----
.* Creation date
:VAR      NAME=mbrcrtdat
          CLASS=datecl.
.* -----
.* Creation time
:VAR      NAME=mbrcrttim
          CLASS=timecl.
.* -----
.* Member last source change date
:VAR      NAME=mbrschgdat
          CLASS=datecl.
.* -----
.* Member last source change time
:VAR      NAME=mbrschgtim
          CLASS=timecl.

```

```

.* -----
.* Member remote source file indicator
:VAR      NAME=mbrsrcfil
          CLASS=yesnocl.
.* -----
.* Member remote file indicator
:VAR      NAME=mbrremote
          CLASS=yesnocl.
.* -----
.* Member type of file
:VAR      NAME=mbrtypf
          CLASS=typfcl.
.* -----
.* Member ODP sharing
:VAR      NAME=mbrodpshr
          CLASS=yesnocl.
.* -----
.* Member current number of records
:VAR      NAME=mbrcurrec
          CLASS=bin31.
.* -----
.* Member number of deleted records
:VAR      NAME=mbrdltrec
          CLASS=bin31.
.* -----
.* Member data space size
:VAR      NAME=mbrspsiz
          CLASS=bin31.
.* -----
.* Member access path size
:VAR      NAME=mbracpsiz
          CLASS=bin31.
.* -----
.* Number of database file members
:VAR      NAME=mbrdbfmbrs
          CLASS=bin31.
.* -----
.* Member change date.
:VAR      NAME=mbrchgdat
          CLASS=datecl.
.* -----
.* Member change time
:VAR      NAME=mbrchgtim
          CLASS=timecl.
.* -----
.* Member save date
:VAR      NAME=mbrsavdat
          CLASS=datecl.
.* -----
.* Member save time
:VAR      NAME=mbrsavtim
          CLASS=timecl.
.* -----
.* Member restore date
:VAR      NAME=mbrstdat
          CLASS=datecl.
.* -----
.* Member restore time
:VAR      NAME=mbrsttim
          CLASS=timecl.
.* -----
.* Member expiration date
:VAR      NAME=mbrexpdat
          CLASS=datecl.
.* -----

```

```

.* Member expiration time
:VAR      NAME=mbrexptim
          CLASS=timecl.
.* -----
.* Member number of days used
:VAR      NAME=mbrdysuse
          CLASS=bin31.
.* -----
.* Member date last used
:VAR      NAME=mbrlastuse
          CLASS=datecl.
.* -----
.* Member use reset date
:VAR      NAME=mbrresetdat
          CLASS=datecl.
.* -----
.*
.*
Additional VAR tags in member T0011PN2 are not shown here
.*
.* -----
.* Variables for padding in variable record definitions.
.* Padding is needed in variable records so the layout
.* of the record matches a list format returned from
.* an API. The pad variables are used as placeholders for
.* variables not used in the API format or for reserved space
.* in the API format.
.* -----
:VAR      NAME=pad1
          CLASS=pad1cl.
:VAR      NAME=pad2
          CLASS=pad2cl.
:VAR      NAME=pad10
          CLASS=pad10cl.
:VAR      NAME=pad13
          CLASS=pad13cl.
:VAR      NAME=pad48
          CLASS=pad48cl.
:VAR      NAME=pad50
          CLASS=pad50cl.
.*
.*
The first VARRCD tags in member T0011PN2 are not shown here
.*
.* -----
.* Define a variable record for list of members.
.* The layout of this record is designed to match the
.* List Database File Members API (QUSLMBR) format name MBRL0200.
.* -----
:VARRCD   NAME=mbrl0200
          VARS='mbr mtype pad13 pad13 mtext'
          NOPUT='pad13'
          NOGET='mtype pad13 mtext'
.*
.* -----
.* Define a variable record for member description.
.* The layout of this record is designed to match the
.* Retrieve Member Description API (QUSRMBRD) format name MBRD0200.
.* -----
:VARRCD   NAME=mbrd0200
          VARS='pad48 mbrsrc mbrcrtdat mbrcrttim mbrschgdat'
          VARS='mbrschgtim pad50 mbrsrcfil mbrremote mbrtypf'
          VARS='mbrdpsshr pad2 mbrcurrrec mbrdltrrec mbrspcsiz'

```

```

VARS='mbracpsiz mbrdbfmbms mbrchgdat mbrchgtim mbrsavdat'
VARS='mbrsavtim mbrstddat mbrsttim mbrexpdat mbrexpim'
VARS='mbrdysuse mbrlastuse mbrsetdat'
NOPUT='pad48 pad50 pad2'
NOGET='pad48 pad50 pad2'

```

```

:
:
Additional VARRCD tags in member T0011PN2 are not shown here
:
:

```

```

All LISTDEF tags in member T0011PN2 are not shown here
:
:

```

```

.*
.* -----
.* Define all conditions
.* -----

```

```

.* Condition for physical files
:COND      NAME=pf
           EXPR='fattr="PF      "'

```

```

:
:
Additional COND tags in member T0011PN2 are not shown here
:
:

```

```

.* -----
.* Condition for source files
:COND      NAME=srcpf
           EXPR='mbrsrcfil="1"'

```

```

.* -----
.* Condition for member in a physical file
:COND      NAME=mbrpf
           EXPR='mbrtypf="0"'

```

```

.* -----
.* Condition for member in a logical file
:COND      NAME=mbrlf
           EXPR='mbrtypf="1"'

```

```

:
:
Additional COND tags in member T0011PN2 are not shown here
:
:

```

```

All TT tags in member T0011PN2 are not shown here
:
:

```

```

All MBAR tags in member T0011PN2 are not shown here
:
:

```

```

The first KEYL tags in member T0011PN2 are not shown here
:
:

```

```

.*
.* -----
.* Define basic key for panels without a command line
.* or multiple views.
.* -----

```

```

:KEYL      NAME=basickeys  5
           HELP=keyl.
:KEYI      KEY=F1
           HELP=helpf1
           ACTION=HELP.
:KEYI      KEY=F3
           HELP=exit
           ACTION='EXIT SET'
           VARUPD=NO.

```

```

F3=Exit
:KEYI      KEY=F12
           HELP=cancel
           ACTION='CANCEL SET'
           VARUPD=NO.
F12=Cancel
:KEYI      KEY=F24
           HELP=morekeys
           ACTION=MOREKEYS.
F24=More keys
:KEYI      KEY=ENTER
           HELP=enter
           ACTION=ENTER.
:KEYI      KEY=HELP
           HELP=help
           ACTION=HELP.
:KEYI      KEY=PAGEDOWN
           HELP=pagedown
           ACTION=PAGEDOWN.
:KEYI      KEY=PAGEUP
           HELP=pageup
           ACTION=PAGEUP.
:KEYI      KEY=PRINT
           HELP=print
           ACTION=PRINT.
:EKEYL.
.
.
.
The first PANEL tags in member T0011PN2 are not shown here
.
.
.*
.* -----
.* Define Display Member Description panel
.* -----
:PANEL     NAME=dspnbr
           HELP='dspnbr/'
           KEYL=basickeys 5
           ENTER='RETURN 500'
           TOPSEP=SPACE.
Display Member Description 1
.*
.* -----
.* Define a data presentation area to display the
.* library/file and member name whose description is displayed.
.* -----
:DATA     DEPTH=3
           SCROLL=NO
           LAYOUT=2 D
           BOTSEP=SPACE
           COMPACT
.
.* -----
.* Divide the layout width into two columns.
.* The first column is for the prompt text with leader dots.
.* The second column is for the variable values.
:DATACOL  WIDTH=22.
:DATACOL  WIDTH='*'.
.* -----
.* Display qualified file name
:DATAGRP  GRPSEP=QINDENT
           HELP='dspnbr/filelib'
           COMPACT
.
:DATAI    VAR=file 2
           USAGE=OUT

```

```

File
:DATAI  VAR=lib 2
        USAGE=OUT

Library
:EDATAGRP.
.* -----
.* Display member name
:DATAI  VAR=mbr 2
        HELP='dspmbr/mbr'
        USAGE=OUT

Member
.*
:EDATA.
.*
.* -----
.* Define a data presentation area to display the
.* member definition.
.* -----
:DATA   DEPTH='*'
        SCROLL=YES 4
        LAYOUT=1
        BOTSEP=SPACE
.*
.* -----
.* Divide the layout width into two columns.
.* The first column is for the prompt text with leader dots.
.* The second column is for the variable values.
:DATACOL WIDTH=35.
:DATACOL WIDTH='*'.
.* -----
.* Display information about the file
:DATAGRP GRPSEP=NONE
        COMPACT
.*
.* -----
.* Display type of file
:DATAI  VAR=mbrtypf 3
        HELP='dspmbr/mbrtypf'
        USAGE=OUT

Type of file
.* -----
.* Display remote file
:DATAI  VAR=mbrremote
        HELP='dspmbr/mbrremote'
        USAGE=OUT

Remote file
.* -----
.* Display ODP sharing
:DATAI  VAR=mbrodpsshr
        HELP='dspmbr/mbrodpsshr'
        USAGE=OUT

Allow ODP sharing
:EDATAGRP.
.* -----
.* Display information only if file is a source file
:DATAGRP GRPSEP=NONE
        COMPACT
        COND=srcpf

```



```

.* -----
.* Display source type
:DATAI  VAR=mbrsrc
        HELP='dspmbr/mbrsrc'
        USAGE=OUT
.*
Source type
.* -----
.* Display last source change date and time
:DATAI  VAR=mbrschgdat
        HELP='dspmbr/mbrschgdt'
        USAGE=OUT
.*
Last source change date and time
:DATAIX VAR=mbrschgtim
        USAGE=OUT
.*
:EDATAGRP.
.* -----
.* Display create and change information
:DATAGRP GRPSEP=NONE
        COMPACT
.*
.* -----
.* Display creation date and time
:DATAI  VAR=mbrcrtdat
        HELP='dspmbr/mbrcrtdt'
        USAGE=OUT
.*
Creation date and time
:DATAIX VAR=mbrcrttim
        USAGE=OUT
.*
.* -----
.* Display change date and time
:DATAI  VAR=mbrchgdat
        HELP='dspmbr/mbrchgdt'
        USAGE=OUT
.*
Change date and time
:DATAIX VAR=mbrchgtim
        USAGE=OUT
.*
:EDATAGRP.
.* -----
.* Display information about the size of the member
:DATAGRP GRPSEP=NONE
        COMPACT
.*
.* -----
.* Display current records for physical file member
:DATAI  VAR=mbrcurrec
        COND=mbrpf
        HELP='dspmbr/mbrcurrec'
        USAGE=OUT
.*
Number of records
.* -----
.* Display current index entries for logical file member
:DATAI  VAR=mbrcurrec
        COND=mbrlf
        HELP='dspmbr/mbrcurrec'
        USAGE=OUT
.*
Number of index entries
.* -----

```

```

.* Display deleted records
:DATAI  VAR=mbrdltrec
        HELP='dspmbr/mbrdltrec'
        USAGE=OUT
.
Deleted records
.* -----
.* Display data space size for physical file member
:DATAI  VAR=mbrspcsiz
        COND=mbrpf
        HELP='dspmbr/mbrspcsiz'
        USAGE=OUT
.
Data space size
.* -----
.* Display access path size
:DATAI  VAR=mbracpsiz
        HELP='dspmbr/mbracpsiz'
        USAGE=OUT
.
Access path size
.* -----
.* Display database file members for logical file member
:DATAI  VAR=mbrdbfmbhrs
        COND=mbrlf
        HELP='dspmbr/mbrdbfmbhrs'
        USAGE=OUT
.
Number of database file members
:EDATAGRP.
.* -----
.* Display save restore information
:DATAGRP GRPSEP=NONE
        COMPACT
.
.* -----
.* Display save date and time
:DATAI  VAR=mbrsavdat 3
        HELP='dspmbr/mbrsavdt'
        USAGE=OUT
.
Save date and time
:DATAIX VAR=mbrsavtim 3
        USAGE=OUT
.
.* -----
.* Display restore date and time
:DATAI  VAR=mbrrstdat
        HELP='dspmbr/mbrrstdt'
        USAGE=OUT
.
Restore date and time
:DATAIX VAR=mbrrsttim
        USAGE=OUT
.
:EDATAGRP.
.* -----
.* Display expiration date
:DATAI  VAR=mbrexpdat
        HELP='dspmbr/mbrexpdt'
        USAGE=OUT
.
Expiration date and time
:DATAIX VAR=mbrexp tim
        USAGE=OUT
.
.* -----

```

```

.* Display usage information
:DATAGRP  GRPSEP=NONE
          COMPACT
          .
.* -----
.* Display number of days used
:DATAI   VAR=mbrdysuse
          HELP='dspmbr/mbrdysuse'
          USAGE=OUT
          .
Number of days used
.* -----
.* Display date last used
:DATAI   VAR=mbrlastuse
          HELP='dspmbr/mbrlastuse'
          USAGE=OUT
          .
Date last used
.* -----
.* Display use reset date
:DATAI   VAR=mbrresetdat
          HELP='dspmbr/mbrresetdat'
          USAGE=OUT
          .
Use reset date
:EDATAGRP.
.* -----
.* Display text description
:DATAI   VAR=mtext
          HELP='dspmbr/text'
          USAGE=OUT
          .
Text
.*
:EDATA.
.*
:EPANEL.
  .
  Additional PANEL tags in member T0011PN2 are not shown here
  .
.*
.* -----
.* End of panel group source
.* -----
:EPNLGRP.

```

データ表示パネルのアプリケーション・プログラミング

334 ページの図 108に示されたデータ表示パネルを表示するアプリケーション・プログラムの例は、ライブラリー QUSRTOOL 内のソース・ファイル QATTSYSC にあるメンバー T0011CP3 に含まれています。これは適切な UIM アプリケーション・プログラミング・インターフェース (API) を呼び出してパネルを表示する、ILE C/C++ プログラムです。このプログラムは、316 ページの図 104に示されたリスト・パネルの例から、オプション 5 (表示) を処理するために UIM によって呼び出されます。

- | UIM API を使用して書かれた RPG アプリケーションの一般的な例は、QUSRTOOL にあります。ライブラリー QUSRTOOL 内のソース・ファイル QATTRPG にあるメンバー T0011RP5 を参照してください。

データ表示パネルの例を表示するプログラムをいずれかの言語で記述するには、そのプログラムが以下を行う必要があります。

1. プログラムが、316 ページの図 104に示されたリスト・パネルの例からオプション 5 (表示) を処理するために UIM によって呼び出されない場合、そのプログラムは最初に以下を行う必要があります。これはたとえば、ユーザーが CL コマンドを使用してメンバー記述を直接表示できる場合に該当します。

- a. Open Display Application (QUIOPNDA) API を呼び出して、パネル・グループを開きます。パネル・グループは、パネル・グループ作成 (CRTPNLGRP) コマンドを使用して作成済みでなければなりません。
- b. 以下のダイアログ変数の値を含むバッファをセットアップします。

FILE ファイルの名前を示す CHAR 10 変数。

LIB ファイルが存在するライブラリーの名前を示す CHAR 10 変数。

FATTR

ファイルのファイル属性を示す CHAR 10 変数。この変数は、物理ファイルだけに許可されるリスト・オプションに対して条件設定するために使用されます。

- c. Put Dialog Variable (QUIPUTV) API を呼び出して、変数レコード FILELIB および直前のステップで初期化されたバッファを使用することにより、ダイアログ変数の内容を変更します。
- d. 以下のダイアログ変数の値を含むバッファをセットアップします。

MBR メンバー名を示す CHAR 10 変数。

MTYPE

メンバー・タイプを示す CHAR 10 変数。

PAD13

バッファ内の CHAR 13 予約済みスペース。

PAD13

バッファ内の CHAR 13 予約済みスペース。

MTEXT

メンバーについて説明するテキストを示す CHAR 50 変数。

注: このバッファのレイアウトは、QUSLMBR API によって戻される、ユーザー・スペース内の項目のレイアウトと一致するように設計されています。そのため、バッファをセットアップする代わりに、アプリケーション・プログラムはユーザー・スペースに存在するままのバッファを渡すことができます。

- e. Put Dialog Variable (QUIPUTV) API を呼び出して、変数レコード MBRL0200 および直前のステップで初期化されたバッファを使用することにより、ダイアログ変数の内容を変更します。

注: プログラムが、リスト・パネルの例からオプション 5 (表示) を処理するために UIM によって呼び出される場合、ファイル名およびライブラリー名はリスト・パネルの表示前にアプリケーション・プログラムによって設定済みです。さらに、UIM がリスト・オプション処理を実行するときには、MBR、MTYPE、および MTEXT ダイアログ変数は、オプション 5 を処理するアプリケーション・プログラムを呼び出すリスト項目からの値にすでに設定されています。

2. 以下のダイアログ変数の値を含むバッファをセットアップします。

PAD48 バッファ内の CHAR 48 予約済みスペース。

MBSRC メンバーのソース・タイプを示す CHAR 10 変数。

MBRCRTDAT BASETYPE=DATE ダイアログ変数に必要な形式での、メンバーの作成日付を示す CHAR 7 変数。

- MBRCRTTIM** BASETYPE=TIME ダイアログ変数に必要な形式での、メンバーの作成時刻を示す CHAR 6 変数。
- MBRSCHGDAT**
BASETYPE=DATE ダイアログ変数に必要な形式での、メンバーのソース変更日付を示す CHAR 7 変数。
- MBRSCHGTIM**
BASETYPE=TIME ダイアログ変数に必要な形式での、メンバーのソース変更時刻を示す CHAR 6 変数。
- PAD50** バッファ内での CHAR 50 予約済みスペース。
- MBRSRCFIL** メンバーがソース・ファイル内に存在するかどうかを示す CHAR 1 変数 ("0"=いいえ、"1"=はい)。
- MBRREMOTE**
メンバーがリモート・ファイル内に存在するかどうかを示す CHAR 1 変数 ("0"=いいえ、"1"=はい)。
- MBRTYPF** メンバーが物理ファイルまたは論理ファイルのどちらに存在するかを示す CHAR 1 変数 ("0"=物理、"1"=論理)。
- MBRODPSHR** メンバーに ODP 共有が許可されているかどうかを示す CHAR 1 変数 ("0"=いいえ、"1"=はい)。
- PAD2** バッファ内での CHAR 2 予約済みスペース。
- MBRCURREC** メンバー内のレコード数を含む BINARY 4 変数。
- MBRDLTREC** メンバー内の削除済みレコード数を含む BINARY 4 変数。
- MBRSPCSIZ** メンバーのスペースのサイズを含む BINARY 4 変数。
- MBRACPSIZ** メンバーのアクセス・パスを含む BINARY 4 変数。
- MBRDBFMBRS**
この論理ファイル・メンバーのデータベース・ファイル・メンバー数を含む BINARY 4 変数。
- MBRCHGDAT**
BASETYPE=DATE ダイアログ変数に必要な形式での、メンバーの最終変更日付を示す CHAR 7 変数。
- MBRCHGTIM** BASETYPE=TIME ダイアログ変数に必要な形式での、メンバーの最終変更時刻を示す CHAR 6 変数。
- MBRSAVDAT** BASETYPE=DATE ダイアログ変数に必要な形式での、メンバーの最終保存日付を示す CHAR 7 変数。
- MBRSAVTIM** BASETYPE=TIME ダイアログ変数に必要な形式での、メンバーの最終保存時刻を示す CHAR 6 変数。
- MBRRSTDAT** BASETYPE=DATE ダイアログ変数に必要な形式での、メンバーの最終リストア日付を示す CHAR 7 変数。
- MBRRSTTIM** BASETYPE=TIME ダイアログ変数に必要な形式での、メンバーの最終リストア時刻を示す CHAR 6 変数。
- MBREXPDAT** BASETYPE=DATE ダイアログ変数に必要な形式での、メンバーの有効期限日付を示す CHAR 7 変数。

MBREXPTIM BASETYPE=TIME ダイアログ変数に必要な形式での、メンバーの有効期限刻を示す CHAR 6 変数。

MBRDYSUSE メンバーが使用された日数を含む BINARY 4 変数。

MBRLASTUSE

BASETYPE=DATE ダイアログ変数に必要な形式での、メンバーの最終使用日付を示す CHAR 7 変数。

MBRRSETDAT

BASETYPE=DATE ダイアログ変数に必要な形式での、メンバーの使用情報がリセットされた日付を示す CHAR 7 変数。

注: このバッファのレイアウトは、フォーマット名 MBRD0200 を使用する Retrieve Member Description (QUSRMBRD) API によって戻される、データのレイアウトと一致するように設計されています。そのため、バッファをセットアップする代わりに、アプリケーション・プログラムは API によって戻されたままのバッファを渡すことができます。

3. Put Dialog Variable (QUIPUTV) API を呼び出して、変数レコード MBRD0200 および直前のステップで初期化されたバッファを使用することにより、ダイアログ変数の内容を変更します。
4. Display Panel (QUIDSPP) API を呼び出して、表示パネル WRKMBR を表示します。以下の 1 つが生じたとき、UIM は制御をアプリケーションに戻します。
 - ユーザーがコマンド行にリスト・オプションまたはコマンドを入力しないで Enter キーを押したとき。機能要求パラメーターとして QUIDSPP API に渡されるプログラム変数は、500 に設定されます。これは、WRKMBR パネルを定義する PANEL タグで、ENTER='RETURN 500' が指定されているためです。
 - ユーザーが F12 (取り消し) キーを押したとき。機能要求パラメーターとして QUIDSPP API に渡されるプログラム変数は、-8 に設定されます。これは CANCEL ダイアログ・コマンドに定義された値です。
 - ユーザーが F3 (終了) キーを押したとき。機能要求パラメーターとして QUIDSPP API に渡されるプログラム変数は、-4 に設定されます。これは CANCEL ダイアログ・コマンドに定義された値です。
5. プログラムが、316 ページの図 104 に示されたリスト・パネルの例からオプション 5 (表示) を処理するために UIM によって呼び出されない場合、そのプログラムは以下を行う必要があります。
 - a. Close Application (QUICLOA) API を呼び出して、UIM アプリケーションを閉じます。これにより、UIM アプリケーションによって使用されているシステム・リソースが解放されます。

データ入力パネル

データ入力パネルは、表示された 1 つ以上のダイアログ変数にユーザーが新規の値を入力できるデータ表示パネルの一形式です。データ項目 (DATAI) およびデータ項目エクステンダー (DATAIX) タグの USAGE 属性は、タグによって表示されるダイアログ変数にユーザーが新規の値を入力できるかどうかを定義します。

USAGE=INOUT が指定された場合、ユーザーはダイアログ変数に新規の値を入力できます。垂直レイアウトを使用するデータ域では、項目の記述テキストに続くリーダー・ドットがコロンではなくピリオドで終わります。コロンは入力を許可しないデータ項目に対して使用されます。

UIM が制御を Display Panel (QUIDSPP) API から戻すとき、表示パネル (PANEL) タグの ENTER 属性にある RETURN ダイアログ・コマンドに指定された値が機能要求パラメーターに含まれていれば、アプリケーション・プログラムは Get Dialog Variable (QUIGETV) API を使用して、ユーザーによって変更されたダイアログ変数の値を取得します。

メニュー・バーのあるパネルの作成

メニュー・バーは、任意のフルスクリーン・パネルに対して定義できます。メニュー・バーは、パネルの拡張として機能し、メニュー・バーの選択項目の定義を含んでいます。メニュー・バーの各選択項目には、プルダウン・メニューが定義されています。ユーザーがメニュー・バーの選択項目を選択すると、UIM はメニュー・バーの下のウィンドウにプルダウン・メニューを表示します。プルダウン・メニュー内の各選択項目は、実行可能なアクションを表します。この例では、「Work with File Members (ファイル・メンバー処理)」パネルがメニュー・バーを使用して定義されます。このパネルの機能は、316 ページの『リスト・パネルの作成』に示されたパネルと似ています。このパネルはリスト・パネルの例と多少異なって見えますが、メニュー・バーを使用するこのパネルの例は同じ機能を実行します。

図 111にあるパネルは、同様にメニュー・バーのある処置リスト・パネルの例を示しています。この例は、メニュー・バーが定義される方法、およびそれが他のパネル定義と相互作用する方法を説明しています。この例は、パネルの処置リスト部分を定義する方法については説明していません。処置リストの作成方法について詳しくは、316 ページの『リスト・パネルの作成』を参照してください。メニュー・バー域のヘルプを作成する方法について詳しくは、431 ページの『メニュー・バー域のヘルプ』を参照してください。

パネルの例は、プルダウン・メニューが表示された状態で示されています。プルダウンは、ユーザーがリスト内の項目を選択して Enter キーを押したためにアクティブになっています。

```

Member  View  Help  1
-----
:  3. Copy...  2      : File Members
:  4. Remove... :
:  5. Display... : F4 for list
:                : library, *CURLIB, *LIBL
:  7. Reorganize :
:  8. Member description... :
:  9. Clear...   :
:                :
:  90. Exit      3      F3 : XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
:                : XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
:XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
/ XXXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
/ XXXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
- XXXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
- XXXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
/ XXXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
- XXXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
More...

Parameters for options 3 and 5 or command
====>
F3=Exit  F4=Prompt  F9=Retrieve  F10=Actions  F12=Cancel

```

図 111. メニュー・バーのあるパネルの例

パネルの例に示されている参照番号は、画面には表示されません。これらは図示のためのものであり、354ページの『メニュー・バーのあるパネルの例のソース』に示されている UIM タグ・ソース内にもあります。これらの参照番号は、ソースのどの部分がパネルに表示されるテキストおよび情報を定義しているかを示します。

- 1** メニュー・バー内の各選択項目は、メニュー・バー選択項目 (MBCAR) タグを使用して定義されます。パネルに示される選択項目は、MBCAR タグのピリオドに続くテキストとして定義されます。
メニュー・バーのすべての選択項目は、MBCAR タグをメニュー・バー (MBCAR) タグと EMBAR タグとの間に配置することによって定義されます。MBCAR タグの NAME 属性に指定された名前は、PANEL タグの MBAR 属性にも指定されます。
- 2** プルダウン・メニュー内の各オプションは、プルダウン・フィールド選択項目 (PDFLDC) タグを使用して定義されます。オプション番号は OPTION 属性を使用して定義され、オプション番号に続くテキストは PDFLDC タグに続くテキストとして定義されます。
プルダウン・メニューのすべての選択項目は、PDFLDC タグをプルダウン・フィールド (PDFLD) タグと EPDFLD タグとの間に配置することによって定義されます。PDFLDC タグおよび EPDFLD タグは、MBCAR タグと EMBAR タグの間に配置されます。
- 3** プルダウン・メニューの右側に表示されるアクセラレーター・キー記述は、PDACCEL タグのピリオドに続くテキストとして定義されます。アクセラレーターは、プルダウン・メニュー内のオプションと同じ機能を実行するキーです。PDACCEL タグは、アクセラレーターが適用されるオプションの PDFLDC タグの直後に配置されます。

このパネルで、アプリケーション・ユーザーは以下のいずれかの方法を使用することにより、リスト内の項目に対して操作を実行できます。

- ユーザーがリスト内の項目を選択して Enter キーを押すと、UIM はメニュー・バー内のそのメンバー選択項目に対するプルダウン・メニューを表示します。これは、**A** に示されている PANEL タグに、SELECT=PULLDOWN が指定されているためです。

ユーザーがプルダウン・メニューからオプション 3、4、5、7、8、または 9 を選択すると、ユーザーが選択したそれぞれのリスト項目ごとに 1 回ずつ、UIM はそのオプションに定義された操作を実行します。UIM がこれを行うのは、**B** で ACTFOR=LIST が PDFLDC タグに指定されているためです。ユーザーがリスト内の項目を選択しないでプルダウン・メニューを表示すると、UIM はこれらのオプションを選択不可にします。

ユーザーがプルダウン・メニューからオプション 90 を選択すると、UIM はパネルを表示したアプリケーション・プログラムに制御を戻します。UIM がこれを行うのは、**C** で PDFLDC タグに ACTFOR=PANEL が指定されているためです。PDFLDC タグの ACTION 属性に EXIT が指定されているため、UIM はユーザーが終了を要求したことを示して制御を呼び出し側プログラムに戻します。

- ユーザーが選択したリスト項目の横にオプション番号を入力して Enter キーを押すと、UIM は各オプションの操作を実行します。UIM がこれを行うのは、**D** でリスト域 (LIST) タグに ACTOR=UIM が指定されていて、**E** のリスト・アクション (LISTACT) タグにより、UIM が各オプション番号に対して実行する操作が指定されているためです。

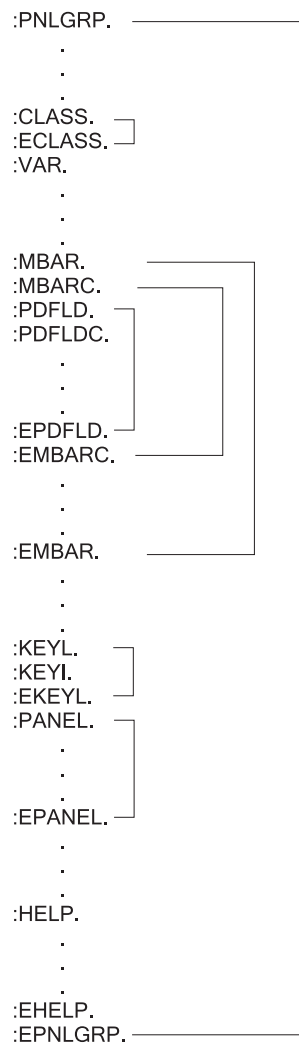
この場合、リストのオプション番号を説明するテキストがパネル上に表示されないことに注意してください。これは、LISTACT タグのピリオドに続いて指定されているテキストがないためです。このパネルに定義される主要なインターフェースは、リスト内の項目を選択して、プルダウン・メニューから操作を選択するというものです。熟練したユーザーのための高速な代替手段として、リスト・オプション番号も使用可能です。リスト・オプション番号はパネル上に表示されませんが、リストの入力列にカーソルを置いてヘルプ・キーを押したときに表示されるヘルプにそれらが説明されています。

この例では、プルダウン・メニュー内のオプション番号に対する操作は、リストに入力可能なオプション番号に対する操作と同じです。この設計は一貫性のあるユーザー・インターフェースのために推奨されますが、UIM ではアプリケーション開発者がプルダウン・メニュー内の操作を、同じ番号のリスト・オプションの操作とは異なるものに定義することも可能です。

メニュー・バーのあるパネルに必要なタグ

図 112は、メニュー・バーのある UIM パネルを作成するために必要なタグを示しています。必須タグおよびオプション・タグの例については、354 ページの『メニュー・バーのあるパネルの例のソース』を参照してください。

注: 図 112にあるタグは、属性が必要です。それらの属性は、示されていません。それらの属性を指定しなければ、図 112の例はコンパイルされません。これらの必須タグに必要な属性について詳しくは、『付録 A. UIM パネル・グループ定義言語』を参照してください。



RBAHG510-0

図 112. メニュー・バーのあるパネルの必須 UIM タグ

メニュー・バーのあるパネルの例のソース

これは、ライブラリー QUSRTOOL 内のソース・ファイル QATTUIM にあるメンバー T0011PN2 の部分リストです。

```
.* -----
.*
.* Beginning of panel group source.
.*
.* -----
:PNLGRP   DFTMSGF=t0011msgf2
          SUBMSGF=t0011msgf2.
.*
.* The import tag specifies that all help is to be found
.* in panel group T0011HL2 found by searching the library list.
:IMPORT   NAME='*'
          PNLGRP=t0011h12.
.*
.* -----
.* Define all variable classes
.* -----
.* -----
.* Option
.* Note: Need WIDTH=1 to preserve column alignment on confirmation panel.
:CLASS    NAME=optc1
          BASETYPE='ACTION'
          WIDTH=1.
:ECLASS.
.* -----
.* Object name
:CLASS    NAME=namec1
          BASETYPE='OBJNAME 10'.
:ECLASS.
.* -----
.* Library name
:CLASS    NAME=libc1
          BASETYPE='OBJNAME 10'.
:TL.
:TI       VALUE='"*LIBL"*.*LIBL
:TI       VALUE='"*CURLIB"*.*CURLIB
:ETL.
:ECLASS.
.* -----
.* File attribute
:CLASS    NAME=attrc1
          BASETYPE='CHAR 10'
          CASE=UPPER.
:ECLASS.
.* -----
.* Descriptive text
:CLASS    NAME=textc1
          BASETYPE='IGC 50'
          SUBST=QUOTED.
:ECLASS.
.* -----
.* Source type
:CLASS    NAME=srctypc1
          BASETYPE='CHAR 10'.
:ECLASS.
:
Additional CLASS tags in member T0011PN2 are not shown here
```

```

:
.* -----
.* Command line parameters
:CLASS   NAME=parmc1
         BASETYPE='CHAR 255'.
:ECLASS.
.* -----
.* Exit program specification for CALL dialog command
:CLASS   NAME=exitc1
         BASETYPE='CHAR 20'.
:ECLASS.
.* -----
.* View number
:CLASS   NAME=vwnumc1
         BASETYPE='BIN 15'.
:ECLASS.
.* -----
.* Classes for pad variables in variable record definitions.
:CLASS   NAME=pad1c1
         BASETYPE='CHAR 1'.
:ECLASS.
:CLASS   NAME=pad2c1
         BASETYPE='CHAR 2'.
:ECLASS.
:CLASS   NAME=pad10c1
         BASETYPE='CHAR 10'.
:ECLASS.
:CLASS   NAME=pad13c1
         BASETYPE='CHAR 13'.
:ECLASS.
:CLASS   NAME=pad48c1
         BASETYPE='CHAR 48'.
:ECLASS.
:CLASS   NAME=pad50c1
         BASETYPE='CHAR 50'.
:ECLASS.
.*
.* -----
.* Define all dialog variables
.* -----
.*
.* -----
.* Variables for file and library
.* -----
.*
.* File name
:VAR     NAME=file
         CLASS=namec1.
.* -----
.* Library name
:VAR     NAME=lib
         CLASS=libc1.
.* -----
.* attributes
:VAR     NAME=fattr
         CLASS=attrc1.
.* -----
.*
.* -----
.* Variables for list of members
.* -----
.*
.* option for list of members
:VAR     NAME=mopt
         CLASS=optc1.
.* -----

```

```

.* Object name
:VAR      NAME=mbr
          CLASS=namecl.
.* -----
.* member type
:VAR      NAME=mtype
          CLASS=attrcl.
.* -----
.* Descriptive text
:VAR      NAME=mtext
          CLASS=textcl.
.
.
Additional VAR tags in member T0011PN2 are not shown here
.
.*
.* -----
.* Variable for command line parameters
.* -----
.* -----
.* Command line parameters
:VAR      NAME=parms
          CLASS=parmc1.
.*
.* -----
.* Variables for specifying CALL/exit programs
.* -----
.* -----
.* Program to call for all UIM exits
:VAR      NAME=exitpgm
          CLASS=exitcl.
.*
.* -----
.* Variables for controlling list views
.* -----
.* -----
.* View number for list of members
:VAR      NAME=mbrview
          CLASS=vwnumcl.
.
.
Additional VAR tags in member T0011PN2 are not shown here
.
.*
.* -----
.* Variables for padding in variable record definitions.
.* Padding is needed in variable records so the layout
.* of the record matches a list format returned from
.* an API. The pad variables are used as placeholders for
.* variables not used in the API format or for reserved space
.* in the API format.
.* -----
:VAR      NAME=pad1
          CLASS=pad1cl.
:VAR      NAME=pad2
          CLASS=pad2cl.
:VAR      NAME=pad10
          CLASS=pad10cl.
:VAR      NAME=pad13
          CLASS=pad13cl.
:VAR      NAME=pad48
          CLASS=pad48cl.
:VAR      NAME=pad50
          CLASS=pad50cl.
.*
.* -----

```

```

.* Define a variable record for file, library and file attribute
.* -----
:VARRCD  NAME=filelib
        VARS='file lib fattr'
        NOGET='fattr'
        .
.*
.* -----
.* Define a variable record for exit program
.* -----
:VARRCD  NAME=exitprog
        VARS='exitpgm'
        .
.*
.* -----
.* Define a variable record for list of members.
.* The layout of this record is designed to match the
.* List Database File Members API (QUSLMBR) format name MBRL0200.
.* -----
:VARRCD  NAME=mbrl0200
        VARS='mbr mtype pad13 pad13 mtext'
        NOPUT='pad13'
        NOGET='mtype pad13 mtext'
        .
        :
        :
Additional VARRCD tags in member T0011PN2 are not shown here
        :
        :
.*
.*
.* -----
.* Define a list of members
.* -----
:LISTDEF NAME=mbrlist
        VARS='mopt mbr mtype mtext'
        MSGID=USR0101.
        .
        :
        :
Additional LISTDEF tags in member T0011PN2 are not shown here
        :
        :
.*
.* -----
.* Define all conditions
.* -----
.* -----
.* Condition for physical files
:COND    NAME=pf
        EXPR='fattr="PF          "'.
.* -----
.* Conditions for views of members list
:COND    NAME=mbrview1
        EXPR='mbrview=0'.
:COND    NAME=mbrview2
        EXPR='mbrview=1'.
        .
        :
Additional COND tags in member T0011PN2 are not shown here
        :
        :
All TT tags in member T0011PN2 are not shown here
        :
        :
.*
.* -----
.* Define menu bar for work with members panel
.* -----

```

```

:MBAR      NAME=mbarmbr  1
           HELP='mbarmbr/'
           .
:MBARC     HELP='mbarmbr/member'
           .
Member 1
:PDFLD.
:PDFLDC   OPTION=3
           ACTFOR=LIST  B
           HELP='mbarmbr/cpyf'
           ACTION='CMD CPYF ?*FROMFILE(&lib./&file.)'
           ACTION=' ?*FROMMBR(&mbr.) &parms.'
```

.

```
Copy... 2
.*
:PDFLDC   OPTION=4
           ACTFOR=LIST
           HELP='mbarmbr/rmvm'
           ACTION='CMD RMVM FILE(&lib./&file.) MBR(&mbr.)'
.*
           CONFIRM=confrmvmb
           USREXIT='CALL exitpgm'
           .
```

```
Remove...
.*
:PDFLDC   OPTION=5
           COND=pf
           ACTFOR=LIST
           HELP='mbarmbr/dsppfm'
           ACTION='CMD DSPPFM FILE(&lib./&file.) MBR(&mbr.) &parms.'
```

.

```
Display...
.*
:PDFLDC   OPTION=7
           COND=pf
           ACTFOR=LIST
           HELP='mbarmbr/rgzm'
           ACTION='CMD RGZPFM FILE(&lib./&file.) MBR(&mbr.)'
```

.

```
Reorganize
.*
:PDFLDC   OPTION=8
           ACTFOR=LIST
           HELP='mbarmbr/dspfd'
           ACTION='CALL exitpgm'
```

.

```
Member description...
.*
:PDFLDC   OPTION=9
           COND=pf
           ACTFOR=LIST
           HELP='mbarmbr/clrm'
.*
           CONFIRM=confc1rmmb
           ACTION='CMD CLRPFM FILE(&lib./&file.) MBR(&mbr.)'
```

.

```
Clear...
.*
:PDFLDC   OPTION=90
           ACTFOR=PANEL C
           HELP='mbarmbr/exit'
           ACTION='EXIT SET'
           VARUPD=NO
           .
```

```
Exit
:PDACCEL.F3 3
```

```

:EPDFLD.
:EMBARC.
.*
:MBARC    HELP='mbarmbr/view'
.
View
:PDFLD.
:PDFLDC   OPTION=1
          ACTFOR=PANEL
          HELP='mbarmbr/view2'
          AVAIL=mbrview1
          ACTION=CHGVIEW
.
Names only
:PDACCEL.F11
:PDFLDC   OPTION=2
          ACTFOR=PANEL
          HELP='mbarmbr/view1'
          AVAIL=mbrview2
          ACTION=CHGVIEW
.
Names and descriptions
:PDACCEL.F11
:EPDFLD.
:EMBARC.
.*
:MBARC    HELP='mbarmbr/help'
.
Help
:PDFLD.
:PDFLDC   OPTION=1
          ACTFOR=PANEL
          HELP='mbarmbr/helphelp'
          ACTION=helphelp
.
Help for help...
:PDFLDC   OPTION=2
          ACTFOR=PANEL
          HELP='mbarmbr/exthelp'
          ACTION=exthelp
.
Extended help...
:PDFLDC   OPTION=3
          ACTFOR=PANEL
          HELP='mbarmbr/keyshep'
          ACTION=keyshep
.
Keys help...
:PDFLDC   OPTION=4
          ACTFOR=PANEL
          HELP='mbarmbr/about'
          ACTION='DSPHELP about'
.
About...
:EPDFLD.
:EMBARC.
:EMBAR.
.
.
.
The first KEYL tags in member T0011PN2 are not shown here
.
.
.*
.*
.* -----

```

```

.* Define keys for work with members panel with a menu bar
.* The keys are the same but some do not have descriptions
.* -----
:KEYL    NAME=mbrkeysmb
        HELP=key1.
:KEYI    KEY=F1
        HELP=helpf1
        ACTION=HELP.
:KEYI    KEY=F3
        HELP=exit
        ACTION='EXIT SET'
        VARUPD=NO.
F3=Exit
:KEYI    KEY=F4
        HELP=prompt
        ACTION=PROMPT
        PRIORITY=30.
F4=Prompt
:KEYI    KEY=F9
        HELP=retrieve
        ACTION=RETRIEVE
        PRIORITY=35.
F9=Retrieve
:KEYI    KEY=F10
        HELP=actions
        ACTION=ACTIONS
        PRIORITY=40.
F10=Actions
:KEYI    KEY=F11
        HELP=mbrviewname
        ACTION=CHGVIEW
        COND=mbrview1.
:KEYI    KEY=F11
        HELP=mbrviewdesc
        ACTION=CHGVIEW
        COND=mbrview2.
:KEYI    KEY=F12
        HELP=cancel
        ACTION='CANCEL SET'
        VARUPD=NO.
F12=Cancel
:KEYI    KEY=F24
        HELP=morekeys
        ACTION=MOREKEYS.
F24=More keys
:KEYI    KEY=ENTER
        HELP=enter
        ACTION=ENTER.
:KEYI    KEY=HELP
        HELP=help
        ACTION=HELP.
:KEYI    KEY=PAGEDOWN
        HELP=pagedown
        ACTION=PAGEDOWN.
:KEYI    KEY=PAGEUP
        HELP=pageup
        ACTION=PAGEUP.
:KEYI    KEY=PRINT
        HELP=print
        ACTION=PRINT.
:EKEYL.
:
Additional KEYL tags in member T0011PN2 are not shown here
:

```



```

:
:
: The first PANEL tags in member T0011PN2 are not shown here
:
:
.*
.*
.* -----
.* Define Work with Members panel with a menu bar to process options
.* -----
:PANEL      NAME=wrkmbmbar
            HELP='wrkmbmbr/'
            MBAR=mbarmbr  1
            KEYL=mbrkeysmb
            ENTER='RETURN 500'
            SELECT=PULLDOWN  A
            TOPSEP=SPACE.
Work with File Members
.*
.* -----
.* Define a data presentation area to display the
.* library/file name whose members are listed.
.* -----
:DATA      DEPTH=3
            SCROLL=NO
            LAYOUT=1
            BOTSEP=SPACE
            COMPACT
            .
.* -----
.* Divide the layout width into two columns.
.* The first column is for the prompt text with leader dots.
.* The second column is for the variable values.
:DATACOL   WIDTH=22.
:DATACOL   WIDTH=12.
:DATACOL   WIDTH='*'.
.* -----
.* Display qualified file name
:DATAGRP   GRPSEP=QINDENT
            HELP='wrkmbmbr/filelib'
            COMPACT
            .
:DATAI     VAR=file
            USAGE=INOUT
            CSRLOC=NO
            PROMPT='CALL exitpgm'
            .
File
:DATAC.F4  for list
:DATAI     VAR=lib
            USAGE=INOUT
            CSRLOC=NO
            .
Library
:DATAC.library, *CURLIB, *LIBL
:EDATAGRP.
.*
:EDATA.
.*
.* -----
.* Define the list area
.* -----
:LIST      DEPTH='*'
            LISTDEF=mbrlist
            ACTOR=UIM  D

```

```

MAXHEAD=4
MAXACTL=3
VIEW=mbrview
PARMS=parms
BOTSEP=SPACE.
:TOPINST.Select members using /, press Enter.
.*
.* -----
.* Specify the action to be taken for each option
.* -----
:LISTACT OPTION=3 E
NOCMD=PROMPT
HELP='wrkubr/cpyf'
ENTER='CMD CPYF ?*FROMFILE(&lib./&file.)'
ENTER=' ?*FROMMBR(&mbr.) &parms.'
PROMPT='CMD ?CPYF ?*FROMFILE(&lib./&file.)'
PROMPT=' ?*FROMMBR(&mbr.) &parms.'.

.*
:LISTACT OPTION=4
HELP='wrkubr/rvmv'
.*
CONFIRM=confrvmmb
ENTER='CMD RMVM FILE(&lib./&file.) MBR(&mbr.)'
PROMPT='CMD ?RMVM ?*FILE(&lib./&file.) ?*MBR(&mbr.)'
USREXIT='CALL exitpgm'.

.*
:LISTACT OPTION=5
COND=pf
HELP='wrkubr/dspfm'
ENTER='CMD DSPPFM FILE(&lib./&file.) MBR(&mbr.) &parms.'
PROMPT='CMD DSPPFM ?*FILE(&lib./&file.) ?*MBR(&mbr.) &parms.'.

.*
:LISTACT OPTION=7
COND=pf
HELP='wrkubr/rgzm'
ENTER='CMD RGZPFM FILE(&lib./&file.) MBR(&mbr.)'
PROMPT='CMD ?RGZPFM ?*FILE(&lib./&file.) ?*MBR(&mbr.)'.

.*
:LISTACT OPTION=8
HELP='wrkubr/dspfd'
ENTER='CALL exitpgm'
PROMPT='CALL exitpgm'.

.*
:LISTACT OPTION=9
COND=pf
HELP='wrkubr/clrm'
.*
CONFIRM=confcrlrmb
ENTER='CMD CLRPFM FILE(&lib./&file.) MBR(&mbr.)'
PROMPT='CMD ?CLRPFM ?*FILE(&lib./&file.) ?*MBR(&mbr.)'.

.*
.*
.* -----
.* Define the columns and headings to display
.* -----
:LISTCOL VAR=mopt
USAGE=INOUT
HELP='wrkubr/option'
MAXWIDTH=6.
:LISTCOL VAR=mbr
USAGE=OUT
HELP='wrkubr/mbr'
MAXWIDTH=10.
Member
:LISTCOL VAR=mtype
USAGE=OUT

```

```

                HELP='wrkabr/type'
                MAXWIDTH=10.
Type
:LISTCOL  VAR=mtxt
        USAGE=OUT
                HELP='wrkabr/text'
                MAXWIDTH='*'.
Text
.*
.* -----
.* Define multiple views for F11 to toggle between
.* -----
:LISTVIEW COLS='mopt mbr mtype mtxt'.
:LISTVIEW COLS='mopt mbr' layout=4.
.*
:ELIST.
.*
.* -----
.* Use a command line and allow parameters to be given
.* -----
:CMDLINE  SIZE=SHORT.
Parameters for options 3 and 5 or command
.*
:EPANEL.
:
Additional PANEL tags in member T0011PN2 are not shown here
:
.*
.* -----
.* End of panel group source
.* -----
:EPNLGRP.

```

メニュー・バー・パネルのアプリケーション・プログラミング

351 ページの図 111に示されたパネルを表示するアプリケーション・プログラムの例は、ライブラリー QUSRTOOL 内のソース・ファイル QATTSYSC にあるメンバー T0011CP2 に含まれています。これは適切な UIM アプリケーション・プログラミング・インターフェース (API) を呼び出してパネルを表示する、ILE C/C++ プログラムです。

アプリケーション・プログラムは、327 ページの『リスト・パネルのアプリケーション・プログラミング』に説明された処理を実行することになります。

RPG プログラムの一般例は、ライブラリー QUSRTOOL 内のソース・ファイル QATTINFO にあるメンバー T0011INF を参照してください。

第 17 章 ユーザー・インターフェース管理機能の詳しい使用方法

この章では、i5/OS ユーザー・インターフェース管理機能 (UIM) の構造について説明し、そのオブジェクトおよび機能に関する概要を示します。

UIM アプリケーションを開く

アプリケーション・プログラムは、パネル・グループを使用してダイアログ変数、リスト、およびパネル定義にアクセスする UIM アプリケーションを最初に開く必要があります。アプリケーションは、Open Display Application (QUIOPNDA) または Open Print Application (QUIOPNPA) API によって開かれ、Close Application (QUICLOA) API によって閉じられます。UIM アプリケーションは、開いたファイルに似た方法で管理されます。これはリソース再利用処理に対する特定の呼び出しについてのみ意味を持ち、ジョブの終わりにリソース再利用処理または終結処理によって必要に応じて閉じられます。

アプリケーションが開くと、UIM は各アプリケーションにアプリケーション・ハンドルを割り当てます。ハンドルは、オブジェクト、ある機能を使用するアプリケーションのインスタンス、または処理セッションを表現する変数です。このハンドルは、アプリケーションと共に機能するすべての API への入力パラメータとして指定する必要があります。同じパネル・グループ・オブジェクトを同じジョブ内の複数のオープン・アプリケーションに関連付けることができますが、各アプリケーションには UIM アプリケーションの状態を定義するダイアログ変数、アクティブ・リスト、および内部制御情報の完全に独立したセットが含まれます。

ダイアログ変数の定義

ダイアログ変数は、値を含む UIM エlement です。この値はオープン・アプリケーションを使用するプログラムによって、およびパネルをユーザーに表示するなどの機能を実行する際に UIM によって、参照および更新されることがあります。

オープン・アプリケーションに含まれるダイアログ変数は、アプリケーションに関連したパネル・グループ・オブジェクトによって決まります。ダイアログ変数は、パネル・グループのソースに含まれる変数定義 (VAR) 言語タグを使用して定義されます。UIM によって定義される特別なダイアログ変数には文字 Z で始まる名前があり、Z 変数と呼ばれます。VAR 言語タグについて詳しくは、692 ページの『VAR (変数定義)』を参照してください。Z 変数について詳しくは、693 ページの『UIM によって定義されるダイアログ変数』を参照してください。

特定のパネル・グループ・オブジェクトに関連したすべてのオープン・アプリケーションには、そのパネル・グループのタグ言語で定義された Z 変数を含む、パネル・グループに定義されたすべてのダイアログ変数の完全なセットが含まれます。アプリケーション内のすべてのダイアログ変数のセットは、そのアプリケーションの変数プールと呼ばれます。UIM は変数の値を取得するためにプログラムのストレージと直接通信することができません。プログラムは変数プールを使用して、パネルに何を表示するかについて UIM と通信できます。パネルが表示されるとき、UIM は変数プールから値を検索して、それらをパネルに表示します。ユーザーがパネル上のフィールドを更新すると、UIM はそれらの値をパネルから読み取り変数プールに入れて、ユーザー・プログラムが検索して使用できるようにします。各オープン・アプリケーションには独自の変数プールがあるので、ユーザー・プログラムがアプリケーション内のダイアログ変数にアクセスするたびに、QUIOPNDA API によって割り当てられたアプリケーション・ハンドルを変数プール API に提示する必要があります。

変数プールについて詳しくは、368 ページの『変数プール・サービスの使用方法』を参照してください。ダイアログ変数は、固定テキストではないパネル・フィールド値を表示するために使用します。ダイアログ変数は、条件定義 (COND) タグを使用してパネルのフォーマットを調整するためにも使用します。COND タグについて詳しくは、540 ページの『COND (条件定義)』を参照してください。

各ダイアログ変数の定義は、変数の内部ストレージの形式および画面上での編集特性を制御する、基本データ・タイプ値を指定します。以下の基本タイプの値がサポートされます。

- 文字
- IGC
- グラフィック
- バイナリー (数値)
- パックおよびゾーン 10 進数
- 日付
- 時間
- アクション (リスト・オプションまたは選択項目)
- 名前
- オブジェクト名
- ポインター

クラス定義 (CLASS) 言語タグは、基本データ・タイプ、特定の妥当性検査、およびディスプレイ値変換機能に関連したダイアログ変数のクラスを定義します。妥当性検査は、ユーザーが画面上の入力フィールドに入力した値に対してのみ行われます。妥当性検査は、API を介してアプリケーション・プログラムにより提供された値に対しては行われません。CLASS タグについて詳しくは、525 ページの『CLASS (クラス定義)』を参照してください。

変換機能により、アプリケーション・プログラムは内部値を使用して作動できるようになります。この値は指定の文字ストリング値との間で UIM により自動的にマップされてから、表示または印刷されます。例として、暦の月があります。アプリケーション・プログラムは 1 から 12 までの値を参照しますが、画面上でユーザーに表示されるのは月の名前だけです。変換機能について詳しくは、686 ページの『TL (変換リスト)』を参照してください。

アプリケーションが最初に開かれるとき、アプリケーション内のすべてのダイアログ変数には初期値があります。アプリケーション・プログラマーによって定義されたダイアログ変数では、初期値は基本タイプに応じて以下のように定義されます。

表 30. ダイアログ変数の初期値

ダイアログ変数	初期値
数値	0
日時	0
文字	1 バイトのブランク
IGC	1 バイトのブランク
グラフィック	2 バイトのブランク
ポインター	NULL
Z 変数	変数の定義に応じて定義される (365 ページの『ダイアログ変数の定義』を参照)

アプリケーション・プログラムによる参照でも UIM による参照でも、設定された値なしで参照されるダイアログ変数には、初期値が割り当てられます。

ダイアログ変数の使用上の制約事項

結果が望ましくないものになることがあるので、ダイアログ変数は、複数の入力フィールドとして 1 つのパネルにつき複数の表示項目がある値には使用しないでください。つまり、入力フィールドとして使用するとき、同じパネル定義の一部である複数のデータ項目 (DATAI)、データ項目エクステンダー (DATAIX)、または リスト列 (LISTCOL) タグにダイアログ変数を条件付けなしで指定しないということです。これにより、同じダイアログ変数が複数の場所で画面に表示されないことが保証されます。

アプリケーションが閉じるまで変数プールが存在するので、UIM およびアプリケーションを使用するすべてのプログラムは、プール内の任意のダイアログ変数に割り当てられた最終値を決定できます。アプリケーション・プログラムがダイアログ変数の値を割り当てた後、変更の必要がなければ、再び割り当てる必要はありません。同様に、入力値がユーザーから受け入れられてダイアログ変数に保管されると、アプリケーション・プログラムまたはユーザーによって更新されるまで、その値は引き続き使用可能です。

ダイアログ変数のエラー・メッセージ

基本タイプ char 1 の変数を VAR タグの ERRVAR 属性に定義することにより、ユーザーは Put Dialog Variable (QUIPUTV) API を使用してこの変数を (X『F1』) に設定し、UIM に変数がエラーであることを通知できます。これを行うと、エラーのフィールドは強調表示されて、カーソルはエラーとなっている最初の入力フィールドに配置されます。アプリケーション・プログラムはエラーの原因を報告し、UIM が Display Panel (QUIDSPP) API のメッセージ参照キー・パラメーターを使用してユーザーにメッセージを表示するように要求します。

オープン・アプリケーション内にあるエラーの変数はすべて、次のダイアログ・コマンドが UIM によって処理されるときにリセットされます。ただし、メニュー・バー・カーソル操作 (ACTIONS)、コマンド行 (CMDLINE)、ビューの変更 (CHGVIEW)、HELP、追加の機能キーの表示 (MOREKEYS)、先頭に移動 (MOVETOP)、PAGEUP、PAGEDOWN、画面の印刷 (PRINT)、およびコマンド・ストリングの取得 (RETRIEVE) の各ダイアログ・コマンドは例外です。各ダイアログ変数のエラー状況は、制御が Display Panel (QUIDSPP) API から戻されるとき、または制御が UIM からメニュー項目、処置リスト、プルダウン選択、または機能キー項目によって示されるプログラムまたはコマンドに渡されるときにリセットされます。

VARUPD=YES が指定されている操作を実行する機能キーが押されたかまたはプルダウン選択が選択されたとき (593 ページの『KEYI (キー・リスト項目)』および 660 ページの『PDFLDC (プルダウン・フィールド選択項目)』を参照)、UIM は入力 (変更) されたすべての入力フィールドを妥当性検査します。エラーが検出された場合、パネルはエラーのフィールドが強調表示されて再表示され、1 つ以上のメッセージが発行されます。入力フィールドに関連したダイアログ変数の値は、ユーザーが正しい値を入力するまで変更されません。

アプリケーション開発者は VARUPD=NO を使用することにより、いくつかの機能キーおよびプルダウン選択が作動する際に変数プールを更新しないように指定できます。そのような機能に対する QUIPSPP API を実行した制御がプログラムに戻される場合、ユーザーが入力したすべての値は失われます。要求した機能により UIM がプログラムまたはシステム・コマンドを呼び出す場合、すべての入力フィールド値は、制御が UIM に戻ってパネルが再表示されるときに表示可能となるような方法で UIM によって内部的に保存されます。これらの入力値は、表示中のパネルのためにローカルに保存されます。VARUPD=NO が指定されたプログラムまたはコマンドにより、同じオープン・アプリケーションを使用して表示されるパネルは、

すべてのダイアログ変数の値を変数プールに存在するままの値で表示します。ただし、VARUPD=YES が指定されている場合、制御が戻されて元のパネルが再表示される時、保存された値ではなく更新された値が使用されます。

ダイアログ変数を使用する表示パネルへのフィールド値の提供

タグ言語により、ダイアログ変数を使用して、プルダウン選択、メニュー選択、機能キー、または処置リスト選択に対して UIM によって呼び出された表示パネルのフィールド値および CL コマンドの置換値を提供できます。ダイアログ変数がこれらの方法の 1 つで呼び出される時は常に、変数プールに保存された内部値は、変数基本タイプの編集規則および変数クラスに定義された変換リストによって決められる外部形式に変換されます。

外部形式のダイアログ変数内に表示装置で不正な文字が含まれる場合、UIM はその不正文字を 16 進 1F 文字 (■) に変換します。外部形式のダイアログ変数内にプリンターで不正な文字が含まれる場合、UIM はその不正文字をプリンター・ファイルで使用可能な置換文字に変換するか、または RPLUNPRT(*NO) がプリンター・ファイルに対して有効な場合にはブランクに変換します。

不正な表示文字は、16 進 FF と等しいかまたは 40 未満の 16 進値の文字で、16 進 00、0E、および 0F を含みません。不正なプリンター文字は、不正な表示文字と同じですが、ただし 16 進 00 も不正となります。この変換は、変換リストを使用して変換された変数値では生じません。

不正文字を含むダイアログ変数が入力フィールドとして使用され、プログラムにより Get Dialog Variable (QUIGETV) API を使用して検索される場合、その不正文字は 16 進 3F に変更されることがあります。これは、ユーザーがそのフィールドを変更するか、またはヘルプ・キーを押す場合に生じます。

UIM はパネルを表示するとき、すべての変数フィールドの現行値を変数プールまたはリスト項目から検索します。ユーザーが値を入力フィールドに入力した場合、UIM はその値を妥当性検査して、関連したダイアログ変数の CLASS 属性に基づいて変換してから、その値をダイアログ変数または関連したリスト項目に保存します。アプリケーション・プログラムは、Get Dialog Variable (QUIGETV) または Get List Entry (QUIGETLE) API を使用してダイアログ変数値を検索することにより、ユーザーによって入力された値を判別できます。

変数プール・サービスの使用方法

Get Dialog Variable (QUIGETV) および Put Dialog Variable (QUIPUTV) API は、アプリケーション・プログラムがダイアログ変数を検索および更新するための基本インターフェースです。ダイアログ変数が更新される時、新しい値の属性がそのダイアログ変数の基本タイプと一致する必要があります。ダイアログ変数が検索される時、値はダイアログ変数の基本タイプによって指定された内部形式で常にアプリケーション・プログラムに戻されます。変数プール・サービスの API を呼び出すとき、変数レコードが必要となります。UIM パネル・グループ・ソース内の VARRCD タグを使用して定義された変数レコードは、API に対する 1 回の呼び出しでまとめて更新またはアクセスされる、ダイアログ変数のグループを定義します。

QUIGETV および QUIPUTV API を使用して変数値を検索または更新するとき、データ変換または妥当性検査は行われません。これらの API について詳しくは、『API』のトピックを参照してください。

Z 変数と呼ばれる、UIM によって定義されたいくつかのダイアログ変数は、検索できますが変更はできません。タグ言語では、読み取りアクセスだけを持つ変数をパネル定義で入力フィールドに使用することはできません。アプリケーション・プログラムが QUIPUTV API を使用してそのような Z 変数を更新しようとすると、UIM によって例外が戻されます。

ダイアログ変数および特殊値

TIME の BASETYPE が ZONE オプションと共に定義された変数は、タイム・ゾーンに対する特殊値を持つことがあります。詳しくは、525 ページの『CLASS (クラス定義)』 タグを参照してください。ディスプレイ装置を使用中のユーザーまたはアプリケーションのどちらかが、特殊値を指定できます。値が変数プールに入れられると、UIM はこれらの特殊値を解決します。そのため、変数の値を検索する次の操作では、その変数には入力されたままの特殊値ではなく解決された値が含まれています。

そのため、ディスプレイ装置を使用中のユーザーは値「01:30:00 *LCL」をデータ項目に入力できます。パネル上の別のデータ項目が同じ変数を表示するとき、またはアプリケーションがダイアログ変数の値を検索するときは、その値は「01:30:00 CST」のように表示されます。同様に、アプリケーションが特殊値のあるダイアログ変数を更新する場合、それを再び取得すると値が変更されていることがあります。アプリケーションで特殊なタイム・ゾーン値を使用する場合には、この性質に注意してください。

文字セットおよびコード・ページについての考慮事項

常に特定の文字セットおよびコード・ページに保存されているデータは、UIM によって表示または印刷するとき、別の文字セットおよびコードに変換できます。アプリケーションに変換の必要なデータがある場合、以下の 1 つを行ってください。

- 変換の必要な変数の CLASS タグに、CHRID=PNLGRP をコーディングします。525 ページの『CLASS (クラス定義)』を参照してください。その後、データの図形文字セットおよびコード・ページの数、パネルの作成に使用したパネル・グループの作成 (CRTPNLGRP) コマンドの CHRID パラメーターに指定します。CRTPNLGRP コマンドについて詳しくは、iSeries Information Center の『CL』のトピックを参照してください。
- パネルの作成に使用したパネル・グループの作成 (CRTPNLGRP) コマンドの CHRID パラメーターに、*JOBCCSID を指定します。CRTPNLGRP コマンドについて詳しくは、iSeries Information Center の『CL』のトピックを参照してください。

表示

UIM は、装置の文字セットおよびコード・ページをパネル・グループの文字セットおよびコード・ページと比較します。それらが異なる場合、アウトバウンドおよびインバウンド変換テーブルを使用して適切なダイアログ変数を変換します。変換テーブルを使用できない場合、Open Display Application (QUIOPNDA) API は診断メッセージを送り、続行します。

装置がコード・ページ 420 または 424 を使用するように構成されている場合のみ、パネル・グループ (PNLGRP) タグに BIDI=LTR または BIDI=RTL が指定されたアラビア語およびヘブライ語の両方向パネル・グループの表示操作が許可されます。

表 31を参照してください。

表 31. 表示用の UIM CCSID/CHRID 変換

データのタイプ	パネル・グループ CHRID		
	XXX	*JOBCCSID	*DEVD
パネル・グループの固定テキスト	変換なし。	パネル・グループの 1 次ソース・ファイル CCSID から装置 CCSID への変換。	変換なし。
CLASS タグに CHRID=PNLGRP が指定された変数	XXX から装置 CHRID への変換。	ジョブ CCSID から装置 CCSID への変換。	変換なし。

表 31. 表示用の UIM CCSID/CHRID 変換 (続き)

データのタイプ	パネル・グループ CHRID		
	XXX	*JOBCCSID	*DEV D
CLASS タグに CHRID=PNLGRP が指定さ れていない変数	変換なし。	ジョブ CCSID から装置 CCSID への変換。	変換なし。

印刷

UIM は、プリンター・ファイルの文字セットおよびコード・ページをパネル・グループの文字セットおよびコード・ページと比較します。それらが異なる場合、アウトバウンド変換テーブルを使用して適切なダイアログ変数を変換します。パネル・グループまたはプリンター・ファイルのどちらかに CHRID(*DEV D) が指定されている場合、ダイアログ変数の変換は行われません。変換テーブルを使用できない場合、Open Display Application (QUIOPNDA) または Add Print Application (QUIADDPA) API は診断メッセージを送り、続行します。

プリンター・ファイルが印刷装置上で印刷される場合、プリンター・ファイルの文字セットおよびコード・ページと、プリンターにロードされた文字セットおよびコード・ページとが比較されます。それらが異なる場合、UIM タグからの固定テキストを含むすべてのプリンター・ファイル・データは、プリンターの文字セットおよびコード・ページに変換されます。

文字セットおよびコード・ページの変換が行われる回数を最小にするために、プリンター・ファイルおよびパネル・グループに同じ CHRID 値を指定してください。

パネル・グループ (PNLGRP) タグに BIDI=LTR または BIDI=RTL が指定されたアラビア語およびヘブライ語の両方向パネル・グループの印刷操作では、プリンター・ファイルにコード・ページ 420 または 424 が指定されている必要があります。さらに、両方向パネル・グループの QUIADDPA API に対する呼び出しでは、表示装置で使用されるものと同じコード・ページがプリンター・ファイルに指定されている必要があります。

表 32を参照してください。

表 32. 印刷用の UIM CCSID/CHRID 変換

印刷ファイル CHRID	パネル・グループ CHRID またはメニュー CHRID		
	XXX	*JOBCCSID	*DEV D
YYY	パネル・グループの固定テキスト: 変換なし。	パネル・グループの固定テキスト: パネル・グループの 1 次ソース・ファイル CCSID から YYY への変換。	パネル・グループの固定テキスト: 変換なし。
	CLASS タグに CHRID=PNLGRP が指定された変数: XXX から YYY への変換。	CLASS タグに CHRID=PNLGRP が指定された変数: ジョブ CCSID から YYY への変換。	CLASS タグに CHRID=PNLGRP が指定された変数: 変換なし。
	CLASS タグに CHRID=PNLGRP が指定されていない変数: 変換なし。	CLASS タグに CHRID=PNLGRP が指定されていない変数: ジョブ CCSID から YYY への変換。	CLASS タグに CHRID=PNLGRP が指定されていない変数: 変換なし。

表 32. 印刷用の UIM CCSID/CHRID 変換 (続き)

印刷ファイル CHRID	パネル・グループ CHRID またはメニュー CHRID		
	XXX	*JOBCCSID	*DEVD
*JOBCCSID	パネル・グループの固定テキスト: 変換なし。	パネル・グループの固定テキスト: パネル・グループの 1 次ソース・ファイル CCSID から ジョブ CCSID への変換。	パネル・グループの固定テキスト: パネル・グループの 1 次ソース・ファイル CCSID から ジョブ CCSID への変換。
	CLASS タグに CHRID=PNLGRP が指定された変数: XXX から ジョブ CCSID への変換。	CLASS タグに CHRID=PNLGRP が指定された変数: 変換なし。	CLASS タグに CHRID=PNLGRP が指定された変数: 変換なし。
	CLASS タグに CHRID=PNLGRP が指定されていない変数: 変換なし。	CLASS タグに CHRID=PNLGRP が指定されていない変数: 変換なし。	CLASS タグに CHRID=PNLGRP が指定されていない変数: 変換なし。
*DEVD	パネル・グループの固定テキスト: 変換なし。	パネル・グループの固定テキスト: 変数はジョブ CCSID にあり、装置 CCSID は不明であるため、パネル・グループの 1 次ソース・ファイル CCSID から ジョブ CCSID への変換。	パネル・グループの固定テキスト: 変換なし。
	CLASS タグに CHRID=PNLGRP が指定された変数: 変換なし。	CLASS タグに CHRID=PNLGRP が指定された変数: 変換なし。	CLASS タグに CHRID=PNLGRP が指定された変数: 変換なし。
	CLASS タグに CHRID=PNLGRP が指定されていない変数: 変換なし。	CLASS タグに CHRID=PNLGRP が指定されていない変数: 変換なし。	CLASS タグに CHRID=PNLGRP が指定されていない変数: 変換なし。

リストの管理

UIM リストは、リスト項目の順次セットです。各項目には、1 つ以上のダイアログ変数の値のコピーが含まれます。リスト内の項目は、ユーザーに対して画面のスクロール可能域に表示できます。UIM は、アプリケーション・プログラムがリスト上の以下の操作を実行できるようにする、アプリケーション・プログラミング・インターフェース (API) を提供します。

- リスト内の任意の 2 つの項目間に新規の項目を追加する。
- リストの最初または最後に項目を追加する。
- リスト項目内の値を更新する。
- リスト項目を除去する。
- リストの現行の項目ポインターを特定の項目に対して配置する。いくつかの配置オプションが使用可能です。
- リストが表示されるときに UIM 処理を制御するリスト属性を設定および取得する。
- アクティブ・リストを削除する。(リストからすべての項目を除去して、そのリストをアプリケーションに対して非アクティブにする。)

リストの定義

リスト定義 (LISTDEF) 言語タグを使用する各リストの定義は、パネル・グループ・オブジェクトのソースに指定されます。リスト定義 (LISTDEF) タグの属性は、リストの名前、およびどのダイアログ変数をリストに関連付ける必要があるか (つまり、どの変数値を各リスト項目に保存するか) を指定します。

LISTDEF タグについて詳しくは、625 ページの『LISTDEF (リスト定義)』を参照してください。

リスト内の各項目には、そのリストに関連したすべてのダイアログ変数の値のコピーが含まれます。ダイアログ変数の値は、Add List Entry (QUIADDLE) または Add List Multiple Entries (QUIADDLM) API を使用して項目を最初にリストに追加するとき、および Update List Entry (QUIUPDLE) API を使用してその項目を更新するときに、リスト項目にコピーされます。リスト項目内の値は、Get List Entry (QUIGETLE) または Get List Multiple Entries (QUIGETLM) API を使用して項目を検索するときに、対応するダイアログ変数にコピーされます。

リストの初期化

リストを処理するためのすべての UIM API は、操作のターゲットとなるリストを識別するために、リスト・オブジェクト定義 (LISTDEF) タグにリスト名を指定するプログラムを必要とします。パネル・グループに定義された各リストは、そのパネル・グループを使用するオープン・アプリケーションごとに、アクティブまたは非アクティブのどちらかです。アプリケーションが Open Display Application (QUIOPNDA) または Open Print Application (QUIOPNPA) API によって開かれるとき、各リストは最初は非アクティブです。Delete List (QUIDLTL) API を使用してリストを削除すると、そのリストは非アクティブになります。同じパネル・グループを使用する複数のオープン・アプリケーションでは、1 つのリストがアクティブであることができます。Add List Entry (QUIADDLE) または Add List Multiple Entries (QUIADDLM) API を使用して最初の項目をリストに追加したとき、あるいは Set List Attribute (QUISETLA) API を使用してリスト項目を設定したとき、リストはアクティブになります。

1 リストの最大サイズは、約 16MB です。リストに追加可能な項目の最大数は、各項目のサイズに基づきます。以下の計算を使用して、リストのサイズを推定できます。

1 1. 以下の値を加算します。

- 1 • 項目のサイズ
- 1 • 項目ごとに、19 バイトのオーバーヘッド値
- 1 • 項目ごとに、5 から 15 バイトの変数オーバーヘッド

1 2. 16MB (16 777 216 バイト) から 4096 バイト (i5/OS でのオブジェクトのヘッダー) を差し引いた値を、結果の値で割ります。

1 たとえば、各リスト項目に 94 バイトの値を定義した場合、以下の計算を行います。

1 $(16\,777\,216 - 4096) / (94 \text{ バイト} + 19 \text{ バイト} + 5 \text{ バイト}) =$

1 $16\,773\,120 / 118 =$

1 142,145 項目

リストの表示

リスト域を含むパネルが表示されるとき、リストからの項目を使用してユーザーに表として表示される表示域が作成されます。各リスト域は、1 つのリスト定義に関連付けられ (603 ページの『LIST (リスト域)』を参照)、表示するための独立した列のセレクションを含みます。LISTCOL 言語タグについて詳しくは、621 ページの『LISTCOL (リスト列)』を参照してください。1 つのリストを、複数のリスト域が参照することもできます。リスト域により、1 つのリストを同じオープン・アプリケーション内のさまざまなパネルで表示できます。1 つのパネル定義に複数のリスト域が存在する場合、リスト域ごとに異なるリストを使用する必要があります。

LISTCOL タグを使用して、リスト域に表示される特定の値を指定します。リスト域は、関連するリスト項目に含まれるすべての値を表示することも、各リスト項目で使用可能な値のサブセットだけを表示することもできます。ユーザーに表示されるリスト域には、リスト項目ごとに 1 つの行、およびリスト域の各ビューの定義にある LISTCOL タグに指定されたフィールドごとに 1 つの列が含まれます。

リスト域で、表示行ごとに 1 つだけのリスト項目に関連した値が表示されるように定義することも、表示行ごとに複数のリスト項目に関連した値が表示されるように定義することもできます。表示行ごとに複数の項目が表示される時、画面は複数のレイアウトにフォーマット設定され、レイアウト列内では上から下に、そしてレイアウト間では左から右に、項目が表示されます。610 ページの『例 2: 3 つのレイアウト列を持つリスト域』を参照してください。

リストの更新

リスト項目の値は、画面にあるリスト域のすべてのフィールドに値を入れるために使用します。621 ページの『LISTCOL (リスト列)』を参照してください。ユーザーは、リスト域にある入力フィールドも含む、画面上の任意の入力フィールドを更新できます。ユーザーが指定した値が関連するダイアログ変数に対するすべての妥当性検査に適合する場合、リスト項目内の対応する値はユーザーが指定した値に更新されます。

リスト項目を検索、追加、および更新するために提供されている UIM API はすべて、リストに関連したダイアログ変数を使用して機能します。ダイアログ変数を参照しないでリスト項目の値を直接設定または変更することは、サポートされていません。リスト項目が検索、追加、または更新されるたびに、ダイアログ変数のすべての値のセットがリスト項目との間でコピーされます。処置リスト内のユーザー・オプションを処理するときなど、UIM がリスト項目を参照または更新するとき、リスト列に対応するダイアログ変数の値も更新できます。

注: 処置リストの作業を行うとき、不完全リストを処理する際には注意してください。望ましくない結果を防止するため、アクション・ダイアログ変数をリスト項目の VARRCD に組み込んで、それを変数レコード内の他の変数と同じ方法で更新してください。これを行わない場合、それぞれの新規リスト項目を、(不完全リストを処理する際に) パネルに最後に入力されたオプションの値で更新する危険があります。たとえば、12 の項目があるリスト・パネルを想定します。1 つの項目にオプション『4』を入力してスクロールダウンします。リストにさらに項目を追加するため、不完全リスト出口が呼び出されます。オプションのダイアログ変数は『4』の値を持つので、追加される各項目はオプション・フィールドに『4』の値を持つことになります。

リストに挿入される項目ごとに、項目がリストから除去されるまでアクティブ・リスト内でその項目を固有に識別する、**リスト項目ハンドル**と呼ばれる ID が割り当てられます。この ID は、オープン・アプリケーション、アクティブ・リスト、およびリスト項目のインスタンスの特定の組み合わせに対してのみ意味を持ちます。他のオープン UIM アプリケーション内では、または同じアプリケーション内でもリストまたは項目が削除された後に再作成された場合には、この ID は意味を持ちません。ID をこの定義に含まれない用途に使用した場合、望ましくない結果が生じる可能性があります。

不完全リストの処理

不完全リスト処理の際に項目を処置リストに追加する場合、QUIADDLE または QUIADDLM API が呼び出されるときにオプション・フィールドに適切な値 (通常はブランク) が含まれているように注意してください。VARRCD を使用する場合、オプション変数は VARRCD の一部であることが必要です。

リストからの項目の除去および挿入

項目は、リストから除去されると、アプリケーション・プログラムによってアクセスできなくなり、パネルのリスト域にはブランク行としてさえ示されません。新規の項目がリストに追加されると、UIM はその項

目に、以前にリストから除去された項目に対して使用されていたものと同じリスト項目ハンドルを割り当てることができます。ID 値は、リスト内の項目に対して定義された順序内での項目の論理位置とは関係がありません。

UIM は、リストと項目との間の操作のための参照点として使用するアクティブ・リストごとに、現行の項目ポインターを維持します。リスト項目操作の API は、リスト項目ハンドル・パラメーターをサポートします。このパラメーターはアプリケーション・プログラムに、操作の最後に現行の項目ポインターが配置された場所を示す ID を戻します。UIM は、現行の項目ポイントを以下に設定します。

- Add List Entry (QUIADDLE) または Add list Multiple Entries (QUIADDLM) API によって追加されたばかりの項目。
- Get List Entry (QUIGETLE) または Get List Multiple Entries (QUIGETLM) API によって要求されたリスト項目。ただし、要求された項目が見つからないか使用不可の場合には変更されません。
- Remove List Entry (QUIRMVLE) API によって除去された項目の直前にあった項目。

現行の項目ポイントは、Update List Entry (QUIUPDLE) API によって変更されません。

Get List Entry (QUIGETLE)、Get List Multiple Entries (QUIGETLM)、および Remove List Entry (QUIRMVLE) API は、現行の項目ポインターの位置をリストの最上部または最下部のどちらかに設定できます。最上部は常にリストの最初の項目よりも論理的に前の位置で、最下部は常に論理的にリストの最後の項目よりも論理的に後の位置です。これらの位置にはそれぞれリスト項目ハンドルのための特別な ID 値がありますが、最上部および最下部は「本当の」項目ではないため、リストから更新または除去することはできません。アプリケーション・プログラムは、リストの最上部よりも前または最下部よりも後に項目を挿入しようとする、エラーを受け取ります。

リスト画面上のリスト項目の制御

アプリケーション・プログラムは、Display Panel (QUIDSPP) API が実行する前に表示位置属性を有効なリスト項目ハンドルに設定することにより、パネルのリスト域の最上位の項目としてどのリスト項目を表示するかを制御できます。たとえば、この機能を使用してユーザーが指定する特定のリスト項目に位置決めすることができます。リストが初期化されたときからリストが最初に表示されるときまでの間に表示位置属性が Set List Attributes (QUISETLA) API によって設定されない場合、UIM は最初の項目をパネルのリスト域の最上部に表示します。Display Panel (QUIDSPP) API について詳しくは、『API』のトピックを参照してください。

表示位置属性は、ユーザーがリストの新しいページに移動するときに常に更新されます。アプリケーション・プログラムが表示位置属性によって識別されていた項目を除去するときには常に、表示位置属性が自動的に更新されて、除去された項目の直前にある項目を参照するようになります。除去された項目の前に項目が存在しない場合、表示位置属性はリストの最上部に設定されて、パネルが次に表示されるときにその最初の項目がパネルのリスト域の最上部に表示されます。

リスト内で項目が追加または更新される時、リストに関連したすべてのダイアログ変数のエラー状態がそのリスト項目と共に保存されます。エラー状態のダイアログ変数について 365 ページの『ダイアログ変数の定義』で説明されている処理は、エラー状態のリスト項目値についても使用されます。この処理は、妥当性検査エラーが UIM によって検出される方法、およびキー項目 (KEYI) タグとプルダウン・フィールド選択項目 (PDFLDC) タグの VARUPD 属性がリスト値の更新に影響を与える方法に関係しています。

リスト画面の対話応答時間の改善

リスト画面の対話応答を改善するために、UIM リスト処理は、アプリケーション・プログラム (不完全リストの出口プログラム) がリストの一部を作成しただけでそのリストを表示できるようにします。UIM は

Display Panel (QUIDSPP) API を実行したプログラムに戻らずに、リストのスクロールをパネルの表示の一部として扱うので、アプリケーション・プログラムは、リスト内の項目が完全なリストの最上部、中央、または最下部のいずれの部分を表すかを UIM に示す必要があります。プログラムが Set List Attribute (QUISETLA) API を呼び出してリスト内容属性を ALL 以外の値に設定した場合で、ユーザーがアプリケーションでは使用可能でもまだリストに追加されていないリスト項目をページ送りして表示しようとした場合には、UIM はアプリケーション・プログラムを呼び出してさらにリスト項目を追加します。また、指定された項目が見つからないものの、まだ追加されていないリストの部分にそれが存在する可能性がある場合にも、アプリケーション・プログラムは Get List Entry (QUIGETLE) または Get List Multiple Entries (QUIGETLM) API 処理の一部として、UIM によって呼び出されます。

リスト域を含むパネルが表示されて、関連したリストがアクティブでないかまたは項目を含まない場合、ユーザーにはブランクのリスト域と、リストに項目がないことを示すメッセージとが表示されます。ただし、リストが完全ではないことを示すようにリスト属性が設定されている場合、アプリケーション・プログラムが呼び出されて項目がリストに追加されます。表示ページの全体を表示するために十分な項目が追加されるか、またはアプリケーション・プログラムが最上部または最下部のどちらかでリストを完全であるとマーク付けするまで、パネルはユーザーに表示されません。アプリケーション・プログラムは、リストが空のときだけではなく、不完全リスト内にパネルのリスト域を満たすだけの項目がないときにも呼び出されます。

処置リストおよび選択リストの使用

UIM は以下のタイプのリストをサポートします。

- 処置リスト
- 選択リスト
- 出力専用リスト
- 入出力リスト

処置リストの使用

処置リストは、ユーザーがオプション番号を入力することによりリスト内の 1 つ以上の項目に対して操作を実行できるリスト域です。処置リストとして表示される UIM リストの定義には、CLASS タグに BASETYPE=ACTION が指定された変数が含まれる必要があります。この変数は、リストのアクション変数と呼ばれます。

処置リスト域を定義するには、アプリケーション開発者は LIST タグに ACTOR=UIM または ACTOR=CALLER を指定する必要があります。さらに、ユーザーが入力可能なオプション番号ごとに 1 つの LISTACT タグが定義されている必要があります。

ACTOR=UIM 処置リストでは、各 LISTACT タグはユーザーがオプション番号を入力して Enter キーを押したときに UIM が実行する操作を指定します。アプリケーション開発者は、ユーザーが PROMPT ダイアログ・コマンドに割り当てられたキーを押したときに UIM が実行する操作を定義することもできます。CL コマンドを実行するか、またはプログラムを呼び出すという、2 つのタイプの操作を実行できます。実行する操作が CL コマンドの実行である場合、CL コマンドがオプションの機能を実行した後でリストを更新するために、処置リスト出口プログラムが必要となることがあります。UIM と CL コマンドとの間のインターフェースで、CL コマンドに呼び出されたプログラムがアプリケーション・ハンドルを取得することを許可するものではありません。そのため、CL コマンドに呼び出されたプログラムが変更または削除操作を実行した場合、リストの更新には処置リスト出口プログラムが必要となります。

ACTOR=CALLER 処置リストでは、アクションの指定はありません。ユーザーがオプション番号を入力して Enter キーまたは PROMPT ダイアログ・コマンドに割り当てられた機能キーを押した場合、UIM は制御

をアプリケーション・プログラムに戻します。機能要求パラメーターには、アプリケーション・プログラムがユーザーの選択したオプションに関連した操作を実行する必要があることを示す値が含まれます。

UIM を処置リストのアクターにすることは、以下の利点により推奨された方式となっています。

- UIM は、ユーザーが入力したすべての操作を実行します。
- UIM は、オブジェクトを削除するオプション 4 などの破壊アクションに対して自動的に確認パネルを表示します。確認サポートを提供するためにアプリケーション・プログラム・コードを記述する必要はありません。
- UIM は、操作が正常に終了しない場合に備えて、処置リスト・パネルを再表示します。UIM がエスケープ・メッセージを受け取ると、UIM はカーソルがそのオプション番号に配置されたリストを再表示します。オプション番号はエラーで示されます。リスト・オプションの結果として表示されたパネルから終了 (F3) または取り消し (F12) 機能が要求された場合、UIM は適宜パネルを再表示します。処置リストの表示中に UIM が終了および取り消しの要求を処理する方法については、388 ページの『リスト・パネルをまとめて閉じる』を参照してください。

アプリケーション・プログラムが処置リストのアクターである場合、上記の処理をアプリケーション・プログラムが行う必要があります。

パネルに処置リストおよびメニュー・バーが含まれる場合、処置リスト内で選択された各項目に対して機能するプルダウン選択をメニュー・バー内に定義できます。選択したリスト項目に対して機能するプルダウン選択が少なくとも 1 つ定義されているとき、UIM はユーザーが有効な選択文字を入力して項目を選択できるようにします。ユーザーはリスト内の 1 つ以上の項目を選択し、メニュー・バーからプルダウン・メニューを選択してから、プルダウン・メニューから選択項目を選択します。UIM は、ユーザーが選択した項目ごとに、選択されたプルダウン選択を実行します。

選択リストの使用

選択リストは、ユーザーがリスト内の 1 つ以上の項目の選択文字を入力できるリスト域です。選択リストは、ユーザーがリスト内の単一項目または複数項目を選択できるように定義されます。

選択リストは、以下の目的で使用できます。

- ユーザーが以前に表示された項目フィールドのオブジェクトまたは値を選択できるようにします。選択リストのタイプは、ユーザーがカーソルを項目フィールドに配置して PROMPT ダイアログ・コマンドを要求したときに表示されます。F4 キーは、PROMPT ダイアログ・コマンドに割り当てておくことをお勧めします。
- ユーザーが項目を選択できるようにしてから、メニュー・バーのプルダウン・メニューに定義された操作を選択できるようにします。

選択リストとして表示される UIM リストの定義には、CLASS タグに BASETYPE=ACTION が指定された変数が含まれる必要があります。この変数は、リストの**アクション変数**と呼ばれます。ユーザーがリスト項目に対して有効な選択文字を入力すると、UIM はその項目のアクション変数を 1000 に設定します。値が 1000 のアクション変数は、そのリスト項目が選択済みであることを常に示します。

パネルに選択リストおよびメニュー・バーが含まれる場合、そのリスト内で選択された各項目に対して機能するプルダウン選択をメニュー・バー内に定義できます。ユーザーはリスト内の 1 つ以上の項目を選択し、メニュー・バーからプルダウン・メニューを選択してから、プルダウン・メニューから選択項目を選択します。UIM は、ユーザーが選択した項目ごとに、選択されたプルダウン選択を実行します。

選択文字の使用

ユーザーは、スラッシュ (/) または国別の選択文字を、選択したいリスト項目のオプションまたは選択フィールドに入力して、メニュー・バーまたは選択リスト・パネルを含む処置リスト・パネル内の項目を選択します。国別の選択文字は、パネル・グループの作成時にパネル・グループごとに決定されます。UIM はパネル・グループのソース・ファイルの CCSID で、メッセージ CPX5A0C の第 1 レベル・テキストを取得します。このメッセージの最初の 2 つの文字は、パネル・グループ・オブジェクトに保存されます。これらの文字は、パネル・グループに定義された選択リストおよび複数項目選択フィールドのための、許可された大文字および小文字の国別の選択文字です。

パネル・グループの作成時に CHRID 属性として *JOBCCSID を指定した場合、国別の選択文字は実行時にパネル・グループのソース・ファイル CCSID からジョブ CCSID に変換されます。これにより、比較が同等の CCSID で行われます。

パネル機能の管理

UIM は、以下のパネル管理機能をサポートします。

- グラフィカル・ユーザー・インターフェース (GUI) へのパネル変換の使用可能化
- スクロール
- ユーザー機能の制限
- 文脈ヘルプの定義
- コマンド行の制約事項
- 機能キーの定義
- パネルのフォーマット設定
- 終了の要求時に複数のパネルをまとめて閉じる
- 複数のリスト・パネルをまとめて閉じる
- パネルの表示時に UIM を要求処理プログラムとする

UIM によって実行されるすべての操作は、経路指定ステップと呼ばれる単一のプロセスで行われます。アプリケーション開発者には、以下の機能ごとに、再帰、ロック、および静的ストレージなどの効果を検討する責務があります。

- 機能キー・アクション
- メニューまたは処置リスト・オプション
- プルダウン選択
- 出口プログラム
- ユーザーがコマンド行に入力したコマンド

UIM は、さまざまなオーバーライドや Set Attention Program (SETATNPGM) コマンドなどプログラムおよび CL コマンドの呼び出し反応効果が、前述のいずれかの方法でパネルからコマンドが実行されるときに保存されることを保証しません。そのコマンドを使用したプログラムが機能の有効範囲となるコマンドは、次のコマンドまたはプログラムが呼び出されるときにその効果を失うことがあります。これは、有効範囲となっていた機能呼び出しした UIM 呼び出しが一時的に消滅するためです。

GUI への変換の使用可能化

UIM は、クライアント・プログラムがパネルをグラフィカル・ユーザー・インターフェース (GUI) に変換できるようにする、パネル内の情報のエンコードをサポートします。

PANEL または PNLGRP タグの ENBGUI 属性を使用して、エンコードされた情報をパネルに含めるかどうかを指定できます。この属性の設定方法については、649 ページの『PANEL (表示パネル)』および 665 ページの『PNLGRP (パネル・グループ)』を参照してください。

スクロールのサポート

UIM は、すべてのパネル情報、データ表示、リスト、およびメニュー域に対して、ページアップおよびページダウンのスクロールをサポートします。複数のスクロール可能域を単一のパネル上に定義できます。UIM ではそれらの順序は問題になりません。ただし、スクロール可能なメニューの使用は、使いやすさの理由から勧められていません。

テキストを左から右にスクロールする機能は提供されていませんが、ビューの変更 (CHGVIEW) ダイアログ・コマンドをリストに対して使用して、より多くのフィールドを表示できます。

スクロール可能域の定義

UIM スクロール機能を使用するには、パネルに 1 つ以上のスクロール可能域があり、機能キーが PAGEUP および PAGEDOWN ダイアログ・コマンドを実行するように定義されている必要があります。PAGEUP および PAGEDOWN ダイアログ・コマンドについては、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

UIM では機能キーが自動的に使用可能にならないので、スクロール可能域のあるパネルでは、機能キーを PAGEUP および PAGEDOWN ダイアログ・コマンドに割り当てる必要があります。通常、これらは Page Up キーおよび Page Down キーですが、² 機能キーを使用する場合は、F7 および F8 が推奨されています。UIM では、異なる機能キーを PAGEUP および PAGEDOWN ダイアログ・コマンドに割り当てるのが可能ですが、ラベル付きページ・キーをその示された機能であるページアップおよびページダウン以外の機能に割り当てることはできません。

スクロール状況情報も UIM によって管理されて、各スクロール可能域の下部に表示されます。

メッセージ行は UIM の制御下にあります。メッセージ行にある正符号 (+) は、Page Down キーを押すことにより、さらに多くのメッセージが表示可能であることを示します。メッセージ領域はスクロール可能なので、スクロールのための機能キーが常に定義されている必要があります。

機能キー・スクロールの定義

UIM は、UIM 定義のパネルに対してすべてのスクロール・キーおよびページ・キーの機能を処理します。これらの機能は、アプリケーションによってオーバーライドされません。ただし、他の機能キーをスクロール・キーと同様に機能するように定義することは可能です。

PAGEUP または PAGEDOWN ダイアログ・コマンドに割り当てられた機能キーが押されると、UIM はパネル・タイプに適切な量をスクロールします。これは常に 1「ページ」となりますが、UIM はデータ項目グループ、メニュー項目グループ、および選択フィールドがページ境界を越えて分割しないように試行します。

2. 5250 キーボードでは、これらはそれぞれ Rolldown キーおよび Rollup キーになります。

カーソルの位置により、スクロールする領域が決まります。カーソルがスクロール可能域にない場合 (プルダウン・メニューの場合を除く)、スクロール要求は画面の一番上にあるスクロール可能域に適用されます。カーソルがスクロール可能域にある場合、スクロール要求はその領域に適用されます。スクロール可能域の境界は、メニュー域 (MENU)、データ表示域 (DATA)、リスト域 (LIST)、および情報域 (INFO) 言語タグで定義されたその領域の最上部および最下部の行です。これらの言語タグについては、511 ページの『付録 A. UIM パネル・グループ定義言語』を参照してください。表題および指示行は領域の一部ですが、領域のスクロール可能部分の一部ではありません。

コマンド行はスクロール可能域の一部としては扱われませんが、それを画面上の特定の領域と関連付けることはできます。たとえば、スクロール可能なメニュー域のあるパネルにコマンド行がある場合、それをメニューに関連付けることができます。ただし、スクロールの目的に関しては、そのコマンド行は領域の一部ではありません。

スクロールおよびエラー条件

メッセージ行は常に存在し、常に画面上の最後のスクロール可能域なので、スクロールの機能キーは常にアクティブであることが必要です。

メッセージ行をスクロールするとき、UIM はダイアログ変数の妥当性検査または更新を行いません。その他の領域をスクロールするとき、ダイアログ変数の妥当性検査または更新が画面上のすべての入力フィールドに対して行われます。エラーが検出されると、スクロール機能は実行されません。スクロール操作が進行するためには、画面上のすべての入力および出力フィールドが正しくなければなりません。

ページ・キーを押しても操作が実行できない場合、メッセージが表示されます。たとえば、Page Up キーを押したときに画面がすでに最上部にある場合、メッセージが表示されます。メッセージ行をその最上部または最下部を越えてスクロールしようとする、アラームが鳴ります。

リスト域のスクロール

リスト処理では、不完全リストが表示されることがあります。Page Down キーを押しても不完全リストの残り部分がリスト域に表示されない場合、UIM は不完全リストを扱う指定されたアプリケーション・プログラムを呼び出します。このプログラムは、リストに追加の項目を追加するか、またはリストの最下部に完全であることを示すマークを付けます。その後、プログラムは UIM に戻されます。UIM は、Set List Attributes (QUISETLA) API にあるリストの内容属性の設定値から、不完全リストであることを知ります。リストが完全になった後、UIM はアプリケーション・プログラムの介入なしにスクロールを処理します。

リストは最上部または最下部のどちらかで、不完全となることがあります。リストの現行位置よりも上で情報が欠落している場合、Page Up キーを押すと同じタイプの処理が行われます。

メニュー域のスクロール

UIM はメニュー項目グループを一緒に保持しようと試行しますが、グループが定義済み領域に収まらない場合はそれを分割します。メニュー・グループの個別のメニュー項目またはテキストは、使用可能なスクロール可能スペース内に収まる必要があります。それが収まらない場合には、パネル・グループが正常に作成されません。

情報域のスクロール

先頭に移動 (MOVETOP) ダイアログ・コマンドを使用すると、カーソルで選択された行がスクロール可能な情報域の上部に移動します。これにより、ユーザーは情報を最も読みやすい方法で配置できます。MOVETOP ダイアログ・コマンドについては、713 ページの『MOVETOP (先頭へ移動)』を参照してください。

データ項目グループのスクロール

UIM は、可能であればデータ項目グループを一緒に保持します。グループは、それがデータ域の列に収まらない場合にのみ分割されます。あるデータ項目グループが別のデータ項目グループ内にネストされていて、外側のグループを分割する必要がある場合、UIM は内側のグループを一緒に保持しようと試行しません。選択フィールドがデータ域の列に収まらない場合、パネル・グループは作成されません。

データ表示域を複数の列に表示することもでき、それらの列は左から右に使用されます。一般には使いやすさが低くなるので、スクロール可能データの表示域を 2 列のレイアウトで定義しないでください。

テキスト域のスクロール

テキスト域のスクロールは、テキスト域の出口プログラムによって処理されます。UIM は、テキスト域のスクロールを処理しません。スクロールのダイアログ・コマンドに割り当てられたキーが押されて、テキスト域をスクロールする必要が生じると、UIM はテキスト域の出口プログラムを呼び出します。制御が戻されると、UIM はパネルを再表示します。テキスト域の出口プログラムは、テキスト域をスクロールするためにテキスト域のダイアログ変数の値を変更する必要があります。

ユーザーが過剰にスクロールした場合、汎用パネル出口プログラムを使用して診断する必要があります。ユーザーが過剰にスクロールした場合、汎用パネル出口は適切なメッセージと、それに続く決定済みの操作を取り消す特殊なメッセージを送信する必要があります。決定済みの操作を取り消すメッセージについては、『API』のトピックで汎用パネル出口プログラムについてのセクションを参照してください。

文脈ヘルプの定義

文脈ヘルプは、UIM とシステム・ヘルプ機能が結合して提供されます。タグ言語を使用して、画面上のさまざまな領域にヘルプ・モジュールを関連付けるようにパネルが定義されます。ヘルプ・キーが押されると、UIM はどのヘルプ・モジュールを表示するかを判別します。

パネルを定義するとき、ユーザーはヘルプ・テキストを以下の領域に関連付けることができます。

- パネル全体
- メニュー・バー
- メニュー・バーの選択項目
- プルダウン・フィールドの選択項目
- 特定のメニュー項目
- 特定のデータ項目
- 特定のデータ選択フィールド
- データ選択フィールドの選択項目
- 特定のリスト列
- リスト列グループ
- 特定の機能キー
- 処置リスト内の特定のオプション
- データ・グループ
- データ域

ヘルプ・キーが押されると、UIM は主にカーソル位置に基づいてヘルプ・テキストを表示します。これには以下のような規則があります。

カーソル位置	ヘルプの選択規則
メニュー項目	そのメニュー項目のヘルプが表示されます。
リスト列	その列のヘルプが表示されます。 列の左および右の境界は、フィールドまたは見出し（どちらか大きいほう）の左および右の端に、ブランク 1 つをどちらかの側に追加して定義されます。列の最上部および最下部の境界は、見出しまたは列グループ見出しの最初の行、およびリスト項目が表示されることのある最後の行によって定義されます。 ヘルプは個別の項目ではなくリスト列グループに関連付けられます。この場合、カーソルが置かれている特定の列には関係なく、カーソルがグループ内に置かれていればそのグループ全体に対するヘルプが表示されます。列グループの境界は、個別の列の境界およびグループの列間にある区切り記号の合計に基づいて定義されます。 ヘルプが表示されている列が処置リストのオプション列である場合、表示されるヘルプにはリスト列に指定されたヘルプ、および各アクティブ・リスト操作に指定されたヘルプが含まれます。
機能キー定義域にある任意の行	すべてのアクティブな機能キーのヘルプが表示されます。
メッセージの表示されたメッセージ行	そのメッセージのヘルプが表示されます。メッセージのヘルプは、「追加のメッセージ情報」画面に示されます。メッセージのヘルプは、パネル・グループ内のヘルプ・モジュールを使用しません。
メニュー・バー選択項目	そのメニュー・バー選択項目、およびそのメニュー・バー選択項目の現在アクティブなすべてのプルダウン選択に対するヘルプが表示されます。
プルダウン選択	そのプルダウン選択のヘルプが表示されます。
プルダウン入力フィールド	プルダウンに関連したメニュー・バー選択項目（上記の『メニュー・バー選択項目』を参照）のヘルプが表示されます。有効なプルダウン選択がフィールドに入力された場合、そのプルダウン選択のヘルプが表示されます。
コマンド行	表示されるヘルプは、コマンド行の内容によって異なります。 コマンド行がメニュー域に関連付けられていて、そのコマンド行に数値が含まれる場合、カーソルがそのメニュー項目に置かれていた場合と同様にそのメニュー項目に対するヘルプが表示されます。表示されているかどうかには関係なく、任意のアクティブなメニュー項目に対してヘルプが使用可能です。 コマンド行にコマンドが含まれると想定される場合、そのコマンドに対するヘルプが表示されます。 コマンド行がパラメーター行として扱われている場合、拡張パネル・ヘルプが表示されます。 コマンド行がブランクの場合、拡張パネル・ヘルプが表示されます。
データ項目	データ項目に対してヘルプが指定されている場合、その特定のヘルプが表示されます。 データ項目にヘルプが指定されていない場合で、そのデータ項目がグループの一部であり、ヘルプがグループ・レベルで指定されている場合には、データ・グループ・ヘルプが表示されます。 項目にヘルプが指定されていない場合で、そのデータ項目がグループの一部ではないかまたはグループ・ヘルプが指定されていない場合には、その領域にヘルプが指定されていれば、領域レベルのヘルプが表示されます。

カーソル位置	ヘルプの選択規則
データ・グループ見出し	カーソルがデータ項目に置かれているときにも、同じ規則が適用されます。ヘルプは最も低いレベル (グループ、領域、または拡張パネル) に対して提示されません。
データ域	領域でヘルプが使用可能な場合は、そのレベルのヘルプが表示されます。そうでない場合は、拡張パネル・ヘルプが表示されます。
その他の位置	上記の条件のいずれにも適用しない場合には、拡張パネル・ヘルプが表示されます。これは、カーソルが表題行、指示行、およびほとんどのブランク・スペース上にある場合に、常に適用されます。表示する特定のヘルプが存在しない場合には常に、全般ヘルプが表示されます。

リスト・オプションまたは機能キー上の特定の操作に対する **HELP** は表示されることがありませんが、それらに対するヘルプ・テキストが個別に定義されています。これにより、アクティブ・オプションおよび機能キーを条件付けすることができます。表示されるヘルプは、アクティブな機能キーまたはオプションについてのみ説明します。

コマンド行の制約事項

UIM は、コマンド行に以下の 2 つのサイズを許可します。

- 1 行の、短いコマンド行
- 2 行の、長いコマンド行

コマンド行が長いかに短いかに応じて、UIM はコマンド行が機能キー域の前の 1 行または 2 行を占めるように、フォーマット設定して表示します。この位置は、スクロール、項目の条件付け、またはリスト項目の数に関係なく不変です。

コマンド行は、コマンド行 (CMDLINE) 言語タグを使用して定義されます。CMDLINE タグについては詳しくは、539 ページの『CMDLINE (コマンド行)』を参照してください。UIM では、このタグを任意のパネルに定義できますが、そのようにする必要はありません。

コマンド行をパネルに定義するとき、F4 キーを PROMPT ダイアログ・コマンドに割り当て、F9 キーを RETRIEVE ダイアログ・コマンドに割り当てることが推奨されています。これにより、ユーザーはコマンド行に入力したコマンドに関するプロンプトを出すこと、および以前に入力したコマンドを取得することができます。ダイアログ・コマンドについては詳しくは、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。また、機能キーの定義方法については詳しくは、596 ページの『KEYL (キー・リスト)』を参照してください。

UIM は、コマンド行をパネル上の領域に関連付けることもあります。メニュー域が存在する場合、コマンド行はメニュー域に関連付けられます。処置リストが存在する場合、コマンド行は処置リストに関連付けられます。どちらも存在しない場合、コマンド行はパネル上の領域に関連付けられません。UIM は、単一のパネル上に 2 つのメニュー域、2 つの処置リスト、またはこれらの組み合わせが存在することを許可しません。さらに、コマンド行とメニュー・オプション行は同時に使用できません。オプション行 (OPTLINE) タグについては詳しくは、647 ページの『OPTLINE (オプション行)』を参照してください。

コマンド行の解釈

コマンド行は、メニュー・オプションの選択、i5/OS または System/36 環境のコマンドの実行、および処置リスト上のオプションに関連したパラメーターの指定に使用されます。UIM は、コマンド行に入力された文字ストリングを選択的に解釈します。

多くのダイアログ・コマンドは、コマンド行への入力を必要としません。以下の状態は、UIM がコマンド行の内容を知る必要のある操作を想定しています。これらの操作は、ENTER、HELP、および PROMPT ダイアログ・コマンドに対するものです。Enter、Help、および Prompt 機能については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

- メニューに関連付けられたコマンド行では、UIM は最初の非ブランクの文字のセットを調べます。それが 1 つの桁 (16 進の F0 から F9 まで) だけで構成されている場合、コマンド行でメニュー項目が選択されたらと想定されます。そうでない場合、コマンド行はコマンドを含むものとして処理されます。
- 処置リストに関連付けられたコマンド行では、いずれかの処置リスト項目でいずれかのオプションが選択された場合、コマンド行にパラメーターが含まれると想定されます。そうでない場合、コマンド行はコマンドを含むものとして処理されます。
- メニューも処置リストも含まないパネルに関連付けられたコマンド行では、コマンド行にコマンドが含まれると常に想定されます。

正確にどの機能が次に実行されるかは、どの機能キーが押されるかに依存します。詳しくは、ENTER、HELP、および PROMPT ダイアログ・コマンドについての説明を参照してください。

長すぎるコマンドの入力

コマンド行に対して長すぎるコマンドも、そのプロンプトを出すことによってサブミットすることが可能です。コマンドが例外を出す場合、現行のコマンド内容を含むコマンド行が表示されます。完全なコマンド・ストリングがコマンド行に対して長すぎる場合、画面には表示可能な分が示されて、表示されたコマンド行の最後の 3 文字が継続を示す省略符号 (...) に置換されます。UIM は依然として、完全なコマンド・ストリングを内部的に維持します。

コマンド行が変更されると、UIM は内部バージョンを廃棄して、変更済みのコマンド行を新規要求として処理します。コマンドが変更されずにプロンプトまたはサブミットされると、内部バージョンはリセットされずに新規コマンド・ストリングとしてサブミットされます。変更は、コマンド行の内容の文字比較に基づいて行われます。

機能キーの定義

UIM は、機能キーを自動的に使用可能にしません。機能キーの正しいセットを使用可能にして、各キーに正しい機能が割り当てられていることを確認するのは、アプリケーション開発者の責務です。機能キーは、キー・リスト (KEYL) およびキー・リスト項目 (KEYI) 言語タグによって定義されます。これらの言語タグについては詳しくは、511 ページの『付録 A. UIM パネル・グループ定義言語』を参照してください。

UIM では、ラベル付きキーに対応するダイアログ・コマンドを割り当てる必要があります。

Help、Enter、Page Up、Page Down、Print、および Home キーには、それぞれのダイアログ・コマンドを割り当てる必要があります。PRINT ダイアログ・コマンド以外のダイアログ・コマンドは、他のキーにも割り当てることが可能です。たとえば、HELP ダイアログ・コマンドを F1 キーおよび Help キーに割り当てることが推奨されています。PRINT ダイアログ・コマンドは Print キーだけに割り当てることができ、条件付けすることはできません。

F1 から F24 までのキーには、PRINT および MOREKEYS を除く任意のダイアログ・コマンドを割り当てることができます。MOREKEYS ダイアログ・コマンドは、F24 にだけ割り当てることができます。

機能キーの書式設定

パネルの作成時に、UIM は機能キーおよびその説明に関するワーストケースの表示を判別して、それに応じて 1 行または 2 行を割り振ります。機能キーを表示するために使用可能な空間の最大量は、2 行です。

その後、この空間は固定されます。たとえば、特定のセットの状態のために 2 行が必要であると UIM が判断した場合、一部の状態で生じる機能キーの説明が 1 行だけであるとしても、常に 2 行が使用されます。

機能キーおよび VARUPD 値の処理

機能キーに割り当てられるダイアログ・コマンドは、特定のキーの機能を決定します。機能キーの扱いについて詳しくは、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

条件付けのために非アクティブになっている機能キーは、定義されていないかのように処理されます。ユーザーが、定義されていない機能キーを押すと、そのキーが許可されていないことを示すメッセージが表示されます。機能キー域に対してヘルプが表示される時、非アクティブのキーに対するヘルプ情報は表示されません。

機能キーを定義するとき、キー・リスト項目 (KEYI) またはプルダウン・フィールド選択項目 (PDFLDC) 言語タグの VARUPD 属性によって、機能キーが押されたときまたはプルダウン選択が選択されたときにダイアログ変数およびリスト項目値をユーザーが入力した値によって更新するかどうか定義されます。パネル上のいずれかのフィールドが妥当性検査に失敗した場合、機能キーまたはプルダウン選択に関連付けられた操作は実行されません。

ほとんどのダイアログ・コマンドには、VARUPD のために事前定義された不変の値がありますが、いくつかのダイアログ・コマンドにはそれがありません。機能キーがこれらのダイアログ・コマンドの 1 つに割り当てられていると、その要求の処理は VARUPD の値に依存します。

VARUPD=YES が指定されているとき、ユーザーがキー入力したすべての値は妥当性検査を通過する必要があります。妥当性検査に不合格となる値が存在する場合、以下が生じます。

- 妥当性検査を通過しないダイアログ変数は、更新されません。
- 指定された機能は実行されません。
- パネルの汎用出口プログラムが指定されている場合、それは呼び出されません。
- UIM は適切なエラー・メッセージと共に、同じパネルを再表示します。

VARUPD=NO が指定されているとき、以下が生じます。

- 妥当性検査は実行されず、ダイアログ変数は更新されません。
- 汎用出口が指定されている場合、それが呼び出されます。これは実行するダイアログ・コマンドが、通常の場合に出口プログラムを呼び出すためのものであると想定します。
- 指定された機能は、UIM がその機能を実行すべきでないことを汎用出口プログラムが示している場合を除いて、実行されます。出口プログラムについて詳しくは、『API』トピックを参照してください。

UIM が制御をアプリケーション・プログラムに戻すようにする VARUPD=NO がダイアログ・コマンドに指定されている場合、アプリケーション・プログラムはユーザーが入力する変数値を使用できません。その値はパネル用に保存されて、Display Panel (QUIDSP) API の再表示パラメーターを使用して再表示できます。これについて詳しくは、『API』トピックを参照してください。VARUPD=NO が CMD および CALL ダイアログ・コマンドに指定されている場合、UIM はダイアログ変数の画面コピーを保存し、指定された操作の完了後にパネルが再表示される時にそれを使用します。

VARUPD=NO が CMD および CALL ダイアログ・コマンドに指定されている場合、ダイアログ変数またはリスト項目が変更されると保存されたバージョンが失われることに注意してください。任意のフィールド

の保存されたバージョンを、そのフィールドの保存以後に基礎となるダイアログ変数が変更されていない場合に使用できます。リストでは、以下の 1 つが生じなければ、リスト項目内の変数の保存されたバージョンが使用されます。

- 基礎となるリスト項目が変更された
- リスト項目が画面上の位置を変えた
- リスト・ビューが変更された

変更が行われた場合、保存されたバージョンは失われて、新規に表示される値がダイアログ変数またはリスト項目から派生します。

たとえば、ダイアログ変数 VARX がパネル PANELX に存在していて、ユーザーが VARUPD=NO の指定された CALL ダイアログ・コマンドを使用する機能キーを押す直前に、それが変更されたと想定します。CALL ダイアログ・コマンドのターゲットが VARX を変更した場合、保存されたバージョンは失われます。呼び出しが戻された後に UIM がパネルを再表示するとき、表示されるバージョンの VARX はその変更に基づくものとなります。これにより、現行パネルに関連していない操作を実行するために、パネルの内容がすべての妥当性検査を通過することが必要ではなくなります。意図された操作が同じパネルまたはそれに関連したいずれかのダイアログ変数、リスト、または条件付けダイアログ変数を使用する場合、VARUPD=NO を CMD または CALL ダイアログ・コマンドと共に使用することはできません。

パネルの書式設定の概念

UIM パネルを書式設定する際は、以下の事柄に注意してください。

- アプリケーション開発者は、行や列を指定してパネルの書式設定を行うものではありません。その代わりに、UIM がタグ言語の記述を解釈して、各エレメントをパネル内のどこに配置するかを決めます。たとえば、表示パネル (PANEL) タグにはパネル・タイトルの記述が含まれますが、そのタイトルをどこに表示するかについての明示的な定義はありません。そのタイトルをパネルの幅の中で中央寄せすること、強調表示すること、および大/小文字混合として扱うことを、UIM が決定します。パネルの他のすべての領域にも同じ概念が適用されます。ただし、他の例のほとんどはより複雑です。
- UIM は、パネル・グループの作成時に可能な限り多くの書式設定を行います。多くのパネル・エレメントの実際の配置は実行時に決定されます。これにより、UIM は画面の縦の長さ、および他の実行時の事情を考慮に含めることができます。たとえば、UIM はメニュー項目が条件付きで表示されるようにすることができます。これらの条件はパネルが表示されるときに評価されて、メニュー域のフォーマットはその結果に基づくものとなります。

パネルの書式設定をいつ行うか

パネルの書式設定とパネル内容とは異なります。パネルの書式設定とは、パネルのさまざまな領域の編成のことです。パネル内容とは、ダイアログ変数の内容によって示されるフィールドの値のことです。

パネルの書式設定は、パネル・グループのコンパイル時に行われ、実行時に再度行われます。

ほとんどの決定はパネル・グループのコンパイル時に行われ、その情報はパネル・グループ・オブジェクト内に保持されます。パネル・グループのコンパイル時に行うことのできない書式設定の決定は、Display Panel (QUIDSPP) API 処理の際に行われます。

この区別の例として、条件付きのデータ項目があります。プロンプト・テキスト・レイアウトのすべての詳細、リーダー・ドットの数、フィールド位置、および可能な選択項目のフォーマット設定は、パネルの作成時に行われます。

実行時には、QUIDSPP API 処理が条件を評価して項目を表示するかどうかを決定し、内部フォーマットを作成します。Open Display Application (QUIOPNDA) および Close Application (QUICLOA) API で定義されるアプリケーション内では、どの時点でも各パネルに 1 つだけの内部フォーマットがあります。内部フォーマットは、パネルが最初に表示されるときに作成されて、アプリケーションが閉じられるまでそのフォーマットが後続の使用のために保持されます。

これは、内部フォーマットがアプリケーション内で変化することはないという意味ではありません。ほとんどの内部フォーマットは変化しませんが、メニュー項目、リスト・アクション、機能キーなどの、いくつかのパネル・エレメントはアプリケーションによって条件付けられています。これらの領域に対する変更の影響はパネルが表示されるたびに評価されて、フォーマットが更新されます。

UIM はコマンドのサブミットやプログラムの呼び出しなど多くの機能を実行するので、QUIDSPP API を呼び出したアプリケーションに制御を戻す前に、特定のパネルがユーザーに複数回表示されることがあります。パネルが表示されるたびに、UIM はパネルの書式設定を再評価します。たとえば、機能キーを押すとアプリケーション・プログラムが呼び出されます。アプリケーション・プログラムが制御を UIM に戻すと、UIM はパネルを再表示しますが、それは内部フォーマットに影響を与える状態がアプリケーション・プログラムによって変更されたかどうかを調べてからになります。たとえば、QUIDSPP API が呼び出されてメニュー域のあるパネルを表示する場合、いくつかのメニュー項目の条件付けに影響を与えるダイアログ変数がアプリケーション・プログラムによって変更されたときは、UIM がメニュー項目を再表示するときに、それはこれらの変更を反映するものとなります。

パネルの書式設定のアプリケーション制御

言語タグを使用すると、アプリケーション開発者は以下の方法でパネルの書式設定を制御できます。

- 列の幅、データ域の垂直対水平の表示、領域の縦の長さなど、書式設定の一部を指定することにより。これらの指定はパネルのコンパイル時に固定されて、実行時にアプリケーションが変更することはできません。これにより、代替の書式設定手法を選択することができ、実行時処理の軽減に役立つ UIM 情報が提供されます。
- 実行時の条件付けを指定することにより。ダイアログ変数を使用して、メニュー項目、メニュー項目グループ、処置リスト・オプション、機能キー、データ項目、プルダウン選択、データ選択フィールド、選択フィールドの選択項目、およびデータ項目グループを条件付けることができます。ほとんどの場合、これらの条件はパネルが表示されるたびに評価されて、表示に何を含めるかが判別されます。

条件付けは、以下のいずれかに基づくものとすることができます。

- アプリケーション定義のダイアログ変数。
- UIM 定義のダイアログ変数。
- システム上のオブジェクト、またはシステム上のオブジェクトに対するユーザー権限の存在。
- アプリケーション・ユーザーのユーザー・クラス
- 条件付けは、ユーザー・プロファイル、ユーザー・クラス、またはユーザー機能の制限 (ZLMTCPB) ダイアログ変数の設定値に基づくことがあります。ダイアログ変数について詳しくは、693 ページの『UIM によって定義されるダイアログ変数』を参照してください。

条件の定義方法について詳しくは、540 ページの『COND (条件定義)』を参照してください。

UIM は、条件付けに対して追加の制御を行うことを許可します。パネル定義では、一度だけ評価する条件、最初に必要とされるときに評価する条件、毎回評価する条件を指定できます。この目的は、主にパフォーマンスに関連しています。ほとんどの項目は、ユーザー・プロファイルや権限などパネルが最初に表示されるときに固定されるものに対して条件付けされます。これらの条件を一度だけ評価するように指定すると、不必要な条件評価を回避できます。

パネル・フォーマッターの制限

UIM には、ほとんどの通常のフォーマット設定要件を扱うために設計されたフォーマッターが備わっています。ただし、フォーマッターは手作業によるフォーマット設定と同じ結果を常に生じさせるとは限りません。フォーマッターは実行時環境によって制約されるため、また一般化した入力を処理する必要があるためです。この例として、機能キーの記述のフォーマット設定があります。手作業によるフォーマット設定では、位置合わせとバランスを最適にするために表現を変更することもできます。UIM には、テキストの長さを変更したり、バランスのとれた長さのセットを想定するオプションがありません。

一般に、パネル上に情報が密集しているほど、UIM フォーマッターの成果は手作業によるフォーマット設定の成果に及ばなくなります。

EXIT の要求時に複数のパネルをまとめて閉じる

アプリケーションが単一の終了要求の処理の一部として複数のパネルおよびプログラムをまとめて閉じる機能をサポートするために、オペレーティング・システムはジョブごとに終了が要求されたかどうかを示すフラグを保守します。この終了フラグは、終了要求が複数のパネルをまとめて閉じるように、Display Panel (QUIDSPP) API 上のユーザー・タスク・パラメーターと組み合わせて使用します。

アプリケーション・プログラムは以下の 2 つの方法で終了フラグをオンにすることができます。

1. EXIT ダイアログ・コマンドに SET パラメーターを指定します。UIM が EXIT ダイアログ・コマンドに SET パラメーターを指定して実行すると、ジョブの終了フラグがオンになります。EXIT ダイアログ・コマンドについて詳しくは、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。
2. Put Dialog Variable (QUIPUTV) API を使用して、UIM 定義の変数 ZEXIT を「1」に設定します。この Z 変数を設定すると、UIM はジョブの終了フラグをオンにします。UIM は操作の処理を開始するとき、終了フラグを自動的にオフにします。

UIM が、機能キーに割り当てられたコマンドのサブミットや、メニュー項目が選択された結果としてのプログラムの呼び出しなど、パネルに定義された操作を実行した後で制御を再取得すると、ジョブの終了フラグにチェックが付きます。フラグがオンであり、パネルが古いユーザー・タスク (QUIDSPP API のユーザー・タスク・パラメーター) として表示される場合、UIM はその時点で呼び出し側アプリケーションに制御を戻します。さらに、機能を要求した QUIDSPP API のパラメーターには、EXIT ダイアログ・コマンドの結果としてパネル表示が終了したことを示す標識が含まれます。

ジョブの終了フラグがオフであるか、またはパネルが新規のユーザー・タスクとして表示される場合、特別なことは何も生じません。UIM は終了フラグをオフにして、パネルを再表示します。

コマンドがコマンド行に入力されたときなど、実行される操作がパネルに定義されたものでない場合、終了フラグは常にオフになっていて、パネルが新規ユーザー・タスクまたは古いユーザー・タスクのどちらとして表示されるかは処理に影響しません。

以下は、終了フラグの機能を示す例です。388 ページの図 113 は、この例を図示しています。プログラム PGMX が、CALL ダイアログ・コマンドに割り当てられた機能キーのある PANELX を表示すると想定します。さらに、PGMY と呼ばれるプログラムが PANELY と呼ばれる別のプログラムを表示するとします。ここで、PANELX が古いユーザー・タスクとして表示され、機能キーが押され、PANELY が表示されたらと想定します。PANELY から、ユーザーは SET パラメーターを使用して EXIT ダイアログ・コマンドに割り当てられた機能キーを押します。その後、制御は PGMY に戻ります。EXIT ダイアログ・コマンドが処理されるとは、UIM は終了フラグをオンにします。ここで PGMY が制御を UIM に戻すと、UIM

は EXIT ダイアログ・コマンドが要求されたことを示す標識と共に制御を PGMX に戻します。結果として、単一の終了要求によって 2 つのパネルが迂回されます。

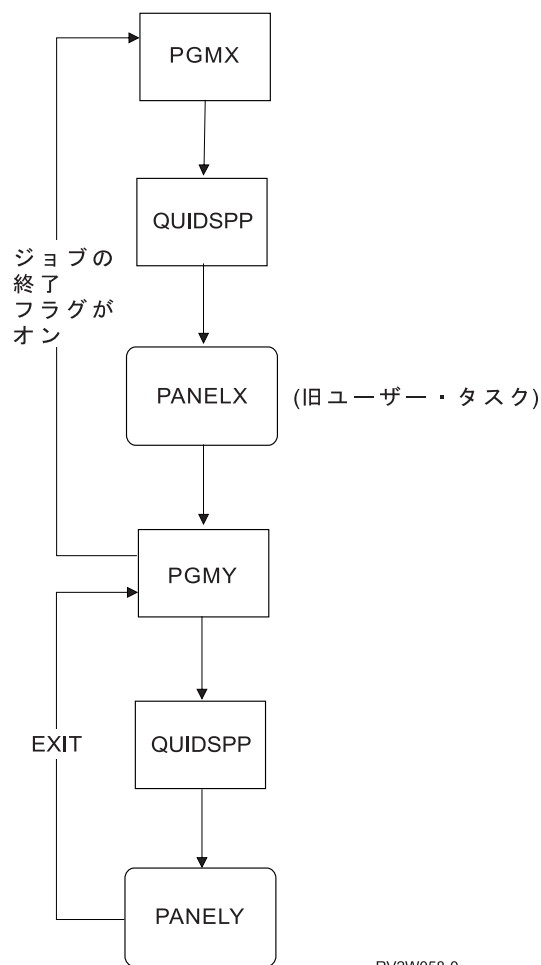


図 113. ジョブ終了フラグの例

EXIT ダイアログ・コマンドに割り当てられた PANELY 上の機能キーが SET パラメーターを使用しない場合、ジョブの終了フラグがオンになっていないので、UIM は PANELX を再表示します。

API について詳しくは、iSeries Information Center の『API』のトピックを参照してください。

リスト・パネルをまとめて閉じる

処置リスト・パネルでリスト・オプションを処理した結果として表示されるパネルでは、CANCEL、EXIT、および ENTER ダイアログ・コマンドは以下のように機能することになります。

オペレーティング・システムはジョブごとに取り消しが要求されたかどうかを示すフラグを保守します。このフラグは、終了処理に関して説明されたフラグと似ています。387 ページの『EXIT の要求時に複数のパネルをまとめて閉じる』を参照してください。ただし、取り消しフラグが複数のパネルを迂回するために使用されることはありません。また、Display Panel (QUIDSPP) API 上のユーザー・タスク・パラメーターに関連して使用されることもありません。

アプリケーション・プログラムは以下の 2 つの方法で取り消しフラグをオンにすることができます。

1. CANCEL ダイアログ・コマンドに SET パラメーターを指定します。 UIM が CANCEL ダイアログ・コマンドに SET パラメーターを指定して実行すると、ジョブの取り消しフラグがオンになります。(SET パラメーターを指定した CANCEL ダイアログ・コマンドについて詳しくは、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。)
2. QUIPUTV API を使用して、UIM 定義の変数 ZCANCEL を「1」に設定します。この Z 変数を設定すると、UIM はジョブの取り消しフラグをオンにします。

取り消しキーを押すとリスト操作の処理が停止して、処理が停止した時点での処置リストが再表示されるようにすることは、アプリケーション開発者の責務です。リスト・オプションの処理の結果として別のパネルを表示できる場合、CANCEL ダイアログ・コマンドに SET パラメーターを指定することにより、取り消しキーが押されたときにジョブの取り消しフラグをオンにする必要があります。UIM は、処置リスト処理を実行するとき、各リスト・アクションの完了後に取り消しフラグを調べます。UIM は、フラグがオンであることを検知すると、残りのリスト・オプションの処理を停止してリストを再表示します。直前に処理されたリスト項目のオプション・フィールドは消去されて、操作が試行されなかったリスト項目のオプションはリストに残ります。UIM は、リスト・アクションの処理の後に取り消しフラグを調べるだけです。

Exit キーを押すとリスト操作の処理が停止して、処理が停止した時点での処置リストが再表示されるか、または処置リスト・パネルを終了するようには、アプリケーション開発者の責務です。リスト・オプションの処理の結果として別のパネルを表示できる場合、Exit ダイアログ・コマンドに SET パラメーターを指定することにより、Exit キーが押されたときにジョブの終了フラグをオンにする必要があります。UIM は、処置リスト処理を実行するとき、各リスト・アクションの完了後に終了フラグを調べます。UIM がフラグがオンであることを検知すると、残りのリスト・オプションの処理は停止して、以下が生じます。

- 処置リスト・パネルが新規のユーザー・タスクとして表示される場合、パネルは再表示されます。
- 処置リスト・パネルが古いユーザー・タスクとして表示される場合、パネルは再表示されません。EXIT が使用されたことを示す戻り関数と共に処置リスト・パネルを表示したプログラムに制御が戻されます。

どちらの場合でも、直前に処理されたリスト項目のオプション・フィールドは消去されて、操作が試行されなかったリスト項目のオプションはリストに残ります。UIM は、メニュー・オプション、機能キー、およびプルダウン・メニュー選択項目の処理が終了して制御が戻ったときに、ジョブの終了フラグも調べます。詳しくは、387 ページの『EXIT の要求時に複数のパネルをまとめて閉じる』を参照してください。

Enter キーを押すとリスト操作の処理が継続するようには、アプリケーション開発者の責務です。取り消しフラグと終了フラグのどちらもオンでないときに、次のリスト・アクションが処理されます。

別のパネルへのポップアップ・ウィンドウの追加

以下のいずれかの方法で、別のパネル上にポップアップ・ウィンドウを追加できます。

- CALL ダイアログ・コマンドを使用して、UIM がアプリケーション出口プログラムを呼び出し、それがウィンドウを追加して表示できるようにします。これは、要求の終了 (ENDRQS) コマンドが一貫性のある結果を生じるようにするための、推奨される設計です。
- RETURN ダイアログ・コマンドを使用して、UIM が基礎となるパネルを表示したプログラムに戻り、ウィンドウを追加して表示できるようにします。

UIM は、ポップアップ・ウィンドウ内にメニューまたは処置リストのあるパネルを表示しているとき、そのパネルにコマンド行もある場合を除いて、要求処理プログラムにはなりません。これは、ウィンドウがコマンド行、処置リスト、またはメニュー域のある基本パネル上に表示されているときにも適用されます。

UIM がフルスクリーン・パネルを表示中に要求処理プログラムになった場合、アプリケーションが CALL ダイアログ・コマンドを使用してポップアップ・ウィンドウを表示するように設計されているときは、プログラム・スタックにフルスクリーン・パネル用の UIM 要求処理プログラムが含まれます。ただし、アプリケーションが RETURN ダイアログ・コマンドを使用してウィンドウを追加および表示するように設計されているときは、プログラム・スタックに UIM 要求処理プログラムが含まれません。そのため、ENDRQS コマンドはアプリケーションの内部設計に応じて異なる結果を生じます。

一般ガイドラインとして、プログラムが適切なきに要求処理プログラムとなるのでなければ、ポップアップ・ウィンドウからのリスト・オプションまたはメニュー・オプションの結果として長い時間実行する機能は設計しないでください。

RETURN ダイアログ・コマンドに依存する設計を使用することにより、プログラムが要求処理プログラムとなって 1 つ以上のポップアップ・ウィンドウを表示する場合、ウィンドウを表示した機能が ENDRQS コマンドで取り消されたときにウィンドウ・スタックがクリーンアップされるようにコーディングする必要があります。これが生じたとき、アプリケーション・ウィンドウ・スタックから削除するウィンドウごとに、Remove Pop-up Window (QUIRMVPW) API を 1 回ずつ呼び出す必要があります。

CALL ダイアログ・コマンドに依存する設計を使用することにより、プログラムが要求処理プログラムとなって 1 つ以上のポップアップ・ウィンドウを表示する場合、ウィンドウを表示した機能が ENDRQS コマンドで取り消されたときに UIM がウィンドウ・スタックを自動的にクリーンアップします。

要求処理プログラムについて詳しくは、iSeries Information Center の『CL』のトピックを参照してください。

メニュー・バーの使用方法

メニュー・バーは、パネルの最上部に配置されてパネル全体の幅まで拡張します。メニュー・バーは、24 x 80 バイトまたは 27 x 132 バイトのパネルでのみ許可され、ポップアップ・ウィンドウ内では許可されません。

メニュー・バーには、ユーザーが要求する可能性のある選択項目のリストが含まれています。これらの選択項目は 1 行から 3 行を使用して水平にリストされ、すべての選択項目の間には 3 つのブランクが置かれます。

ユーザーが Enter キーを押して選択項目を選択すると、プルダウン・メニューがメニュー・バーのすぐ下に表示されます。このメニューから、ユーザーは 1 つの選択項目を選択できます。

プルダウン・メニュー内の各選択項目は、以下の 1 つを行います。

- パネルまたはパネルの領域に対して全体的に操作を行います。この場合、UIM は選択項目がユーザーによって選択されるごとに 1 回ずつ操作を行います。
- パネル上の処置リストまたは選択リストで選択された項目ごとに、操作を行います。この場合、UIM は選択されたリスト項目ごとに 1 回ずつ操作を実行します。ユーザーはスラッシュ (/) または国別の選択文字を目的の項目の隣に入力して、選択リスト上の項目または操作を選択します。

メニュー・バーの定義方法について詳しくは、511 ページの『付録 A. UIM パネル・グループ定義言語』で以下の言語タグを参照してください。

- PANEL (表示パネル)
- MBAR (メニュー・バー)
- MBARC (メニュー・バー選択項目)

- PDFLD (プルダウン・フィールド)
- PDFLDC (プルダウン・フィールド選択項目)

プルダウン・メニューとポップアップ・ウィンドウとの違い

プルダウン・メニューは、視覚的にはポップアップ・ウィンドウと似ています。プルダウン・メニューおよびポップアップ・ウィンドウはどちらも基礎となるパネル・イメージをオーバーレイしますが、ポップアップ・ウィンドウは別個のアプリケーション・パネルであり、プルダウン・メニューはメニュー・バーを含むパネルの拡張機能です。

視覚的には、プルダウン・メニューは以下の点でポップアップ・ウィンドウと異なります。

- メニュー・バーの下に表示される区切り行は、常にプルダウン・メニューの上部の枠として使用されます。ポップアップ・ウィンドウの枠が基礎となるパネルの一部であることはありません。
- プルダウン・メニューには表題がありません。ポップアップ・ウィンドウには表題があります。
- プルダウン・メニューには機能キー記述がなく、プルダウン・メニューが基礎となるパネルの機能キー記述をオーバーレイすることはありません。ポップアップ・ウィンドウには通常、機能キー記述があり、ウィンドウが基礎となるパネルの機能キー記述をオーバーレイすることがあります。
- プルダウン・メニューにはメッセージ行がなく、ユーザーがプルダウン・メニューと対話した結果として表示されるメッセージは、基礎となるパネルのメッセージ行に表示されます。ポップアップ・ウィンドウには常にメッセージ行があり、ユーザーがポップアップ・ウィンドウと対話した結果として表示されるメッセージは、ポップアップ・ウィンドウのメッセージ行に表示されます。

プルダウン・メニューとポップアップ・ウィンドウとは、それぞれに対してユーザーが対話できる方法でも異なります。その違いは、以下のとおりです。

- プルダウン・メニューが表示される時、カーソル・タブ・キーを押すとカーソルがメニュー・バー内の最初の未選択項目に移動します。再びタブ・キーを押すとカーソルが次の未選択項目に移動して、カーソルがプルダウン・メニューに戻るまでこれが繰り返されます。ポップアップ・ウィンドウが表示される時、カーソル・タブ・キーを押すとカーソルがポップアップ・ウィンドウ内に定義された次の入力フィールドに移動します。タブ・キーによって、カーソルがポップアップ・ウィンドウの枠の外側に移動することはありません。
- プルダウン・メニューが表示される時、基礎となるパネルに定義されたすべての機能キーはアクティブです。以下の例外を除いて、各機能キーはプルダウン・メニューが表示されていない場合と同じ機能を実行します。
 - Enter キーまたは ENTER ダイアログ・コマンドに割り当てられた他のキーは、ユーザーが選択したプルダウン選択番号に割り当てられた操作を処理するために使用されます。
 - CANCEL ダイアログ・コマンドに割り当てられたキーは、プルダウン・メニューを除去して基礎となるパネルを再表示します。通常、F12 が CANCEL ダイアログ・コマンドに割り当てられるキーとなります。
 - Page Down および Page Up キー、または PAGEDOWN または PAGEUP ダイアログ・コマンドに割り当てられた他のキーは、カーソルがプルダウン・メニュー内にあるときには使用できません。
 - PROMPT ダイアログ・コマンドに割り当てられたキーは、カーソルがプルダウン・メニュー内にあるときには使用できません。通常、F4 が PROMPT ダイアログ・コマンドに割り当てられるキーとなります。

カーソル移動に影響される機能キーでは、ユーザーはカーソルをプルダウン・メニューの枠の外側に移動して、機能キーを押すことができます。この場合、ユーザーが機能キーを押したときにプルダウン・メニューが表示されていなかったかのように、UIM はプルダウン・メニューを除去して機能キーに割

り当てられた操作を実行します。カーソル移動に影響される機能キーは、以下の表にあるダイアログ・コマンドに割り当てられたいずれかのキーです。この表のダイアログ・コマンドは、通常は示されたキーに割り当てられます。

表 33. ダイアログ・コマンドに割り当てられるカーソル移動に影響される機能キー

カーソル移動に影響される機能キーを割り当てることができるダイアログ・コマンド

ダイアログ・コマンド	ダイアログ・コマンドが通常割り当てられるキー
HELP	Help
PAGEDOWN	Page Down
PAGEUP	Page Up
MOVETOP (先頭へ移動)	F10
CHGVIEW (ビュー変更)	F11
PROMPT	F4

例として、プルダウン・メニューが表示されていると想定します。ユーザーがプルダウン・メニューの枠の外側にある基礎となるパネル内のフィールドにカーソルを移動して、ヘルプ・キーを押した場合、UIM はプルダウン・メニューを除去してそのフィールドに対して定義された文脈ヘルプを表示します。

ポップアップ・ウィンドウが表示されるとき、基礎となるパネルに定義された機能キーでウィンドウの内側にあるものだけがアクティブになります。機能キーが押されるとき、カーソルがポップアップ・ウィンドウの枠の外側にある場合、ウィンドウが再表示されて、カーソルはウィンドウの外側では許可されないことを示すアラームが鳴ります。

プルダウン・メニューとポップアップ・ウィンドウとのその他の違いは、アプリケーション開発者がそれらを制御できる方法です。以下に違いを示します。

- プルダウン・メニューには、番号付き選択項目のリストだけを含めることができます。アプリケーション開発者はプルダウン・メニューに表示される各項目を定義して、各項目に対する操作も割り当てます。ユーザーが選択項目の 1 つを選択すると、UIM はその選択項目に定義された操作を実行します。ポップアップ・ウィンドウには、フルスクリーン・パネルに表示可能な任意のパネル・エレメントを含めることができますが、これには以下の例外があります。
 - ポップアップ・ウィンドウには画面の複数行に渡るフィールド (ダイアログ変数) を含めることができません。すべてのフィールドは、ウィンドウの幅に収まる必要があります。
 - ポップアップ・ウィンドウには、メニュー・バーを含めることができません。
- ユーザー対話に基づいて、UIM はプルダウン・メニューを表示するときと除去するときとを判別しません。アプリケーション・プログラムがプルダウン・メニューをいつ表示または除去するかを制御することはできません。また、アプリケーション・プログラムが表示されたときのプルダウン・メニューの位置またはサイズを制御することもできません。アプリケーション・プログラムは、ポップアップ・ウィンドウをいつどこに表示するかを制御します。ポップアップ・ウィンドウの表示について詳しくは、393 ページの『ウィンドウの追加および除去』を参照してください。ポップアップ・ウィンドウのサイズは、アプリケーション開発者がウィンドウに表示されるパネルの PANEL タグ上に WIDTH および DEPTH 属性を指定することによって、指定されます。

ポップアップ・ウィンドウの使用法

ポップアップ・ウィンドウは、画面の一部をオーバーレイする情報です。ユーザーはウィンドウの内側の情報、および画面内のウィンドウによってオーバーレイされていない部分を見ることができます。ただし、そのウィンドウだけがアクティブです。ユーザーは基礎となるパネルに対して作業することはできません。複数のウィンドウが表示される場合、一度に 1 つのウィンドウのみ活動状態になります。

ポップアップ・ウィンドウに関するいくつかの原則を以下に示します。

- ウィンドウにはメッセージ行が含まれていて、メッセージがウィンドウの境界を越える場合、メッセージは切り捨てられます。メッセージが切り捨てられたことを示すために、省略符号 (...) がメッセージの最後に追加されます。メッセージの残りの部分を表示するには、カーソルをウィンドウのメッセージ行に置いてヘルプ・キーを押します。
- ウィンドウが表示されると、そのサイズおよび位置を変更することはできません。
- カーソルはアクティブ・ウィンドウに配置されます。ユーザーがカーソルをアクティブ・ウィンドウの外側に置いて Print 以外のいずれかの機能キーを押した場合、表示アラームが鳴ってカーソルは直前のウィンドウ位置に戻ります。
- ウィンドウを閉じると、基礎となる画面内のウィンドウを要求する前の位置に戻ります。
- Print キーは現行ウィンドウだけではなく、画面全体に適用されます。

アプリケーション・ウィンドウの定義

アプリケーション定義のウィンドウは、UIM パネルおよび DDS パネル上に表示できます。

アプリケーション定義とは、アプリケーション開発者がウィンドウの正確なサイズをパネル・グループ・ソースの表示パネル (PANEL) タグに定義することです。ウィンドウの位置は、アプリケーション・プログラムが Add Pop-Up Window (QUIADDPW) API を呼び出すときに指定されます。

ウィンドウの追加および除去

UIM は、ウィンドウを追跡するためのスタックを保守します。ウィンドウは、以下の方法の 1 つでスタックに追加されます。

- UIM は、ヘルプ情報およびコマンド行ウィンドウを表示するためにウィンドウをスタックに追加します。
- アプリケーション開発者は、ウィンドウを表示するためにウィンドウをスタックに追加するように要求できます。

UIM またはアプリケーション開発者は、ウィンドウを内部的にスタックから除去できます。

ウィンドウの追加および除去は、アプリケーション・プログラムによって制御されますが、UIM にはこれらのウィンドウを操作するための API のセットが備わっています。

ウィンドウを表示するには、アプリケーション・プログラムは最初に Add Pop-Up Window (QUIADDPW) API を使用して、UIM に次の表示はウィンドウのためのものであることを知らせる必要があります。選択リストを表示しているウィンドウはフィールド隣接配置を使用する必要があり、リストを位置決めするためのプロンプトを表示しているウィンドウはオフセット配置を使用する必要があります。

QUIADDPW API 呼び出しは、ウィンドウに表示されるパネルを指定しません。これはウィンドウの配置に必要なロケーション情報だけを指定します。

ウィンドウを画面に追加できるのは、フルスクリーン・パネルが決定された後だけです。フルスクリーン・パネルは、Display Panel (QUIDSPP) または Set Screen Image (QUISETSC) API を呼び出すことによって決定されます。フルスクリーン・パネルがアプリケーション内で表示された後は、ウィンドウをアプリケーションに追加できます。

フィールド隣接配置を使用して選択リスト・ウィンドウを配置するために、プログラマーは選択ウィンドウが表示されているフィールドの名前を指定する必要があります。パネルがウィンドウ内に表示される時、UIM は汎用のウィンドウ配置規則を使用してウィンドウを配置することにより、可能であればウィンドウが指定のフィールドをオーバーレイしないようにします。

リストを配置するためのプロンプトを表示しているウィンドウでは、Add Pop-Up Window (QUIADDPW) API 呼び出しにオフセット配置を指定する必要があります。ウィンドウの左上隅は基礎となるパネルの左上隅に対して下方に配置され、右にインデントされます。

ウィンドウが画面に追加された後に、Display Panel (QUIDSPP) API を使用してウィンドウ内にパネルを表示します。QUIDSPP API が呼び出されると、UIM は以下の操作を実行します。

1. ウィンドウの実際の位置を決定します。この位置は、QUIADDPW API への呼び出しによって提供されたロケーション情報に基づいて決定されます。
2. パネルをフォーマット設定します。
3. 新規にフォーマット設定されたウィンドウ枠を基礎となるパネルのイメージにマージします。
4. マージされたパネル出力を表示します。

エンド・ユーザーは、他の基礎となるパネルまたはウィンドウと対話する前に、表示された最新のウィンドウと対話する必要があります。

Remove Pop-Up Window (QUIRMVPW) API を使用して、ウィンドウをディスプレイから除去します。UIM が現在操作を処理している場合、またはウィンドウに表示されたパネルの出口プログラムを実行している場合には、ウィンドウを除去できません。複数のウィンドウを同時に除去できます。

QUIRMVPW API によって UIM が基礎となるパネルを自動的に再表示することはありません。パネルがウィンドウなしで再表示されるのは、次に QUIDSPP API を呼び出すとき、または UIM が機能を処理した後か出口プログラムを実行した後にパネルを自動的に再表示するときです。

ウィンドウ内でのコマンド行の使用法

ウィンドウ内のコマンド行によって、永続的なコマンド行のためのパネル上の予約済みスペースを必要とすることなく、ユーザーはコマンドを入力できます。UIM はウィンドウ内にコマンド行を表示するサポートを提供します。UIM アプリケーション・パネルでは、これは機能キーを CMDLINE ダイアログ・コマンドに割り当てることによって行います。F9 キーが推奨されています。CMDLINE ダイアログ・コマンドも、プルダウン・メニュー内の選択項目に対する操作として許可されます。機能キーまたはプルダウン・フィールド選択項目が CMDLINE ダイアログ・コマンドに割り当てられると、そのキーまたは選択項目を ZLMTCPB ダイアログ変数に基づいて条件付けすることができます。これを行うと、ユーザーに制限された権限がない場合に、そのユーザーに対してコマンド行ウィンドウの使用だけを許可することができます。

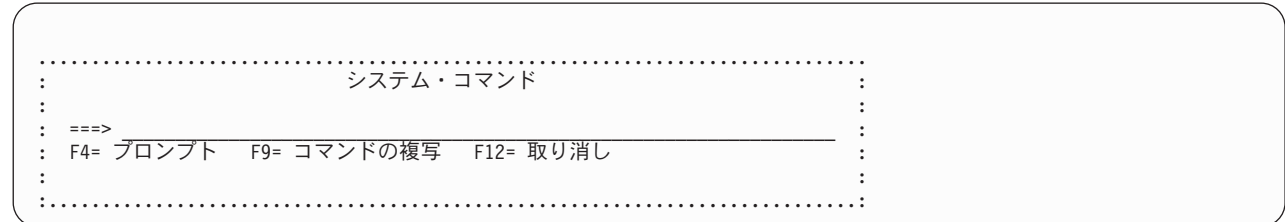
DDS パネルでは、コマンド行は QUSCMDLN API を呼び出すことで提供されます。

ウィンドウ内のコマンド行を使用するときには、以下の事柄に注意してください。

- ユーザーがコマンド行 (CMDLINE) ダイアログ・コマンドの処理を生じさせる機能キーを押したときに、ポップアップ・ウィンドウが表示されます。カーソルはコマンド行の先頭位置に置かれ、ユーザーはコマンド行を使用して任意のシステム・コマンドを入力できます。

- UIM は、ウィンドウ用のパネル定義を提供して、それを画面の最下部に配置します。
- コマンド行のウィンドウは、そのサイズも位置も変更できません。ユーザーは Enter キーまたは F12 (取り消し) 機能キーを押して、ウィンドウを除去する必要があります。コマンドを実行しても、ウィンドウは除去されません。

以下の例は、画面上で使用されるコマンド行ポップアップ・ウィンドウを示しています。



パネルを表示する際の要求処理プログラムとしての UIM

パネルを表示する際に UIM が要求処理プログラムになることがあります。UIM が要求処理プログラムになると、アプリケーションはコマンド行に入力されたコマンドから分離されます。たとえば、要求の終了 (ENDRQS) コマンドを使用してコマンド行に入力されたコマンドを取り消す場合、UIM が終了要求を処理するので、アプリケーション・プログラムは終了しません。以下のタイプのパネルでは、UIM は自動的に要求処理プログラムになります。

- コマンド行のあるパネル。コマンド行からコマンドをサブミットする際に、UIM が要求処理プログラムとなる必要があります。パネルにコマンド行があるとき、UIM は要求処理プログラムになるので、パネルに対して実行されるすべての機能は要求の終了 (ENDRQS) コマンドに関して一貫して機能するようになります。
- 処置リストのあるフルスクリーン・パネル。ほとんどの処置リスト・パネルには、リスト・オプションにパラメーターを指定するためのコマンド行があります。それで整合性のために、UIM は処置リストのあるフルスクリーン・パネルの要求処理プログラムとなります。これにより、リスト・オプションの処理中に ENDRQS コマンドが要求されたとき、コマンド行のない ACTOR=UIM 処置リストはコマンド行のあるものと同様に機能します。これにより、ユーザーは ENDRQS コマンドを実行して、処置リスト・パネルとそれを表示したプログラムまでも取り消すことなく、長時間実行しているオプションを取り消すことができます。ENDRQS コマンドを使用して UIM 処理のリスト・オプションを取り消すとき、未処理のリスト・オプションがあれば、処置リスト内に残されて保留にされます。

処置リストに対して ACTOR=CALLER をコーディングする場合、UIM との整合性のためにコードを要求処理プログラムとすることを検討してください。

- メニュー域のあるフルスクリーン・パネル。ほとんどのメニュー・パネルには、コマンド行がありません。整合性のために、UIM はメニュー域のあるすべてのパネルの要求処理プログラムとなります。これにより、メニュー・オプションの処理中に ENDRQS コマンドが要求されたとき、コマンド行のないメニュー・パネルはコマンド行のあるものと同様に機能します。これにより、ユーザーは ENDRQS コマンドを実行して、メニュー・パネルとそれを表示したプログラムまでも取り消すことなく、長時間実行しているオプションを取り消すことができます。

メニュー項目に対して RETURN ダイアログ・コマンドをコーディングする場合、UIM との整合性のためにコードを要求処理プログラムとすることを検討してください。

要求処理プログラムについて詳しくは、iSeries Information Center の『CL』のトピックを参照してください。

印刷の概念

UIM により、アプリケーションは CL コマンドの OUTPUT(*PRINT) パラメーターで必要とされるレベルの印刷出力を、定義して生成することができます。このサポートは、ハードコピー形式の情報を UIM によって表示されるものと似たフォーマットで提供しますが、一般的な印刷を UIM を使用して行うことはできません。

アプリケーション開発者が印刷パネルおよび表示パネルを提供する必要はありません。ただし、印刷パネルと表示パネルの両方を同じパネル・グループに入れることが勧められています。パネルの印刷には、Print Panel (QUIPRTP) API を使用します。

パネル・グループを、表示または印刷のどちらかのために開くことができます。パネル・グループを表示のために開いた後に、それを使用して印刷出力を生成することもできます。すでに開いた表示アプリケーションで印刷ファイルを開くために、Add Print Application (QUIADDP) API が提供されています。Remove Print Application (QUIRMVPA) API は、QUIADDP API によって開いた印刷装置ファイルを閉じます。

API について詳しくは、iSeries Information Center の『API』のトピックを参照してください。

UIM タグ言語は、印刷ヘッド・パネルおよび印刷パネルを定義するために使用されます。印刷ヘッド・パネルは印刷ヘッド (PRTHEAD) タグによって作成され、各ページの最上部に印刷されるヘッダー情報を定義します。印刷パネルは印刷パネル (PRTPNL) タグによって作成され、印刷するさまざまなパネル領域を定義します。印刷ヘッド・パネルと印刷パネルの両方を、同じオープン・アプリケーションに対して印刷する必要があります。

- | ヘルプ・テキストを作成する際に以下の点について考慮してください。
- | • いくつかのタグは、テキストが印刷域に収まらない場合、テキストの行を切り捨てます。
- | • 他のタグ (XMP など) では、そのテキストを次の行に折り返します。
- | • UIM は 72 桁のデータだけを許可します。
- | タグをネストする場合、ネストされる行の幅も計算に入れる必要があります。たとえば、順不同のリストを
- | 定義してそのリスト項目のいずれかに例を組み込む場合、その例のテキストは、リスト項目のテキストの右
- | 側に 2 バイト字下げされます。

以下に示すいくつかの領域は印刷リストを構成するものであり、その後を示す例は印刷結果を示しています。

タイトル

タイトルは、各ページの最上部に印刷される 2 つの行から構成されます。タイトル行には、出力タイトル、時間および日付、システム名、製品情報、ページ番号などが含まれます。タイトル行の情報はすべて、印刷ヘッド・パネルによって定義されます。

プロローグ

(オプション。) プロローグ・セクションは最初のページだけに印刷されて、タイトル行の直後に配置されます。プロローグ・セクションは、出力の印刷に指定されたパラメーターを示すなどのために使用されます。プロローグ・セクションは、TYPE=PROLOG がデータ表示域 (DATA) または情報域 (INFO) タグに指定された印刷ヘッド・パネルの一部として定義されます。

ヘッダー

(オプション。) ヘッダー・セクションは、各ページに同じデータを印刷するために使用されます。このデータには通常、印刷される情報のファイル名、ライブラリー、メンバー名などが含まれます。このセクションは先頭ページのプロローグの後、および残りのすべてのページのタイトル行の

後に配置されます。ヘッダー・セクションは、TYPE=NORMAL が指定された、またはデフォルトでそうになっている、印刷ヘッド・パネルの一部として定義されます。

ページ本体

ページ本体は、1 つ以上の情報パネルから構成されます。これらの情報パネルは、ディスプレイ上に情報を表示するために使用されるパネルととてもよく似ています。これらのパネルには、データ、情報、およびリスト域が含まれることがあります。メニュー域および選択フィールドは、印刷のために定義できません。スクロール可能な印刷パネルなどは存在しないので、パネルに関連したすべての情報およびデータは Print Panel (QUIPRTP) API によって印刷されます。

トレーラー

トレーラーは、最後のページの最後のパネル領域の後に印刷される単一行のテキストです。このトレーラーは、通常は以下のような内容のものです。

END OF LISTING (リストの終わり)

または

END OF SOURCE (ソース仕様の終わり)

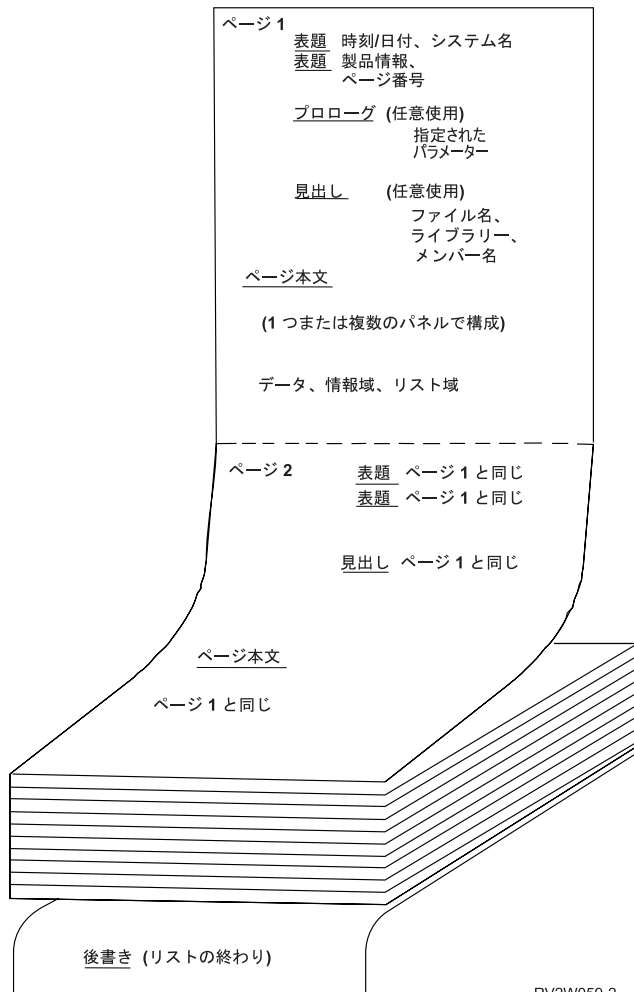


図 114. 印刷出力の例

印刷ヘッド・パネルの印刷

Print Panel (QUIPRTP) API を呼び出して印刷ヘッド (PRTHEAD) パネルを印刷するとき、ヘッディングはフォーマット設定されて、印刷パネルが印刷されるまで保管されます。

印刷パネルの印刷

Print Panel (QUIPRTP) API を呼び出して印刷パネル (PRTPNL) を印刷するとき、UIM は以下を行います。

1. データの 1 ページをフォーマット設定して、それを印刷します。
2. データの次のページをフォーマット設定して、それを印刷します。
3. すべてのページが印刷されるまで、ページごとにフォーマット設定します。
4. パネルの最後の部分をフォーマット設定して、部分ページに印刷します。次に QUIPRTP API を呼び出すとき、ページ排出が要求されていなければ、部分ページの残りの部分がフォーマット設定されて印刷されます。この次のパネルの最初の領域が 1 ページに収まらない場合、この領域は次のページに印刷されます。これにより、パネル・タイトル、最上部の区切り記号、そして (存在すれば) 領域タイトルが、次のページに印刷されます。

分離のためブランク行の使用

UIM によって単一のブランク行が以下の場所に挿入されます。

- 各ページの 2 番目のタイトル行の後
- リスト列およびグループ見出しの後
- データ列見出しの後
- トレーラー行の前

フォントおよび強調表示

UIM が使用できるのは、DDS ソースを使用しないで作成されたプリンター・ファイルだけです。特殊なフォントはサポートされていません。H1 から H4 までのヘッディング・タグは、強調表示またはアンダーラインのない通常のテキストとして印刷されます。H1 ヘッディングは、中央寄せされます。

HP0 から HP9 までの語句の強調表示タグは、印刷パネル定義では許可されていません。強調表示、アンダーライン、および二重印字は、サポートされていません。(二重印字とは、印刷装置がある文字を印刷した後、同じ位置に同じ文字または別の文字を印刷することです。)

トレーラーの印刷

1 行トレーラーは、印刷ヘッド・パネルでトレーラー・メッセージ印刷 (PRTRAIL) タグを使用して指定されます。これは、次のような 1 行のステートメントです。'END OF LISTING'。これは、Remove Print Application (QUIRMVPA) または Close Application (QUICLOA) API が呼び出される時、アプリケーションの印刷出力の最後に印刷されます。

プロローグ域の定義

印刷ヘッド・パネル (PRTHEAD) は、プロローグおよびヘッダー・セクションの定義に使用されます。印刷ヘッド・パネルには、情報およびデータ域だけを含めることができます。リスト域は許可されません。

TYPE=PROLOG が指定されたプロローグ・セクションは、1 つ以上のデータ域、1 つ以上の情報域、またはそれらの組み合わせから構成されます。プロローグ・セクションが印刷されるのは、開いた印刷装置ファイルごとに印刷される最初の印刷ヘッド・パネルからだけです。プロローグ・セクションを含む印刷ヘッ

ド・パネルを印刷しようとする、開いたプリンター・ファイルに対してその印刷ヘッド・パネルが最初に印刷されるものでない場合に、エスケープ・メッセージが出されます。

ヘッダー域の定義

TYPE=NORMAL が指定されたヘッダー・セクションは、1 つ以上のデータ域、1 つ以上の情報域、またはそれらの組み合わせから構成されます。ヘッダー・セクションに含まれる情報およびデータ域の長さの合計は、6 行以下でなければなりません。

最小のページの長さは 18 行です。最初のページは例外となることもありますが、各ページにデータを入れる場所があることを保証するために、UIM はプロローグ・セクションおよびヘッダー・セクションの合計の長さを 14 行に制限します。これらの最大値が守られない場合、コンパイル時のエラー・メッセージが生じます。

すべてのページの最上部にタイトル行を印刷する必要があるため、UIM では印刷ヘッド・パネルを他の印刷パネルよりも先に印刷することが必要です。Print Panel (QUIPRTP) API をプリンター・ファイルごとに印刷ヘッド・パネルに対して 1 度だけ呼び出す必要があります。その後、同じヘッダー情報がプリンター・ファイルの各ページに印刷されます。ヘッダー情報を変更するには、同じまたは異なる印刷ヘッド・パネルに対して、QUIPRTP API を再度呼び出します。

印刷中のページ排出機能の使用方法

自動ページ排出は、印刷パネルを印刷した後に印刷ヘッド・パネルを印刷したとき生じます。ただし、印刷パネル (PRTPNL) タグに排出が指定されている場合でも、次の印刷パネルが印刷されるときに 2 番目のページ排出は生じません。

印刷ヘッド・パネルが Print Panel (QUIPRTP) API によって処理される時、別の QUIPRTP API が印刷パネルに対して呼び出されるまで実際の印刷は行われません。そのため、5 つの異なる印刷ヘッド・パネルに対して QUIPRTP API が呼び出された後で印刷パネルに対して QUIPRTP API が呼び出される場合、最後の印刷ヘッド・パネルだけがリスト作成に影響を与えます。

プリンター・ファイルの共用およびオーバーライド

UIM では、プリンター・ファイルを共用のために開くことにより、複数の UIM アプリケーションが同じプリンター・ファイルに対して印刷することができます。ただし、UIM は複数の UIM アプリケーション間で情報を共用しません。

複数の UIM アプリケーションが同じプリンター・ファイルに対して印刷する場合、プロローグ・セクションがある場合にそれはオープン・アプリケーションごとに 1 回ずつ印刷されるので、プリンター・ファイルごとに複数のプロローグ・セクションがある可能性があります。

共用ファイルを扱うときには、プログラム・スタック内にあるより高位のまたは直前のプログラムと、不注意により共用することがないように注意する必要があります。UIM アプリケーションと DDS レコード・フォーマット・ファイルを使用する DDS レコード様式ファイルとでプリンター・ファイルを共用することはできません。共用ファイルを共用する最後のアプリケーションが、Close Application (QUICLOA) または Remove Print Application (QUIRMVPA) API を使用するとき、共用ファイルは閉じられます。

プリンター・ファイルが開くと、ほとんどのオーバーライドが可能になります。

2 バイト文字セット (DBCS) 印刷の考慮事項

- | UIM はパネル・グループ (PNLGRP) タグの TXTMODE 属性を調べて、2 バイト・データを印刷することになるかどうかを判別します。その可能性がある場合、UIM はプリンター・ファイルを開いて DBCS データを含めます。DBCS についての詳細は、iSeries™ Information Center の 2 バイト文字セットのサポートを参照してください。

UIM に関するよく尋ねられる質問

出口プログラムおよび呼び出し先プログラムに関連した、いくつかのよく尋ねられる質問を以下に示します。

- 出口プログラムおよび呼び出し先プログラムとは何ですか。
 - 出口プログラム -- UIM は、通常の処理の完了後にさらにデータを処理する必要があるとき、呼び出されるプログラムをユーザーが定義できるようにします。たとえば、汎用パネル出口により、UIM 機能が行うものよりも明確な妥当性検査をパネルに対して行うことができます。
 - 呼び出し先プログラム -- 呼び出し先プログラムは、機能キー、プルダウン・オプション、リスト・オプションを処理するときなどに UIM が呼び出すことのできる、ユーザーによって記述されたプログラムです。この処理には、リスト・パネルの最新表示、より詳細なパネルを表示するための他のプログラムの呼び出し、または入力された情報によるファイルの更新が含まれることがあります。
- 出口プログラムと呼び出し先プログラムの違いは何ですか。
 - 出口プログラムは通常の処理の完了後に呼び出され、呼び出し先プログラムは通常の処理の一部として呼び出されます。通常の処理とは、LISTACT タグでの ACTION、ENTER、および PROMPT 操作などの処理項目のことです。

第 4 部 ヘルプ表示のプログラミング

第 18 章 ディスプレイ・ファイルのオンライン・ヘルプ情報をアクセス可能にする	403
ヘルプ・キーの使用可能化	404
ヘルプにパネル・グループとレコードのいずれかを選択する	404
オンライン・ヘルプ情報を必要とする画面領域の定義	405
ディスプレイ・ファイルでヘルプ用にパネル・グループを指定する	407
オプション標識を使用したパネル・グループの定義	410
パネル・グループを使用したヘルプを指定する	411
QUSRTOOL 例のコピー	411
ディスプレイ・ファイルでレコードを指定する	411
オプション標識を使用したレコードの定義	412
ヘルプ情報が含まれるレコードの入力	413
同じディスプレイ・ファイルにおけるヘルプ用のレコードおよび文書の使用	413
レコードに関する制約事項の理解	414
レコードを使用するヘルプ画面間でのページング	415
システムがヘルプ画面をページングする方法についての理解	415
2 次オンライン・ヘルプ情報の表示	416
ヘルプ・キーを押した後に制御をプログラムに戻す	418
ヘルプ画面を表示した後に制御をプログラムに戻す	419
ヘルプ画面を表示せずに制御をプログラムに戻す	420

第 19 章 パネル・グループのオンライン・ヘルプを利用できるようにする	421
定義と説明	421
索引検索へのヘルプ・パネル・グループ・アクセス権を与える	423
ユーザー定義パネル・グループへのヘルプ・パネル・グループ・アクセス権を与える	423
「F18=続きの見出し」へのアクセス権を除去する	424
リスト域のヘルプ	424
ヘルプのコーディング	425
メニュー域のヘルプ	426
ヘルプのコーディング	426
データ域のヘルプ	428
ヘルプのコーディング	429
メニュー・バー域のヘルプ	431
ヘルプのコーディング	432
機能キー域のヘルプ	433
ヘルプのコーディング	434

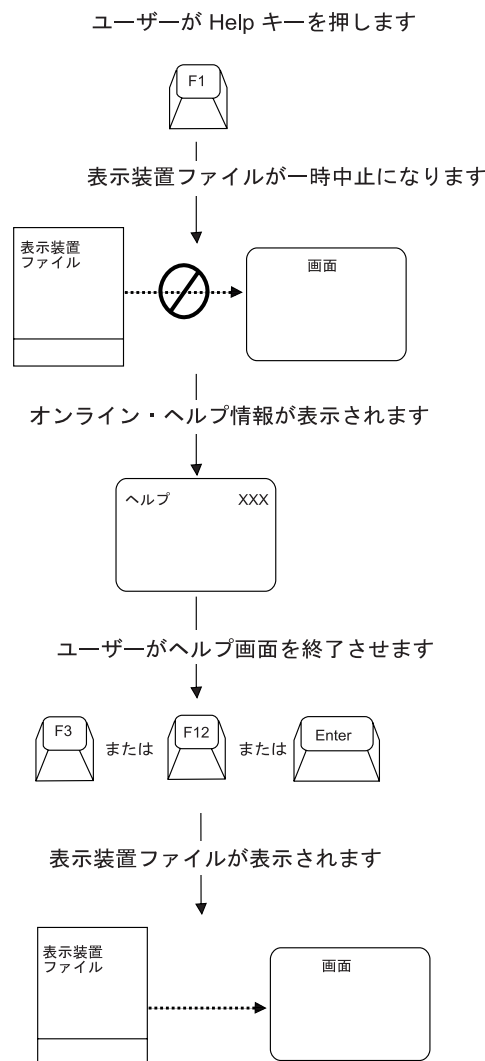
第 20 章 オンライン・ヘルプ情報の定義	437
パネル・グループ中のオンライン・ヘルプ情報の定義	437

ヘルプ用にパネル・グループの UIM ソースを入力する	437
ヘルプ・モジュールを使用したパネル・グループの編成	438
ヘルプ・モジュール中の情報を複数回使用する	438
別のヘルプ・パネル・グループに含まれるヘルプ・モジュールの使用	438
ヘルプ・モジュール中のテキストの強調とフォーマット設定	439
ヘルプ・パネル・グループへの索引検索タグの追加	442
ヘルプ・モジュールのリンク	448
パネル・グループの作成および削除	450
パネル・グループをコマンドのヘルプとして割り当てる	450
検索索引でのパネル・グループの使用法	451
検索索引の作成	451
検索索引への項目の追加	451
検索索引からの項目の除去	451
検索索引の削除	452
パネル・グループ中のヘルプを定義する	452
QUSRTOOL 例のコピー	452
DDS レコードでのオンライン・ヘルプ情報の定義	452

第 18 章 ディスプレイ・ファイルのオンライン・ヘルプ情報をアクセス可能にする

アプリケーション・ヘルプは、システムにより提供される機能であり、オンライン・ヘルプ情報をディスプレイ・ファイルのレコードに定義して、ヘルプ・キーが押された時にそのヘルプをユーザーに表示することを可能にします。

以下の図は、オンライン・ヘルプ情報が使用可能およびアクセス可能な状態でヘルプ・キーを押した場合に何が起こるかを示しています。



RV2W018-4

注: ファイルに対して獲得されている複数のディスプレイ装置でアプリケーション・ヘルプを使用する際には、注意が必要です。これは、すべてのディスプレイ装置が、アプリケーション・ヘルプの表示が完了するまで待機するためです。

ディスプレイにオンライン・ヘルプ情報を表示させるには、以下のことを行う必要があります。

- 必要な DDS キーワードをディスプレイ・ファイルに追加して、ご使用のディスプレイでオンライン・ヘルプ情報にアクセスできるようにします。この章では、これらのキーワードの追加について説明します。
- ユーザーが、パネル・グループ、文書、またはレコードのいずれかを使用して参照するヘルプ情報を定義します。437 ページの『第 20 章 オンライン・ヘルプ情報の定義』では、ヘルプ情報の定義に関して説明します。

- 1 ヘルプ・テキストを作成する際に以下の点について考慮してください。
 - 1 • 一部のタグでは、テキストが表示域に収まらない場合にテキストの行が切り捨てられます。
 - 1 • 他のタグ (XMP など) では、そのテキストを次の行に折り返します。
 - 1 • 画面のサイズ (24 x 80 または 27 x 132) は、使用できるカラム数を決定します。
- 1 タグをネストする場合、ネストされる行の幅も計算に入れる必要があります。たとえば、順不同のリストを
- 1 定義してそのリスト項目のいずれかに例を組み込む場合、その例のテキストは、リスト項目のテキストの右
- 1 側に 2 バイト字下げされます。

DDS キーワードについて

この章で説明されている DDS キーワードについて詳しくは、iSeries Information Center にある『DDS』トピックを参照してください。

ヘルプ・キーの使用可能化

ご使用のディスプレイでオンライン・ヘルプ情報を使用できる場合、キーボードでヘルプ・キーを使用可能にする必要があります。ヘルプ・キーを使用可能にするには、DDS ソースのファイルまたはレコード・レベルで HELP キーワードを指定します。

ヘルプにパネル・グループとレコードのいずれかを選択する

以下のいずれかを使用してオンライン・ヘルプ情報を定義できます。

表 34. オンライン・ヘルプ情報を定義するさまざまな方法

方式	説明
パネル・グループ	ユーザー・インターフェース管理機能 (UIM) ソースが入力されるオブジェクト
レコード	ソース・ファイル・メンバーに含まれる DDS キーワードのセット

それぞれの方式には、自分に最も適した方式を選択するのに役立つ特性があります。以下の表では、これらの特性と、それが適用される方式をリストします。

表 35. オンライン・ヘルプ情報に関する異なる方式の特性

特性	パネル・グループ	レコード
H 仕様を使用して DDS 記述画面にアクセス可能	○	○
カーソル検知可能ヘルプが使用可能	○	○
全般ヘルプが使用可能	○	○
索引検索機能が使用可能	○	
ウィンドウが使用される	○	

表 35. オンライン・ヘルプ情報に関する異なる方式の特性 (続き)

特性	パネル・グループ	レコード
ハイパーテキスト・リンクが使用可能	○	
システム表示画面で使用される	○	
コマンド・ヘルプに使用可能	○	
スペル検査などのワード処理機能が使用可能	○ (注を参照。)	
ブックマークおよび出口機能が使用可能 (次にヘルプ・キーを押した際にヘルプの同じ場所に戻ることができる)		
ヘルプ情報を印刷できる	○	○
表示されている情報の索引が使用可能		
表示されている情報の目次が使用可能		
表示ソースとして同じメンバー内に配置できる		○
表示ソースと同じソース・タイプ		○
QUSRTOOL 例が使用可能	○	
オプション標識が使用可能	○	○
DBCS オンライン・ヘルプ情報をサポートする	○	○
テキストの右から左および左から右の方向付け	○	
獲得される装置のサポート		○

オンライン・ヘルプ情報を必要とする画面領域の定義

ディスプレイ・ファイルを定義して、以下の図にある陰影域で表されているように、画面のすべてまたは一部でオンライン・ヘルプ情報を表示することができます。



1

2

RV2W020-2

- 1 文脈ヘルプ: レコードまたはレコード様式の一部によって定義される長方形の領域の情報。文脈情報は、画面上の 1 つ以上のフィールドについて説明します。
- 2 全般ヘルプ: 画面の目的と使用方法に関する情報から始まり、続いて各フィールドの文脈ヘルプを含む、画面全体に関するすべての情報。文脈ヘルプが定義されていない画面領域でヘルプ・キーを押すと、全般ヘルプが表示されます。ご使用のオンライン・ヘルプがパネル・グループを使用して定義されている場合、F2 を押して文脈ヘルプから全般ヘルプに到達することもできます。

カーソル位置は、ヘルプ・キーを押したときに決定されます。ヘルプ・キー使用可能であり、カーソルがアクティブなヘルプ域にある場合、そのヘルプ域に関連したオンライン・ヘルプ情報が表示されます。

画面上の各ヘルプ域は、以下のものを使用して DDS ソース内で定義されます。

H 仕様

それを含むレコードのヘルプを定義します。 DDS ソースの列 17 にある H はヘルプ仕様レベルを示しています。ヘルプ仕様は、レコード内の最初のフィールドの前に定義されます。

ヘルプ域 (HLPARA) キーワード

長方形の領域の左上の行/列および右下の行/列を指定することによって定義します。これらの座標は画面区域内になければなりません、レコード域にある必要はありません。

HLPARA キーワードに *RCD を指定すると、ヘルプはレコード域全体に関連付けられます。これには、レコードによって占められる行のすべての列が含まれます。カーソルがレコード域にある状態でヘルプ・キーを押すと、HLPRCD キーワード上のレコードからのオンライン・ヘルプ情報が表示されます。

ヘルプ (HLPxxx) キーワード

パネル・グループ (HLPPNLGRP)、文書 (HLPDOC)、またはレコード様式 (HLPRCD) に、ヘルプ・キーを押した際に表示される実際のオンライン・ヘルプ情報を含めるかどうかを定義します。

以下の図では、H 仕様を DDS ソース内に入力する方法を示します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
          H                               HLPARA(*RCD)
                                         HLPxxx(required-variables)
```

図 115. DDS ソース内の H 仕様のサンプル

このサンプル DDS はレコード用のヘルプを定義します。ヘルプ・キーワードに必要な変数は、HLPxx が HLPPNLGRP、HLPDOC、または HLPRCD キーワードとして定義されるかどうかによって異なります。ファイルに対して定義できるヘルプ・キーワードは 1 つだけですが、レコードに対しては 1 つ以上の H 仕様を指定できます。

見つかった最初の一致が選択されるため、H 仕様の順序は重要です。このため、HLPARA のより具体的な位置を最初にリストする必要があります。

以下のサンプル画面は、2 つのレコード様式 (HEADER および SINFO) を使用して定義されています。

```
1
2
3
4
5           Add, Update, Display Supplier
6
7           Enter new or existing name: _____
8
9
10
11
12
13
14
15          Address: _____
16          City:   _____
17          State:  _____
18
19
20
21
22
23
24
```

レコード
HEADER
(5~7 行目)

レコード
SINFO
(15~17 行目)

RV2W022-3

以下の表では、サンプル画面でオンライン・ヘルプ情報を定義する方法の 1 つを示します。

表 36. サンプル画面のヘルプ

画面上の位置	DDS ソースでの位置	提供されるヘルプの説明
行 5-7	レコード様式 HEADER 全体	ヘルプがレコード全体に対して指定されている
行 15	レコード様式 SINFO の最初の行	ヘルプが行に対してのみ指定されている
行 16	レコード様式 SINFO の 2 番目の行	ヘルプが行に対してのみ指定されている
行 17	レコード様式 SINFO の 3 番目の行	ヘルプが行に対してのみ指定されている
残りの行	レコード様式を使用して定義されていないディスプレイ・ファイル内の領域	ヘルプがファイル全体に対して指定されている

使用するヘルプ方式に応じて、以下のいずれかを継続して DDS ソース内で H 仕様を定義します。

パネル・グループ (UIM)

『ディスプレイ・ファイルでヘルプ用にパネル・グループを指定する』

レコード (DDS)

411 ページの『ディスプレイ・ファイルでレコードを指定する』

オンライン・ヘルプ情報を作成するさまざまな方法を比較対照するには、404 ページの『ヘルプにパネル・グループとレコードのいずれかを選択する』を参照してください。

ディスプレイ・ファイルでヘルプ用にパネル・グループを指定する

DDS 内の HLPPNLGRP キーワードは、画面のオンライン・ヘルプ情報が含まれるパネル・グループを示します。HLPPNLGRP キーワードは、H 仕様またはファイル・レベルで指定することができます。HLPPNLGRP キーワードを使用するには、ヘルプ・モジュール名、パネル・グループ名、およびそのヘルプ・モジュールが含まれるライブラリーの名前を知っている必要があります。

他のいくつかの DDS 機能がパネル・グループで使用可能です。それらの機能とそれに関連した DDS キーワードが、以下の表で説明されています。

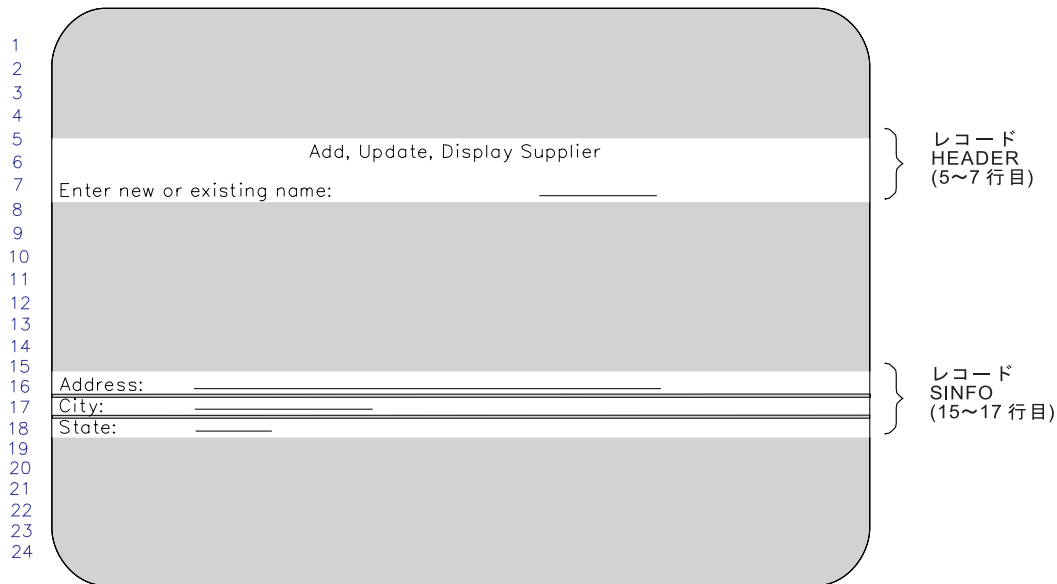
表 37. UIM ヘルプ用の他の DDS キーワード

機能	DDS キーワード	説明
ヘルプ画面の名前を定義する	HLPTITLE	ヘルプ画面の最初の行に表示されるテキストは、ファイルまたはレコード・レベルの DDS キーワード HLPTITLE (ヘルプのタイトル) を使用して定義されます。このテキストは、ヘルプ・キーを押した際に表示される画面の名前であるはずですが、HLPTITLE キーワードはディスプレイ・ファイル内で使用する必要があります。このキーワードは、ヘルプのタイトルをヘルプ・ソースで指定しなかった場合、ヘルプの全画面表示でのみ使用されません。

表 37. UIM ヘルプ用の他の DDS キーワード (続き)

機能	DDS キーワード	説明
全画面のオンライン・ヘルプ情報を示す	HLPFULL	<p>DDS ファイル・レベル HLPFULL (ヘルプの全画面表示) キーワードを使用すると、アプリケーションの UIM 定義オンライン・ヘルプ情報は、ウィンドウではなく全画面の代替画面に表示されます。</p> <p>HLPFULL キーワードを指定しない場合、ユーザーのプロファイルが別のキーワードを指定していなければ、ヘルプはウィンドウに表示されます。</p>
2 次ヘルプ情報としてパネル・グループ・ヘルプを除外する	HLPEXCLD	<p>HLPEXCLD キーワードは、重複した名前を持つヘルプ・パネル・グループが全般ヘルプで表示されないようにします。これを行うには、同一のヘルプ・パネル・グループに名前を付ける最初の H 仕様以外のすべてに HLPEXCLD を配置します。HLPEXCLD は、H 仕様に関連した情報を全般ヘルプの一部として表示しないことを示しています。</p> <p>HLPEXCLD キーワードが指定されていない場合、全般ヘルプは、ファイル・レベルの HLPPNLGRP キーワード (もしあれば) と、アクティブなすべての H 仕様にある HLPPNLGRP キーワードの両方に関連した情報で構成されます。</p> <p>少なくとも 1 つのヘルプ・パネル・グループ名が HLPEXCLD キーワードを指定していません。</p>
索引検索機能の使用可能化	HLP SCHIDX	<p>索引検索機能が使用可能になり、この機能で使用される検索索引オブジェクトがファイル・レベルの DDS キーワード HLP SCHIDX (ヘルプの索引検索) で指定されます。</p>

405 ページの『オンライン・ヘルプ情報を必要とする画面領域の定義』でサンプル画面を使用する場合、以下のオンライン・ヘルプ情報が各ヘルプ域ごとに定義されます。



RV2W023-5

表 38. パネル・グループを使用したサンプル画面のヘルプ

画面上の位置	DDS ソースでの位置	ライブラリー SMPLIB 内のパネル・グループ SMPPNL にあるヘルプ・モジュールの名前
行 5-7	レコード様式 HEADER 全体	HLPCMPY
行 15	レコード様式 SINFO の最初の行	HLPADDR
行 16	レコード様式 SINFO の 2 番目の行	HLPCITY
行 17	レコード様式 SINFO の 3 番目の行	HLPST
すべての陰影域	レコードを使用して定義されていないディスプレイ・ファイル内の領域	SUPHELP

以下の DDS ソースは、パネル・グループを使用してサンプル画面にヘルプ域を定義する方法を示しています。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
                                HELP
                                HLPPNLGRP(SUPHELP SMPLIB/SMPPNL)
                                HLPTITLE('ADD,UPDATE,DISPLAY SUPPLIER')
R HEADER                          OVERLAY
H                                HLPARA(*RCD)
                                HLPPNLGRP(HLPCMPY SMPLIB/SMPPNL)
                                5 25'ADD, UPDATE, DISPLAY SUPPLIER'
                                7 10'ENTER NEW OR EXISTING NAME:'
CONAME                            10A I 7 47
R SINFO                          OVERLAY PROTECT
H                                HLPARA(15 1 15 79)
                                HLPPNLGRP(HLPADDR SMPLIB/SMPPNL)
H                                HLPARA(16 1 16 79)
                                HLPPNLGRP(HLPCITY SMPLIB/SMPPNL)
H                                HLPARA(17 1 17 79)
                                HLPPNLGRP(HLPST SMPLIB/SMPPNL)
                                15 10'ADDRESS:'
ADDR                              30A B 15 32
                                16 10'CITY:'
CITY                              10A B 16 32
                                17 10'STATE:'
STATE                             2A B 17 32

```

図 116. HLPPNLGRP を示すサンプル DDS ソース

SUPHELP 内のファイル・レベルのヘルプは、画面の全般ヘルプを提供します (カーソルが HEADER または SINFO のいずれかのレコード域にない場合)。これは、HLPARA(*RCD) が、H 仕様で指定された場所であるためです。

オンライン・ヘルプ情報が含まれるパネル・グループは、ソース・ファイル・メンバーから UIM ソースを使用して作成する必要があります。パネル・グループの作成に関する詳細情報は、437 ページの『パネル・グループ中のオンライン・ヘルプ情報の定義』にあります。

注: HLPPNLGRP と HLPDCD は同じディスプレイ・ファイルで使用することはできません。HLPPNLGRP と HLPDOC も同じディスプレイ・ファイルで使用することはできません。

オプション標識を使用したパネル・グループの定義

HLPPNLGRP キーワードは、オプション標識を使用して指定できます。以下の例では、SINFO レコードで標識 90 がオンになっており、カーソルが H 仕様に定義されているヘルプ域にあることを前提としています。ヘルプ・キーを押すと、パネル・グループ HELP1 が表示されます。標識 90 がオフになっていると、パネル・グループ HELP2 が表示されます。

このファイルの DDS は次のとおりです。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
                                HELP
                                OVERLAY
                                5 25'ADD, UPDATE, DISPLAY SUPPLIER'
                                7 10'ENTER NEW OR EXISTING NAME'
R HEADER
                                CONAME      10A  I  7 47
R SINFO
H                                OVERLAY PROTECT
                                HLPARA(15 9 15 61)
90                                HLPNNLGRP(HELP1 SMPLIB/SMPPNL)
                                HLPARA(15 9 15 61)
N90                                HLPNNLGRP(HELP2 SMPLIB/SMPPNL)
                                15 10'ADDRESS'
                                ADDR      30A  B 15 32

```

図 117. HLPPNLGRP およびオプション標識を示すサンプル DDS ソース

パネル・グループを使用したヘルプを指定する QUSRTOOL 例のコピー

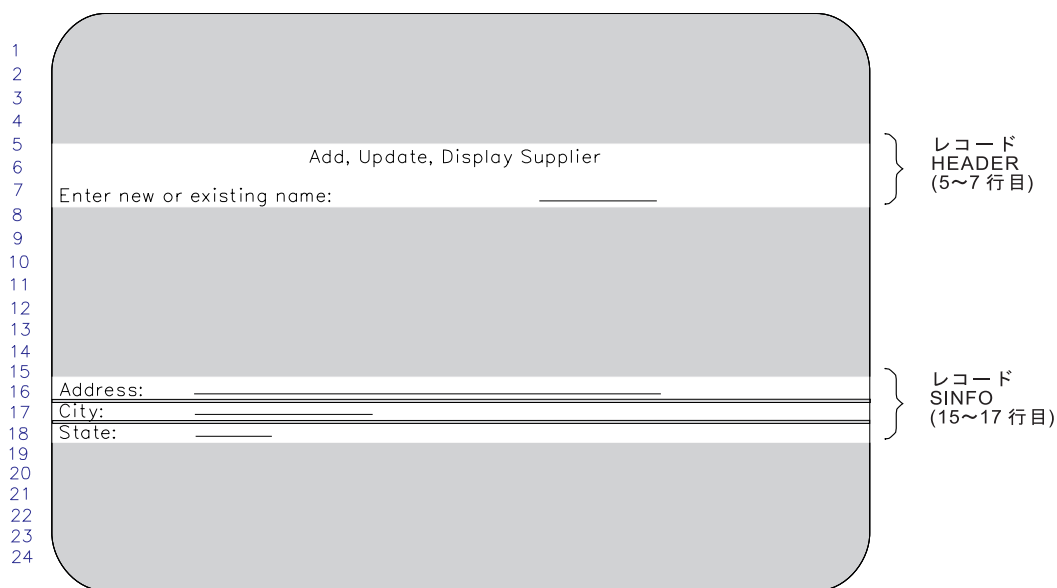
QUSRTOOL ライブラリーは、パネル・グループを使用してオンライン・ヘルプ情報にアクセスする DDS 記述の画面のサンプルを 4 つ提供します。選択したライブラリーにこれらのサンプル画面のソースをコピーしてから、独自の使用法に合うように調整できます。これらのサンプル画面についての詳細は、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

ディスプレイ・ファイルでレコードを指定する

DDS ソース内のヘルプ・レコードによって識別されるオンライン・ヘルプ情報のソースは、ヘルプ仕様またはファイル・レベルの DDS キーワード HLPRCD (ヘルプ・レコード) を使用して指定されます。

HLPRCD と HLPPNLGRP は同じディスプレイ・ファイルで使用することはできません。

405 ページの『オンライン・ヘルプ情報を必要とする画面領域の定義』でサンプル画面を使用する場合、以下のオンライン・ヘルプ情報が各ヘルプ域ごとに定義されます。



RV2W023-5

表 39. HLPBCD を使用したサンプル画面のヘルプ

画面上の位置	DDS ソースでの位置	ヘルプが含まれるレコード様式の名前
行 5-7	レコード様式 HEADER 全体	レコード様式 HLPBCMPY
行 15	レコード様式 SINFO の最初の行	レコード様式 HLPADDR
行 16	レコード様式 SINFO の 2 番目の行	レコード様式 HLPBCITY
行 17	レコード様式 SINFO の 3 番目の行	レコード様式 HLPST
すべての陰影域	ヘルプ・レコードを使用して定義されていないディスプレイ・ファイルの領域	レコード様式 SUPHELP

以下の DDS ソースは、ヘルプ・レコードを使用してサンプル画面にヘルプ域を定義する方法を示しています。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
                                HELP
                                HLPBCD(SUPHELP)
                                OVERLAY
R  HEADER                        HLPBCD(*RCD)
H                                HLPBCD(HLPBCMPY)
                                5 25'ADD, UPDATE, DISPLAY SUPPLIER'
                                7 10'ENTER NEW OR EXISTING NAME:'
                                CONAME      10A  I  7 47
R  SINFO                        OVERLAY PROTECT
H                                HLPBCD(15 1 15 61)
                                HLPBCD(HLPADDR)
H                                HLPBCD(16 1 16 41)
                                HLPBCD(HLPBCITY)
H                                HLPBCD(17 1 17 33)
                                HLPBCD(HLPST)
                                15 10'ADDRESS:'
ADDR      30A  B 15 32
                                16 10'CITY:'
CITY      10A  B 16 32
                                17 10'STATE:'
STATE     2A  B 17 32

```

図 118. Sample DDS Source Showing HLPBCD

ファイル・レベルのヘルプは、カーソルが HEADER または SINFO のいずれかで定義されたヘルプ域にならない場合、画面の一般ヘルプを提供するのに使用されます。

異なるレコードを使用して各ヘルプ域を定義する必要はありません。これは、画面上で 1 つ以上のヘルプ域 (画面全体を定義するヘルプ域を含む) を定義するのに同じレコードを使用できることを意味します。

オプション標識を使用したレコードの定義

HLPBCD キーワードは、オプション標識を使用して指定できます。以下の例では、SINFO レコードが標識 90 がオンの状態で書き込まれており、カーソルが H 仕様に定義されているヘルプ域にあることを前提としています。ヘルプ・キーを押すと、レコード HELP#1 が表示されます。標識 90 がオフになっていると、レコード HELP#2 が表示されます。

このファイルの DDS は次のとおりです。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
                                HELP
                                OVERLAY
                                5 25'ADD, UPDATE, DISPLAY SUPPLIER'
                                7 10'ENTER NEW OR EXISTING NAME'
                                CONAME      10A I 7 47
R  HEADER
R  SINFO      OVERLAY PROTECT
H            HLPARA(15 9 15 61)
90          H            HLPARCD(HELP#1)
N90         H            HLPARA(15 9 15 61)
                                HLPARCD(HELP#2)
                                15 10'ADDRESS'
                                ADDR      30A B 15 32

```

図 119. HLPARCD およびオプション標識を示すサンプル DDS ソース

ヘルプ情報が含まれるレコードの入力

実際のオンライン・ヘルプ情報を提供するレコードを、アプリケーション画面用の DDS ソースと同じファイルに組み込むことができます。レコードは、別のファイルに含めることもできます。レコードが別のファイルに含まれている場合、そのファイルとファイルを含むライブラリーは HLPARCD キーワード上で識別されなければなりません。

情報が入っている場所にかかわらず、前の例の DDS ソースは以下のようになります。

```

|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
R  SUPHELP
                                2 5'TO ADD, UPDATE, OR DISPLAY -
                                THE SUPPLIER NAME AND -
                                ADDRESS, ENTER THE SUPPLIER -
                                NAME. ITS CURRENT ADDRESS, -
                                IF ANY, WILL BE SHOWN AND MAY -
                                BE UPDATED.'
R  HLPMPY
                                2 5'ENTER THE FIRST 10 -
                                CHARACTERS OF THE COMPANY'
R  HLPADDR
                                2 5'ENTER THE NEW OR CHANGED -
                                ADDRESS.'
R  HLPCITY
                                2 5'ENTER THE FIRST 10 -
                                CHARACTERS OF THE CITY'
R  HLPST
                                2 5'ENTER THE 2 CHARACTER ABBR -
                                FOR THE STATE.'

```

図 120. HLPARCD を使用したサンプル DDS ソース

同じディスプレイ・ファイルにおけるヘルプ用のレコードおよび文書の使用

HLPARCD と HLPDOC は同じディスプレイ・ファイルで使用することができます。ただし、以下の考慮事項が適用されます。

- カーソル位置が文書で記述されているヘルプ域の中にある状態でヘルプ・キーを押すと、その文書は画面移動キーを使用する際に表示される唯一のヘルプになります。他の文書またはヘルプ・レコードは表示されません。

- カーソル位置がレコードで記述されているヘルプ域の中にある状態でヘルプ・キーを押すと、文書が無視した状態で、レコードに使用される通常の順序付けが続けられます。
- HLPDOC および HLPRCD キーワードは、ファイル・レベルまたは同じ H 仕様で両方とも指定することはできません。
- HLPBDY キーワードは、HLPDOC キーワードとともに指定することはできません。HLPBDY キーワードについての詳細は、416 ページの『2 次オンライン・ヘルプ情報の表示』を参照してください。

レコードに関する制約事項の理解

以下の制約事項は、アプリケーション・ヘルプのレコード様式を使用する際に適用されます。

- アプリケーション・プログラムはレコードの表示を制御しません。ヘルプ・キーを押すと、システムによりヘルプの表示が制御されます。システムは、アプリケーション・プログラムのバッファまたは非表示のメッセージ行の領域を制御しないため、以下のことが起こります。
 - レコード様式の出カフィールドはブランクで表示されます。
 - ALIAS キーワードは使用可能ですが無視されます。
 - CSRLOC および MSGID はシステムにより処理されますが、隠しフィールド、およびプログラムシステム間フィールドはアプリケーション・プログラムから渡されません。
- ユーザー・プログラムに戻される入力はありません。
 - レコード上の入力可能なフィールドは下線付きで表示されます。ただし、アプリケーション・プログラムに戻される入力はありません。入力フィールドに入力されたすべての入力は、アプリケーション・ヘルプの完了時に失われます。
 - 応答標識は戻されません。
- ヘルプ・レコードに H 仕様を含めることはできますが、それらは無視されます。
- オプション標識はオフになっていると見なされます。
- 画面サイズの条件付けを使用できます。
- レコードに指定されたときにアクティブになるのは、以下のキーワードのみです。

COLOR

DATE

DFT

DSPATR

DSPSIZ

MSGCON

MSGLOC

TEXT

SLNO (定数値)。SLNO(*VAR) が指定されている場合、開始行番号には 1 が使用されます。

SYSNAME

TIME

USER

その他のディスプレイ・ファイル・キーワードは、使用可能な場合でも、レコードには意味をなさず、ヘルプの表示中に処理される場合もあればそうでない場合もあります。

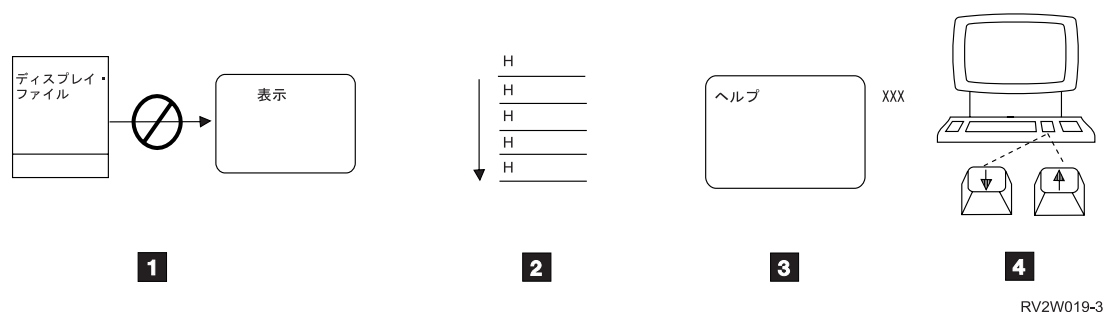
レコードを使用するヘルプ画面間でのページング

レコードを使用してヘルプ情報を定義する場合、ヘルプ・キーを押した際に表示される最初の画面は、**1 次ヘルプ**と呼ばれます。1 次ヘルプ画面で次ページまたは前ページ・キーを押すと、**2 次ヘルプ**が表示されます。

システムがヘルプ画面をページングする方法についての理解

システムは、画面上にあるレコード様式からすべてのアクティブな H 仕様のリストを保守します。この H 仕様のリストは、ヘルプ・リストと呼ばれます。ヘルプ・リストの前にある H 仕様は、ユーザーがヘルプ・キーを押したときに最初に検索されます。

画面上のヘルプ域でヘルプ・キーを押すことによって、以下が行われます。



- 1** ディスプレイ・ファイルを中断します。
- 2** アクティブなヘルプ域のリストを検索して、ヘルプ・キーを押した際にカーソル位置が含まれるリストで、最初のものを見つけます。
- 3** ヘルプ・レコードが含まれるディスプレイ・ファイルを開き、情報を表示します。
- 4** ユーザーがオンライン・ヘルプ情報の中を前後にページ送りできるようにします。

リスト内の H 仕様の順序および内容は、以下のものによって決定されます。

- レコード様式を画面に追加すると、そのレコード様式の H 仕様はリストの前に置かれます。
- レコード様式に複数の H 仕様が含まれる場合、それらは、ディスプレイ・ファイルで定義された順序でヘルプ・リストに追加されます。
- 画面を消去するレコード様式が書き込まれるか、HLPCLR キーワードを使用可能にするレコード様式が書き込まれると、ヘルプ・リストは消去されます。
- レコードで OVERLAY キーワードが有効になっている場合、またはそのレコードに H 仕様が含まれている場合、あるいはレコードがすでに画面上にある別のレコードを完全にあるいは部分的にオーバーラップする場合、ヘルプ・リストはさまざまな仕方で更新されます。以下の表では、3 つの要因の組み合わせごとに、レコードのヘルプ・リスト更新のタイプについて説明します。

オーバーレイ	H 仕様	オーバーラップ	ヘルプ・リストの更新
なし	なし	適用外	ヘルプ・リストが消去されます。
なし	あり	適用外	ヘルプ・リストが消去され、レコードの H 仕様がヘルプ・リストに追加されません。
あり	なし	なし	ヘルプ・リストは変更されません。

オーバーレイ	H 仕様	オーバーラップ	ヘルプ・リストの更新
あり	なし	あり	H 仕様は、書き込まれるレコードの境界内にある場合、ヘルプ・リストから除去されます。
あり	あり	なし	レコードの H 仕様がヘルプ・リストに追加されます。H 仕様は、追加されるレコードの H 仕様の境界内にある場合、ヘルプ・リストから除去されます。
あり	あり	あり	H 仕様は、書き込まれるレコードの境界内にある場合、ヘルプ・リストから除去されます。続いて、レコードの H 仕様がリストに追加されます。最後に、追加されるレコードのいずれかの H 仕様の境界内に H 仕様がある場合、それらはヘルプ・リストから除去されます。

注: ヘルプ域で *NONE が指定された H 仕様は、上で定義されたヘルプ域を持つ最初の H 仕様を除去すると、ヘルプ・リストから除去されます。ヘルプ域で *NONE が指定されたヘルプ仕様が最初のヘルプ仕様である場合、ヘルプ・リストが消去されるか、またはヘルプ域を持つヘルプ仕様がこのヘルプ仕様の上に置かれたときのみ、このヘルプ仕様が除去されます。この H 仕様は、その上にある H 仕様が消去されると、除去されます。

2 次オンライン・ヘルプ情報の表示

2 次ヘルプとして表示されるレコード様式は、同じヘルプ・グループに由来します。または、使用可能なヘルプ・グループが存在しない場合、現在表示されているレコード様式と同じヘルプのサブリストに由来します。

ヘルプ・グループは、HLPSEQ キーワードを使用して定義され、同じグループ名が指定されたそれらのレコード様式で構成されます。HLPSEQ キーワードを使用すると、ヘルプ・グループ名およびヘルプの順序付け番号を指定することができます。ヘルプの順序付け番号は、ヘルプを表示する順序を指定します。別のディスプレイ・ファイルにある 2 つのレコード様式が同じグループ名を持っている場合、それらは依然として別のヘルプ・グループ内にあると見なされます。HLPSEQ キーワードを指定していないレコード様式は、1 つのグループであると見なされます。

ヘルプのサブリストには、ヘルプ境界間で定義されるすべての H 仕様が含まれます。HLPBDY キーワードは、ヘルプ境界を定義することによって、ヘルプ・リストをサブリストに区分します。(HLPBDY キーワードがコーディングされている H 仕様は、境界の前にあると見なされます。) サブリストは、ページ送りキーを使用してさらにオンライン・ヘルプ情報を参照する場合に重要です。

2 次ヘルプの順序を決定する: 次ページ・キーまたは前ページ・キーを使用するかどうかによって、システムは以下のように 2 次ヘルプを選択します。

1. 現在のヘルプ・レコード様式と同じヘルプ・グループ名にあり、2 番目に高いヘルプの順序付け番号 (次ページ・キーを押した場合)、または 2 番目に低いヘルプの順序付け番号 (前ページを押した場合) のレコード様式が選択されます。
2. 現在のヘルプ・レコード様式で、最も高いヘルプの順序付け番号 (次ページ・キーを押した場合) または最も低いヘルプの順序付け番号 (前ページ・キーを押した場合) がすでにグループ内にある場合、現在表示されているレコード様式と同じヘルプ・グループを参照しない次の H 仕様が、ヘルプのサブリストで検索されます。

注:

- a. 予期しないヘルプが組み込まれないようにするため、アプリケーション・ディスプレイ・ファイルの各レコードの最後の H 仕様で HLPBDY キーワードを指定することをお勧めします。これにより、ヘルプが含まれるレコードごとに 1 つのサブリストが定義されます。ただし、複数のレコードが画面上にある場合に、これが望ましくない場合があります。この場合、HLPBDY キーワードは画面に書き込まれた最後のレコードにのみ影響するはずですが。
 - b. この 2 番目の方式が機能するのは、サブリストが識別可能な場合のみであるため、1 次ヘルプがファイルのデフォルトのレコード様式であった場合 (たとえば、ファイル・レベルの HLPRCD キーワードから選択した場合など) には、これは使用されません。
3. サブリストでの検索は、サブリストの境界に達するまで継続されます。さらに、検索はサブリストの他の端に折り返し、表示中の現在の H 仕様に達するまで継続されます。
 4. 満足のいく HLPRCD を持つ H 仕様が見つからない場合、同じヘルプ・グループ名および最も低いヘルプの順序番号 (次ページ・キーを押した場合) または最も高いヘルプの順序番号 (前ページを押した場合) が含まれるヘルプ・ディスプレイ・ファイルのレコード様式が選択されます。現在のヘルプ様式は常にこの基準を満たす (他の様式が満たさない場合) ため、この方式は常に一致を見出します。

以下の例では、2 次情報の順序に影響するものが何もないため、フィールドおよび HLPARA キーワードは指定されません。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
                                HELP
                                HLPRCD(HELP11)
R RECORD1
H                                HLPRCD(HELP1)
H                                HLPRCD(HELPSCR1)
                                HLPBDY
H                                HLPRCD(HELP2SRC)
                                HLPBDY
```

図 121. 2 次ヘルプを示すサンプル DDS ソース

サンプル・ファイル HELPFILE には、以下のように HLPSEQ キーワードがコーディングされているレコード様式が含まれます。

レコード	キーワード仕様
HELP	HLPSEQ(GROUPA 1)
HELP1	HLPSEQ(GROUPA 2)
HELP11	HLPSEQ(GROUPA 3)
HELPSCR1	HLPSEQ(GROUPB 1)
HELPSCR2	HLPSEQ(GROUPB 2)
HELP2SRC	

ヘルプ・レコード様式 HELP と HELPSCR2 はアプリケーション・ディスプレイ・ファイルでは参照されないことに注意してください。これらは、上の方法で参照されないため 1 次ヘルプではありませんが、以下のように 2 次ヘルプとして表示されます。

順序 1:

1 次ヘルプ様式	HELP11
前ページ・キーを押す	HELP1
前ページ・キーを押す	HELP
前ページ・キーを押す	HELP11

この最初の順序では、HELP1 は、GROUPA において直前のヘルプ・レコード様式であるため、前ページ・キーを押した場合に表示されます。同様に、HELP は、前ページ・キーを 2 回目に押した際に表示されます。前ページ・キーを 3 回押すと、ヘルプ・グループの終わりに達します。続いて、HELP11 が再度選択されます。これが元のヘルプ・グループの最後のレコード様式であるためです。

順序 2:

1 次ヘルプ様式	HELPSCR1
次ページ・キーを押す	HELPSCR2
次ページ・キーを押す	HELP1
前ページ・キーを押す	HELP

2 番目の順序は、1 次ヘルプ・レコード様式として表示されている HELPSCR1 から始まります。HELPSCR2 は GROUPB における次のヘルプ・レコード様式であるため、次ページ・キーを押すと HELPSCR2 が表示されます。次ページ・キーを再度押すと、グループの終わりを越えてしまい、HELP1 が検出されます。これは HELP1 が、HELPSCR1 が含まれるサブリスト内で、折り返し後に検索される次のヘルプ・レコード様式であるためです。前ページ・キーを押すと、GROUPA の直前の項目である HELP が表示されることとなります。

順序 3:

1 次ヘルプ様式	HELP2SRC
次ページまたは前ページ・キーを押す	HELP2SRC

HELP2SRC は、ヘルプ・グループ内に存在せず、そのサブリストで唯一の項目であるため、どの方向に画面移動しても表示されます。

2 次ヘルプのレコードに関する制約事項の理解: 以下の制約事項は、アプリケーション・ヘルプのレコード様式を使用する際に適用されます。

- アプリケーション・ヘルプは、ヘルプ・レコードを表示する際のすべての機能キーを制御します。
 - 次ページ・キーは使用可能であり、2 次ヘルプの表示に関する規則に従って、次のレコードを表示します。
 - 前ページ・キーは使用可能であり、2 次ヘルプに関する規則に従って、前のレコードを表示します。
 - Enter キーは使用可能であり、アプリケーション・プログラムに戻るのに使用します。レコードに対して使用可能な CAnn または CFnn キーもまた、アプリケーション・プログラムに戻るのに使用します。他の機能キーは、ヘルプ・キーを含めすべて無視されます。
- USRDFN、SFL、および SFLCTL キーワードを使用したレコードは、レコードとして使用されません。ディスプレイ・ファイルの作成時に、これらのキーワードのいずれかとともにレコード上で HLPSEQ キーワードが見つかり、診断が実行されます。アプリケーションの実行中に、これらのキーワードのいずれかのあるレコードがヘルプとして使用されると、エラー・リセット・メッセージ CPD4050 が発行されます。ヘルプ・レコードは表示されません。

KEEP および ASSUME キーワードは、予測できない結果を生じさせるため、レコード上で使用しないでください。

ヘルプ・キーを押した後に制御をプログラムに戻す

DDS のコーディングの仕方に応じて、ヘルプ・キーを押した後に以下の 2 つの方法のいずれかで制御をプログラムに戻すことができます。

- オンライン・ヘルプ情報を表示してから制御をプログラムに戻すことができます。
- オンライン・ヘルプ情報を表示せずに制御をすぐにプログラムに戻すことができます。

ヘルプ画面を表示した後に制御をプログラムに戻す

ヘルプ・コマンド・キー (HLPCMDKEY) キーワードを使用して、コマンド・アテンション (CAnn) またはコマンド機能 (CFnn) キーが押された後に、アプリケーション・ヘルプ・レコード様式からアプリケーション・プログラムに制御を戻します。CFnn キーはデータをアプリケーション・プログラムに戻しますが、CAnn キーは戻しません。

コマンド・キーは、アプリケーション・レコード様式およびアプリケーション・ヘルプ・レコード様式の両方に適用される必要があります。CAnn または CFnn キーがアプリケーション・ヘルプ・レコード様式に適用されない場合、HLPCMDKEY キーワードは無視されます。

ヘルプ・レコードを使用する以下の DDS ソースでは、HLPCMDKEY キーワードを指定する方法を示します。

```
|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...
      R APPRCD                                CA01
                                              CA03
                                              CF04
                                              HELP
      H                                        HLPKCD(HELPKCD)
                                              HLPARA(1 1 24 80)
                                              8 2'THIS IS THE APPLICATION'
                                              9 2'RECORD FORMAT'
      INPUT1                                10 B 12 10
      INPUT2                                10 B 13 10
      INPUT3                                10 B 14 20
*
      R HELPKCD                                HLPCMDKEY
                                              CA01
                                              CF03
                                              8 2'SPECIFY COMPANY NAME'
                                              9 2'SPECIFY STREET'
                                              10 2'SPECIFY CITY, STATE, ZIP'
```

図 122. HLPCMDKEY を示すサンプル DDS ソース

ユーザーが以前のデータ記述仕様を使用するアプリケーション・ヘルプ画面を開いている場合、各種のキーを押すと以下のことが生じます。

- CF04 キーは、アプリケーション・レコード様式でのみ指定され、Enter キーと同じ働きをします。
- CMD3 キーは、Enter キーと同じ働きをします。制御をプログラムに戻すには、対応する CAnn または CFnn キーをアプリケーション・ヘルプのレコード様式とアプリケーション・レコード様式の両方で指定しなければなりません。
- CA01 キーは、制御をアプリケーション・プログラムに戻します。

次の例では、応答標識が使用されます。応答標識がアプリケーション・レコード様式の CAnn または CFnn キーで指定されている (たとえば、CF12(12) など) 場合、アプリケーション・ヘルプのレコード様式が表示された後に応答標識が戻されます。応答標識がアプリケーション・ヘルプのレコード様式の CAnn または CFnn キーで指定されている (たとえば、CF01(11) など) 場合、応答標識は無視されます。

```

|...+...1...+...2...+...3...+...4...+...5...+...6...+...7...
      R APPRCD                      CA01
                                      CA03
                                      CF12(12)
                                      HELP
      H                              HLPBCD(HLPBCD)
                                      HLPARA(1 1 24 80)
                                      8 2'THIS IS THE APPLICATION'
                                      9 2'RECORD FORMAT'
      INPUT1                          10 B 12 10
      INPUT2                          10 B 13 10
      INPUT3                          10 B 14 20
*
      R HELPRCD                      HLPCMDKEY
                                      CF01(11)
                                      CF12
                                      8 2'SPECIFY COMPANY NAME'
                                      9 2'SPECIFY STREET'
                                      10 2'SPECIFY CITY, STATE, ZIP'

```

図 123. HLPBCDKEY および応答標識を示すサンプル DDS ソース

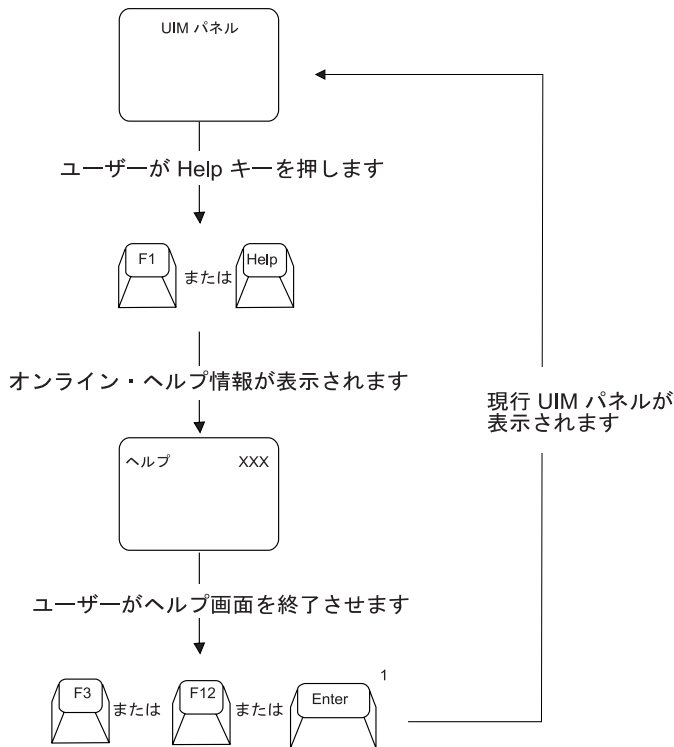
ヘルプ画面を表示せずに制御をプログラムに戻す

HLPRTN キーワードを使用すると、ヘルプ・キーが押されたときに、プログラムにすぐに (ヘルプを表示せずに) 制御を戻します。

第 19 章 パネル・グループのオンライン・ヘルプを利用できるようにする

ほとんどのキーボードのヘルプ・キーは、明確に Help というラベルになっているキーか、またはヘルプ・キーとして機能するよう定義された別の名前のラベルのキーです。そのキーはキーボードによって異なりますが、一般的には F1 キーがヘルプ・キーの代わりとして定義されています。

ユーザーがヘルプ・キーを押すと、以下の図に示されている一連のイベントが起こります。



¹ 実行キーを押したとき、カーソルがハイパーテキスト・リンク上に位置していた場合には、現行 UIM パネルではなく、ハイパーテキスト情報が表示されます。

RV2W061-2

ユーザーがヘルプ・キーを押すと、UIM によって現行のパネルが中断され、ヘルプ・パネルが表示されて、特定の項目、項目のグループ、パネル上の領域、またはパネル全体に関する情報が示されます。表示されたヘルプをユーザーが読み終えたら、ユーザーは F3、F12、または Enter キーを押して、ヘルプ・キーを押す前の現行パネルに戻ります。

定義と説明

パネルに関するヘルプは、パネル定義中で参照されるヘルプ・モジュールで構成されます。読者は UIM のためにパネル・グループ中にパネルのオンライン・ヘルプ情報を定義する必要があり、UIM は HELP ダイアログ・コマンドが要求された際にユーザーにヘルプを表示します。

ユーザーは、HELP ダイアログ・コマンドに割り当てられたキーを押してこのコマンドを要求します。一般的には、Help および F1 のラベルのキーが HELP ダイアログ・コマンドに割り当てられます。この節では、ヘルプ・キーという用語は、HELP ダイアログ・コマンドに割り当てられたいずれかのキーのことを指します。

オンライン・ヘルプ情報とは、キーボード上のヘルプ・キーを押すとユーザーに表示される情報のことです。表示されるヘルプのレベルは、ユーザーがヘルプ・キーを押す際のカーソルの位置に応じて変わります。

パネル全体のヘルプは常に使用できます。このヘルプのことを全般ヘルプと呼び、パネルに関する一般情報になります。全般ヘルプは初めにパネルの目的と使用法に関する情報を示し、次にパネル上の項目ごとの文脈ヘルプを示します。全般ヘルプには、現在画面上に表示されていないものの、Page Up および Page Down キーを使用すると表示できる項目に関する文脈ヘルプが含まれています。(文脈ヘルプには、ユーザーがヘルプを要求した際にカーソル位置にある単一の項目または関連した項目のグループに関する情報が示されます。)パネル上の特定の項目に関する COND 属性を使用すると、そのパネルの項目が現在アクティブでない場合に、その項目が全般ヘルプから除外されます。

文脈ヘルプが定義されていない画面領域に対してヘルプ・キーを押すと、全般ヘルプが表示されます。また、文脈ヘルプの表示中に F2 (拡張ヘルプ) を押しても、全般ヘルプが表示されます。

全般ヘルプは、表示パネル (PANEL) タグの HELP 属性で定義されます。この属性は、パネルに関する全般ヘルプの先頭を含むヘルプ・モジュールを識別します。

パネルに関するオンライン・ヘルプをコーディングする際に、ユーザーがヘルプ・キーを押すと表示される情報が実際に含まれるタグは、ヘルプ・モジュール (HELP) タグです。このタグの属性については詳しくは、579 ページの『HELP (ヘルプ・モジュール)』を参照してください。

パネル内のヘルプをコーディングするには、そのヘルプが指定されているタグに、HELP 属性を指定しなければなりません。ヘルプ情報が含まれているヘルプ・モジュールの名前を、タグの HELP 属性に指定します。

HELP 属性で識別されるヘルプ・モジュールは、HELP 属性があるパネル・グループまたはメニュー中にある場合もあれば、別のパネル・グループ中にある場合もあります。パネル・グループまたはメニューを作成する際に、UIM コンパイラーは、ヘルプ・キーが押された際に UIM がヘルプ・モジュールを検索するパネル・グループ (HELP 属性で指定されている名前と一致するヘルプ・モジュール名を検索する) を、以下のいずれかの場所であると判別します。

1. 作成しているパネル・グループまたはメニューに、HELP タグの NAME 属性中の名前と一致する名前が含まれている場合、ヘルプ・モジュールは、ヘルプ・キーが押されたパネル・グループまたはメニュー中にあります。
2. 作成しているパネル・グループまたはメニューに、インポート (IMPORT) タグの NEWNAME 属性中の名前と一致する名前が含まれている場合、ヘルプ・モジュールは、IMPORT タグの PNLGRP 属性で指定されているパネル・グループ中にあります。
3. 作成しているパネル・グループまたはメニューに、NAME='*' が指定された IMPORT タグが含まれている場合、ヘルプ・モジュールは、IMPORT タグの PNLGRP 属性で指定されているパネル・グループ中にあります。

上記のいずれも当てはまらない場合は、UIM コンパイラーはエラー・メッセージを報告し、パネル・グループまたはメニューは作成されません。

含まれていて、ユーザーのライブラリー・リスト中になければならず、ユーザーにはこのオブジェクトに対する *USE 権限がなければなりません。これらの基準が満たされている場合は、すべての UIM ヘルプ・パネルと検索索引パネルに、機能キー「F9=ユーザー定義メニュー」が表示されます。このパネル・グループはメニュー形式でセットアップされることを意図しています (ユーザーがヘルプ・パネルで F13 を押すと表示される「情報援助」メニューと同様)。パネル・グループの例を以下に示します。

```
:PNLGRP.
:KEYL  NAME=mainkeys help=helpname.
:KEYI  KEY=f1          HELP=helpname ACTION='help'.
F1=Help
:KEYI  KEY=f3          HELP=helpname ACTION='exit' VARUPD=no.
F3=Exit
:KEYI  KEY=f12         HELP=helpname  ACTION='cancel' VARUPD=no.
F12=Cancel
:KEYI  KEY=f24         HELP=helpname  ACTION='morekeys'.
F24=More keys
:KEYI  KEY=enter       HELP=helpname  ACTION='enter'.
:KEYI  KEY=help        HELP=helpname  ACTION='help'.
:KEYI  KEY=print       HELP=helpname  ACTION='print'.
:EKEYL.
:PANEL  NAME=userdef  HELP=helpname  KEYL=mainkeys  TOPSEP=sysnam
        ENTER='MSG CPD9817'.
```

```
User's Info Assist Menu
:MENU  DEPTH='*' BOTSEP=space SCROLL=no.
:TOPINST.To select one of the following, type its number below and
press Enter:
:MENUI  HELP=helpname  OPTION=1 ACTION='cmd dsplibl'.
Display user's library list
:EMENU.
:OPTLINE.Type a menu option below
:EPANEL.
:HELP  name=helpname.Option 1 - Help
```

help text for given help name

```
:EHELP.
:EPNLGRP.
```

「F18=続きの見出し」へのアクセス権を除去する

ユーザーが使用中の検索索引に変更を加えられないようにするには、ユーザーのライブラリー・リスト中に QUHISF18 というデータ域を作成します。ユーザーには、少なくともこのデータ域に対する *USE 権限がなければなりません。このデータ域を作成すると、「検索索引」パネル上の「F18=続きの見出し」機能キーがオフになります。

リスト域のヘルプ

リスト域には UIM リストの内容が表示されます。このリストは、可変情報の行と列で構成され、表形式で表示されます。以下の画面は、リスト域を使用して 8 行のリストを表示するパネルの一例です。

このパネルの場合、ユーザーは「OPT」列に数値 2、4、5、7、または 8 を入力して、リスト内のいずれかの行に対して実行するアクションを指定できます。


```

:listcol var=filenbr help='splf/file_number' usage=out maxwidth=6.Nbr
:listcol var=username help='splf/user_name' usage=out maxwidth=12.User
:listcol var=filepty help='splf/file_priority' usage=out maxwidth=6.Pty
:listgrp col=filecrt help='splf/created_date_time'.Created
:listcol var=filedat usage=out maxwidth=8.Date
:listcol var=filetim1 usage=out maxwidth=8.Time
:elistgrp.
:listview cols='option filenam filenbr username filepty filecrt'.

```

メニュー域のヘルプ

メニュー域には 1 つ以上の項目が含まれており、個々の項目にはオプション番号と実行できるアクションの説明が含まれています。ユーザーは、オプションを選択するには、コマンドかオプション行にオプションの番号を入力し、Enter キーを押します。以下の画面は、メニュー域のあるパネルの一例です。

このパネルには、ユーザーが選択できるオプションが 6 つあり、個々のオプションには実行されるアクションに関する要旨があります。ユーザーが選択したオプション番号を、「選択」行に入力します。

```

Work with Files                               System:  xxxxxxxx

Select one of the following:

  1. Display file attributes
  2. Display file contents
  3. Change ownership
  4. Change authorizations
  5. Delete
  6. Backup to tape

Selection
  -

```

メニュー域のヘルプは、項目レベルで定義しなければなりません。

項目レベルのヘルプ

メニュー域のヘルプを定義するには、メニュー項目 (MENU) タグに HELP 属性を指定します。この属性は、メニュー項目の目的について説明したヘルプ・モジュールを識別します。

カーソルがメニュー項目のテキスト上にある場合、またはカーソルがコマンドかオプション行の位置にあり、有効なメニュー項目番号が入力されている場合に、ヘルプが要求されると、HELP 属性によって識別されるオンライン情報が表示されます。

ヘルプのコーディング

以下のソースは、426 ページのサンプル・パネルのヘルプを定義する方法を示しています。

この例では、参照番号 (**n**) を使用して、タグの HELP 属性を使用したヘルプ・モジュールへの参照と、HELP タグを使用したヘルプ・モジュールの定義との間の関係を示しています。

```

:menu option=1 help='option/display_attr' action='call menuipgm' 1
  .Display file attributes
:menu option=2 help='option/display_cont' action='call menuipgm' 2

```

```

        .Display file contents
:menui option=3 help='option/change_owner' action='call menuipgm' 3
        .Change ownership
:menui option=4 help='option/change_auth' action='call menuipgm' 4
        .Change authorizations
:menui option=5 help='option/delete' action='call menuipgm' 5
        .Delete
:menui option=6 help='option/backup_tape' action='call menuipgm' 6
        .Backup to tape
    :
:help name='option/display_attr' 1
        .Display File Attributes - Help
:xh3.1. Display file attributes
:p.Choose this option to display the attributes associated with this file.
The attributes include all the information about the definition of the file.
:ehelp.
:help name='option/display_contents' 2
        .Display File Contents - Help
:xh3.2. Display file contents
:p.Choose this option to display the data contained in this file.
:ehelp.
:help name='option/change_owner' 3
        .Change Ownership - Help
:xh3.3. Change ownership
:p.Choose this option to change the owner of this file.
You are prompted to enter the name of the new owner.
:ehelp.
:help name='option/change_auth' 4
        .Change Authorizations - Help
:xh3.4. Change authorizations
:p.Choose this option to change the list of users
who have authority to access this file.
You are prompted for the user names and authorizations for the file.
:ehelp.
:help name='option/delete' 5
        .Delete - Help
:xh3.5. Delete
:p.Choose this option to delete this file.
The file and all the data in the file is erased from the system
and the storage used by the file is made available for other use.
:ehelp.
:help name='option/backup_tape' 6
        .Backup to Tape - Help
:xh3.6. Backup to tape
:p.Choose this option to
save a copy of this file on a magnetic tape.
You are prompted for more information about how to back up the
file on the tape.
:ehelp.

```

データ域のヘルプ

データ域には、ユーザーが変更できる可変情報のある 1 つ以上のデータ入力項目が含まれます。また、ユーザーが表示されるが変更できない可変情報のある出力項目もデータ域に含まれます。以下の画面は、データ域のあるパネルの一例です。

```
Sample Entry Panel

Type choices, press Enter:

File name . . . . . _____ Name of document to be printed

Type style for
printing . . . . . 1          1=Prestige Elite (12 pitch)
                              2=Courier (10 pitch)
                              3=Essay Standard (proportional)
                              4=Essay Bold (proportional)
```

データ域のヘルプは、項目ごとに以下の 3 つのレベルのいずれかで定義しなければなりません。

領域レベルのヘルプ

領域レベルでヘルプを定義するには、データ表示域 (DATA) タグに HELP 属性を指定します。この属性は、領域内のすべての項目について説明したヘルプ・モジュールを識別します。

DATA タグに HELP 属性が指定されていない場合は、データ・グループ (DATAGRP)、データ項目 (DATAI)、およびデータ選択フィールド (DATASLT) タグで HELP 属性を指定して、ヘルプ・モジュール名を領域内の個々の項目に関連付けなければなりません。

ユーザーが、DATA タグに LAYOUT=HORIZ 属性を指定したデータ域に関するオンライン情報にアクセスしやすくするには、個々の DATAI タグに HELP 属性を指定する代わりに、DATA タグに HELP 属性を指定します。

グループ・レベルのヘルプ

グループ・レベルでヘルプを定義するには、DATAGRP タグに HELP 属性を指定します。この属性は、グループ中のすべての項目について説明したヘルプ・モジュールを識別します。

DATA タグ、外部 DATAGRP タグ、またはネストされた DATAGRP タグに HELP 属性を指定しない場合は、グループ中のすべての DATAI および DATASLT タグに HELP 属性が必要です。

項目レベルのヘルプ

項目レベルでヘルプを定義するには、DATAI または DATASLT タグに HELP 属性を指定します。この属性は、データ項目または選択フィールドについて説明したヘルプ・モジュールを識別します。

DATA タグまたはデータ項目を含む DATAGRP タグに HELP 属性を指定しない場合は、DATAI および DATASLT タグに HELP 属性が必要です。

ユーザーが LAYOUT=HORIZ データ域に関するオンライン情報にアクセスしやすくするには、水平域の個々の DATAI タグに HELP 属性を使用する代わりに、DATA タグに HELP 属性を使用します。

HELP 属性は、データ項目に関連したすべてのデータ項目エクステンダー (DATAIX) タグに適用されません。

DATASLT タグに HELP 属性を指定した場合は、DATASLTC タグの選択項目ごとにも HELP 属性を指定できます。選択フィールド中の 1 つの DATASLTC タグに HELP 属性を指定した場合は、その選択フィールド中のすべての DATASLTC タグに HELP 属性を指定しなければなりません。

複数項目選択フィールドの場合、カーソルが選択フィールド内のいずれかの位置にある際に表示される文脈ヘルプの一部として、選択項目ごとに識別されるオンライン情報が組み込まれます。

単一項目選択フィールドの場合、カーソルが選択項目のテキスト上にある際にヘルプが要求されると、選択項目ごとに識別されるオンライン情報が表示されます。このオンライン情報は、カーソルが選択フィールド上にあるが、フィールド内のいずれかの選択項目のテキスト上にはない場合に表示される文脈ヘルプの一部としても組み込まれます。このことは、カーソルが選択フィールドのプロンプト・テキスト上にあるときや、選択フィールドの入力フィールド内にあるときにも該当します。カーソルが入力フィールド内にあり、有効な選択項目が入力されている場合は、ヘルプが要求されると、その選択項目に関するヘルプが表示されます。

ヘルプのコーディング

以下のソースは、前述のサンプル・データ域パネルのヘルプを定義する方法を示しています。このサンプル・データ域のヘルプ・モジュールは、すべてパネル・グループ *LIBL/DATAXMP からインポートされます。

```
⋮
:import name='*' pnlgrp='*libl/dataxmp'.
⋮
:datai var=filename help='print/filename' usage=inout.File name
:dataac.Name of document to be printed
:datai var=typestyle help='print/style' usage=inout.Type style for printing
:dataac.1=Prestige Elite (12 pitch)
:dataac.2=Courier (10 pitch)
:dataac.3=Essay Standard (proportional)
:dataac.4=Essay Bold (proportional)
:datai var=leftmarg help='print/left_margin' usage=inout.Left margin
:dataac.Number of spaces from the left edge of the paper (1-20)
:datai var=copies help='print/copies' usage=inout.Copies
:dataac.Number of copies (1-99)
:datai var=duplex help='print/duplex' usage=inout.Duplex
:dataac.1=Yes (Print both sides of paper)
:dataac.2=No (Print one side only)
⋮
```

以下のソースは、パネル・グループ *LIBL/DATAXMP 中のヘルプ・モジュールを定義しています。これらのヘルプ・モジュールは、前述のサンプル・データ域に関するヘルプ・キーを押すとインポートされます。

```
⋮
:help name='print/filename'.File Name - Help
:xh3.File name
:p.Enter the name of document to be printed.
:ehelp.
:help name='print/style'.Type Style for Printing - Help
:xh3.Type style for printing
:p.Enter one of the following to select the type style.
:parml.
:pt.1
:pd.A 12 pitch elite style is used.
:pt.2
:pd.A 10 pitch courier style is used.
:pt.3
:pd.A proportional essay standard style is used.
:pt.4
:pd.A proportional essay bold style is used.
:eparml.
:ehelp.
:help name='print/left_margin'.Left Margin - Help
:xh3.Left margin
:p.Enter a number from 1 to 20 for the number of spaces from
the left edge of the paper to the beginning of the printed text.
:ehelp.
:help name='print/copies'.Copies - Help
:xh3.Copies
:p.Enter a number from 1 to 99 for the number of copies of the
```

```

document to be printed.
:ehelp.
:help name='print/duplex'.Duplex - Help
:xh3.Duplex
:p.Enter one of the following to select whether or not to print on
both sides of the paper.
:parml.
:pt.1
:pd.Print text on both sides of the paper.
:pt.2
:pd.Print text only on one side of the paper.
:eparml.
:ehelp.
:

```

以下のパネルの例は、水平レイアウトでデータ域内のファイル名とライブラリー名を表示します。

```

Work with File Members

File:  FILE01      Library:  QGPL

```

以下のソースは、前述のサンプル・データ域のヘルプを定義する方法を示しています。この例は、領域レベルのヘルプを提供します。カーソルがファイル名を含む行のいずれかの位置にある場合は、ファイル名とライブラリーに関するヘルプが表示されます。

この例では、参照番号 (**n**) を使用して、タグの HELP 属性を使用したヘルプ・モジュールへの参照と、HELP タグを使用したヘルプ・モジュールの定義との間の関係を示しています。

```

:
:
:data depth=2 help='file_and_library' layout=horiz. 1
:datai var=filename usage=out.File
:datai var=library usage=out.Library
:edata.
:
:
:help name='file_and_library'.File and Library - Help 1
:xh3.File and library
:p.The qualified name of the file whose members are being displayed.
:ehelp.
:

```

以下のパネルの例は、2 列の垂直レイアウトで、データ域内の修飾ファイル名とレコード名を表示します。

```

Work with Field Definitions

File . . . . . FILE      Record . . . . . MASTER01
Library . . . . . QGPL

```

以下のソースは、前述のサンプル・データ域のヘルプを定義する方法を示しています。修飾ファイル名に関するグループ・レベルのヘルプも提供されています。カーソルがファイルまたはライブラリーの位置にある場合は、修飾ファイル名に関するヘルプが表示されます。カーソルがレコード名の位置にある場合は、レコード名に関するヘルプが表示されます。

この例では、参照番号 (**n**) を使用して、タグの HELP 属性を使用したヘルプ・モジュールへの参照と、HELP タグを使用したヘルプ・モジュールの定義との間の関係を示しています。

```

:
:
:data depth=3 layout=2.
:datacol width=16.
:datacol width='*'.
:datagrps grpsep=qindent help='library/file' compact. 1
:datai var=filename usage=out.File
:datai var=library usage=out.Library
:edatagrps.
:datai var=record help=record usage=out.Record 2
:edata.
:
:
:help name='library/file'.File and Library - Help 1
:xh3.File and library
:p.The qualified name of the file whose field definitions are displayed.
:ehelp.
:help name=record.Record - Help 2
:xh3.Record
:p.The name of the record whose field definitions are displayed.
:ehelp.

```

メニュー・バー域のヘルプ

メニュー・バー域は、パネルの上部にあり、パネルの使用可能なアクションにユーザーがアクセスできる選択項目が含まれています。ユーザーは、メニュー・バー内の選択項目を選択するには、ご希望の選択項目の位置にカーソルを置いて Enter キーを押します。ユーザーがメニュー・バー内の選択項目を選択すると、メニュー・バーの下にプルダウン・メニューが表示され、パネルに対するアクションに関する選択項目が含まれています。

以下の画面の一部には、メニュー・バー域の例が含まれています。メニュー・バーは、パネルの最上部付近の行で、ユーザーの選択項目として「ファイル」と「ヘルプ」をリストしています。ユーザーはどちらかの選択項目を選択し、その選択項目に関するプルダウン・メニューを表示できます。この例の 2 つ目の画面は、「ファイル」を選択した後に表示されるプルダウン・メニューを示しています。

```

File Help
-----
Work with Programs
System: ROCH0001
Type choices, press Enter.
2=Change 4=Delete 5=Display

Opt Program Library Text
- P P P P P P P P L L L L L L L L L Description textxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
- P P P P P P P P L L L L L L L L L Description textxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
- P P P P P P P P L L L L L L L L L Description textxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

```

「ファイル」を選択すると、パネルに関するアクションの選択項目を含むプルダウン・メニューが表示されます。プルダウン・メニュー内の使用可能なアクションは「変更」、「削除」、「ディスプレイ」、および「終了」です。メニュー・バー内に「ヘルプ」がある場合、ユーザーがこのメニュー・バー選択項目を選択すると、別のアクションのリストを含むプルダウン・メニューが表示されます。

```

File _Help
-----
: 2. Change      :          Work with Programs
:               :
:               :                               System: ROCH0001
: 4. Delete     : er.
: 5. Display    : 5=Display
: 6. Exit       : F3
:.....: y      Text
- P P P P P P P P L L L L L L L L L Description textxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
- P P P P P P P P L L L L L L L L L Description textxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
- P P P P P P P P L L L L L L L L L Description textxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

```

メニュー・バー選択項目に関する文脈ヘルプには、メニュー・バー内の選択項目に関するヘルプや、メニュー・バー選択項目のプルダウン・メニュー内のアクティブな選択項目ごとのヘルプが組み込まれています。メニュー域のヘルプは、メニュー・バー選択項目レベルおよびプルダウン選択項目レベルで定義しなければなりません。

メニュー・バー選択項目レベルのヘルプ

メニュー・バー選択項目レベルでヘルプを定義するには、メニュー・バー選択項目 (M_{BARC}) タグに HELP 属性を指定します。この属性は、メニュー・バー選択項目の目的について説明したヘルプ・モジュールを識別します。

プルダウン選択項目レベルのヘルプ

プルダウン選択項目レベルでヘルプを定義するには、プルダウン・フィールド選択項目 (P_{DFLDC}) タグに HELP 属性を指定します。この属性は、プルダウン選択項目の目的について説明したヘルプ・モジュールを識別します。

ヘルプのコーディング

以下のソースは、432 ページの 1 つ目のメニュー・バー選択項目とそのプルダウン・メニューのヘルプを定義する方法を示しています。

この例では、参照番号 (**n**) を使用して、タグの HELP 属性を使用したヘルプ・モジュールへの参照と、HELP タグを使用したヘルプ・モジュールの定義との間の関係を示しています。

```

:
:
:mbarc help='mbarpgm/file'.File 1
:pdfld.
:pdfldc option=2 help='mbarpgm/file/change' 2
      action='cmd ?CHGPGM PGM(&var2./&var1.)'
      actfor=list.Change
:pdfldc option=4 help='mbarpgm/file/delete' 3
      action='cmd DLTPGM PGM(&var2./&var1.)'
      actfor=list confirm=confpgm usrexit='call exitpgm'.Delete
:pdfldc option=5 help='mbarpgm/file/display' 4
      action='cmd DSPPGM PGM(&var2./&var1.)'
      actfor=list.Display
:pdfldc option=6 help='mbarpgm/file/exit' action='exit set' 5
      varupd=no.Exit
:pdaccel.F3
:epdfld.
:embarc.
:
:
:help name='mbarpgm/file'.File - Help 1
:xh3.File
:p.Select this choice to display a pull-down menu containing options
to perform against selected programs.
:ehelp.
:help name='mbarpgm/file/change'.Change - Help 2
:parml.

```



```

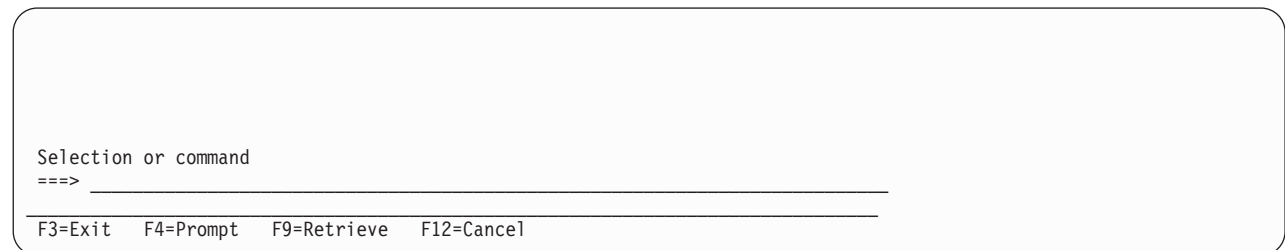
:pt.Change
:pd.Choose this option to change attributes of the selected
programs.
This choice is not available if no programs were selected
from the list.
:eparm1.
:ehelp.
:help name='mbarpgm/file/delete'.Delete - Help 3
:parml.
:pt.Delete
:pd.Choose this option to delete the selected programs.
This choice is not available if no programs were selected
from the list.
:eparm1.
:ehelp.
:help name='mbarpgm/file/display'.Display - Help 4
:parml.
:pt.Display
:pd.Choose this option to display the attributes of the selected
programs.
This choice is not available if no programs were selected
from the list.
:eparm1.
:ehelp.
:help name='mbarpgm/file/exit'.Exit - Help 5
:parml.
:pt.Exit
:pd.Choose this option to end the current task and return
to the display from which the task was started.
:eparm1.
:ehelp.

```

機能キー域のヘルプ

機能キー域は、パネルの下部にあり、ユーザーが機能キーを押すと使用可能になるアクションの説明が含まれています。機能キー域は、ほとんどのパネルで表示されます。

以下の画面の一部には、機能キー F3、F4、F9、および F12 の機能キー域の例が含まれています。これらのキーの説明は、機能キー域にリストされます。機能キー域にはテキスト「F3=終了」、「F4=プロンプト」、「F9=コマンドの複写」、および「F12=取り消し」が含まれます。



機能キー域の文脈ヘルプには、キーのリスト全体に関するヘルプと、アクティブな機能キーごとのヘルプが組み込まれます。パネルにテキストが表示されない機能キーも含まれます。機能キー域のヘルプは、領域レベルで定義でき、項目レベルで定義しなければなりません。

領域レベルのヘルプ

領域レベルでヘルプを定義するには、キー・リスト (KEYL) タグに HELP 属性を指定します。この属性は、機能キー域全体の目的について説明したヘルプ・モジュールを識別します。

項目レベルのヘルプ

項目レベルでヘルプを定義するには、キー項目 (KEYI) タグに HELP 属性を指定します。この属性は、機能キーの目的について説明したヘルプ・モジュールを識別します。

ヘルプのコーディング

以下のソースは、433 ページの機能キー域のヘルプを定義する方法を示しています。ヘルプ・モジュール *fkey* には、ヘルプ情報は含まれていません。このヘルプ・モジュールは、機能キー域の文脈ヘルプのタイトルと、全般ヘルプ中の機能キーの見出しのみ提供します。文脈ヘルプは個々のキーではなく機能キー域全体のヘルプなので、機能キーごとのヘルプにはタイトルや全般ヘルプ見出しのタグはありません。

この例では、参照番号 (**n**) を使用して、タグの HELP 属性を使用したヘルプ・モジュールへの参照と、HELP タグを使用したヘルプ・モジュールの定義との間の関係を示しています。

```
⋮
:keyl name=keys help=fkey. 1
:keyi key=f3 help='fkey/exit' action='exit set' varupd=no.F3=Exit 2
:keyi key=f4 help='fkey/prompt' action=prompt.F4=Prompt 3
:keyi key=f9 help='fkey/retrieve' action=retrieve.F9=Retrieve 4
:keyi key=f12 help='fkey/cancel' action='cancel set' varupd=no.F12=Cancel 5
:keyi key=enter help='fkey/enter' action=enter. 6
:keyi key=help help='fkey/help' action=help. 7
:keyi key=pagedown help='fkey/pagedown' action=pagedown. 8
:keyi key=pageup help='fkey/pageup' action=pageup. 9
:keyi key=print help='fkey/print' action=print. 10
:ekeyl.
⋮
:help name='fkey'.Function Keys - Help 1
:xh3.Function keys
:ehelp.
:help name='fkey/exit'. 2
:parml.
:pt.F3=Exit
:pd.Ends the current task and returns you to the display from
which the task was started.
:eparml.
:ehelp.
:help name='fkey/prompt'. 3
:parml.
:pt.F4=Prompt
:pd.Provides assistance in entering or selecting a command.
:eparml.
:ehelp.
:help name='fkey/retrieve'. 4
:parml.
:pt.:F9=Retrieve
:pd.Shows the last command you entered on the command line,
along with any parameters you included. By pressing this key
once, you receive the last command you ran. By pressing this
key twice, you receive the next to last command that you ran,
and so on.
:eparml.
:ehelp.
:help name='fkey/retrieve'. 5
:parml.
:pt.F12=Cancel
:pd.Returns to the previous menu or display.
:eparml.
:ehelp.
:help name='fkey/enter' 6
:parml.
:pt.Enter
:pd.Submits information on the display for processing.
:eparml.
```

```
:ehelp.  
:help name='fkey/help' 7  
:parml.  
:pt.Help  
:pd.Provides more information about using the display.  
:eparml.  
:ehelp.  
:help name='fkey/pagedown' 8  
:parml.  
:pt.Page Down or Roll Up  
:pd.Moves forward to show additional information for this display.  
:eparml.  
:ehelp.  
:help name='fkey/pageup' 9  
:parml.  
:pt.Page Up or Roll Down  
:pd.Moves backward to show additional information for this display.  
:eparml.  
:ehelp.  
:help name='fkey/print' 10  
:parml.  
:pt.Print  
:pd.Prints the information currently shown on the display.  
:eparml.  
:ehelp.  
:  
:
```

第 20 章 オンライン・ヘルプ情報の定義

ディスプレイ・ファイルの DDS ソースでオンライン・ヘルプ情報を表示できることが指定されている場合に、ヘルプ・キーを押すと、システムはパネル・グループまたは文書によって参照される情報か、または DDS を使用した場合はレコードに含まれる情報を表示します。

この章の以下の節では、ユーザーが目にする実際の情報を作成する方法について説明されています。

- 『パネル・グループ中のオンライン・ヘルプ情報の定義』

UIM を使用してオンライン・ヘルプを作成することについて詳しくは、421 ページの『第 19 章 パネル・グループのオンライン・ヘルプを利用できるようにする』および 299 ページの『第 3 部 パネル・グループを使用したアプリケーション表示のプログラミング』を参照してください。

- 452 ページの『DDS レコードでのオンライン・ヘルプ情報の定義』

ご使用のディスプレイからオンライン・ヘルプ情報にアクセスできることがディスプレイの DDS ソースで指定されていない場合は、『第 18 章 ディスプレイ・ファイルのオンライン・ヘルプ情報をアクセス可能にする』に進んでください。

パネル・グループ中のオンライン・ヘルプ情報の定義

ユーザー・インターフェース管理機能 (UIM) は、パネル・グループを使用してオンライン・ヘルプ情報にアクセスします。パネル・グループ中のオンライン・ヘルプ情報は、以下のいくつかの方法で使用できません。

- ディスプレイ・ヘルプ
- コマンド・ヘルプ
- 索引検索トピック

オンライン・ヘルプ情報用にパネル・グループを使用するには、アプリケーション・ディスプレイ用の DDS ソースにそれを指定しなければなりません。ヘルプ用にパネル・グループを使用する計画で、まだ DDS ソースに指定していない場合は、407 ページの『ディスプレイ・ファイルでヘルプ用にパネル・グループを指定する』を参照してください。

ヘルプ用にパネル・グループの UIM ソースを入力する

ディスプレイ・ファイルが DDS ソースをコンパイルして作成されるのと同様に、パネル・グループは UIM ソースをコンパイルして作成されます。パネル・グループの UIM ソースは、ソース・ファイル・メンバーに入力します。すべてのパネル・グループのソース・タイプは *PNLGRP です。ソース・ファイル・メンバーを作成してデータを入力するステップは、『第 1 章 オンライン・ヘルプ情報のあるサンプル画面の作成』で説明されています。

UIM タグは、常に先頭がコロン (:) で末尾がピリオド (.) で、フォーマットに役立てたり、情報を識別したりするのに使用されます。UIM タグに関する詳細な情報は、『付録 A. UIM パネル・グループ定義言語』で説明されています。

ヘルプ・モジュールを使用したパネル・グループの編成

ディスプレイ・ヘルプ、コマンド・ヘルプ、索引検索トピックのいずれかで使用するにかかわらず、どのパネル・グループの UIM ソースも、先頭は `:PNLGRP` タグで末尾は `:EPNLGRP` タグになります。

ヘルプ情報の単位のことをヘルプ・モジュールと呼び、パネル・グループ中の `:PNLGRP` タグと `:EPNLGRP` タグの間で定義されます。個々のヘルプ・モジュールは、以下のように、先頭が名前指定した `:HELP` タグで末尾が `:EHELP` タグになります。

```
:PNLGRP.  
:HELP name=firsthelp.Title of First Help Module  
:P.  
Information for first help module  
:EHELP.  
:HELP name=secondhelp.Title of Second Help Module  
:P.  
Information for second help module.  
:EHELP.  
:EPNLGRP.
```

1 つのパネル・グループに 1 つ以上のヘルプ・モジュールを含めることができます。ヘルプ・モジュール名は、`:HELP` タグの `name` 属性の値で、ヘルプ・モジュールを識別し、パネル・グループ中のヘルプ・モジュールごとに固有です。`:HELP` タグのピリオド (.) の後のテキストは、オンライン・ヘルプ情報の表示時のタイトルとして使用されます。

ヘルプ・モジュール中の情報を複数回使用する

`:IMHELP` タグを使用すると情報を組み込むことができます。つまり、以下のように、複数回使用される情報を 1 つのヘルプ・モジュール中に定義してから、別のヘルプ・モジュール中で使用できます。

```
:PNLGRP.  
:HELP name=pacific.Pacific Ocean  
:P.  
The Pacific Ocean is the largest ocean in the  
world.  
:IMHELP name=ocean.  
:EHELP.  
:HELP name=atlantic.Atlantic Ocean  
:P.  
The Atlantic Ocean separates North and South America from  
Europe and Africa.  
:IMHELP name=ocean.  
:EHELP.  
:HELP name=ocean.  
:P.  
An ocean is one of the five large bodies of salt water, which  
together cover nearly three-fourths of the world.  
:EHELP.  
:EPNLGRP.
```

注: 上記の UIM ソースで使用される `:P` タグは、段落の先頭を示しています。`:P` タグの詳細については、『付録 A. UIM パネル・グループ定義言語』を利用できます。

ユーザーが上記の例のオンライン・ヘルプ情報を参照する際には、`ocean` の定義が、Pacific Ocean および Atlantic Ocean の両方の定義の 2 番目の文になります。

別のヘルプ・パネル・グループに含まれるヘルプ・モジュールの使用

反復情報を含むヘルプ・モジュールが別のパネル・グループに含まれている場合は、`:IMPORT` タグが必要です。`:IMPORT` タグは、必要なヘルプ・モジュールが含まれるヘルプ・モジュールを識別し、そのパネ

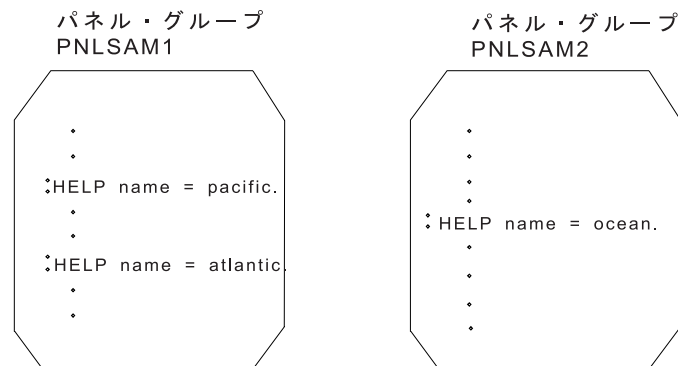
ル・グループを現行のパネル・グループ中で使用できるようにします。 :IMPORT タグは、ヘルプ・モジュールを参照するパネル・グループ中の :PNLGRP タグと最初の :HELP タグの間に入れます。

1 つ目のパネル・グループ (名前 *PNLSAM1*):

```
:PNLGRP.  
:IMPORT pnlgrp=pnlsam2 name=ocean.  
:HELP name=pacific.Pacific Ocean  
:P.  
The Pacific Ocean is the largest ocean in the  
world.  
:IMHELP name=ocean.  
:EHELP.  
:HELP name=atlantic.Atlantic Ocean  
:P.  
The Atlantic Ocean separates North and South America from  
Europe and Africa.  
:IMHELP name=ocean.  
:EHELP.  
:EPNLGRP.
```

2 つ目のパネル・グループ (名前 *PNLSAM2*):

```
:PNLGRP.  
:HELP name=ocean.  
:P.  
An ocean is one of the five large bodies of salt water, which  
together cover nearly three-fourths of the world.  
:EHELP.  
:EPNLGRP.
```



RV2W056-2

ヘルプ・モジュール中のテキストの強調とフォーマット設定

パネル・マークアップ・タグとは、ヘルプ・モジュール中で使用され、情報のフォーマットや強調に役立つ UIM タグのことです。パネル・マークアップ・タグの詳細については、タグの使用法を示す例を含めて、『付録 A. UIM パネル・グループ定義言語』で説明されています。

段落と注記の定義: 以下のパネル・マークアップ・タグは、テキストのブロックを、段落または注記にフォーマットします。

機能	UIM タグ	使用法
注記	:NT および :ENT、:NOTE および :ENOTE	1 つ以上の段落から成る注記。注記は、先頭が Note: で、普通はその前のテキスト中の特定の語句を参照します。注記のテキストは字下げして表示されます。
段落	:P	段落を構成するテキストのブロック。段落のテキストは、ブランク行で他の段落やテキストから分けられます。
段落の連結	:PC	図、例、リストなどの別のパネル・マークアップ・タグによって割り込まれた段落の連結。テキストの段落の連結を指定すると、段落のフォーマット設定が戻ります。

見出しの追加: 以下のパネル・マークアップ・タグは、関連付けられたテキストを見出しとして識別します。

機能	UIM タグ	使用法
拡張ヘルプの見出し	:XH1 から :XH4 まで	情報のメイン・トピックとサブトピックの条件付き見出し。(文脈ヘルプを表示する際には、最初の拡張ヘルプ見出しタグは無視されるので、そのテキストは表示されません。) 見出しは、テキストのセクションを区切ったり編成したりするのに便利です。見出しテキストは、表示される際には強調表示され、独立した行になります。
見出し	:H1 から :H4 まで	情報のメイン・トピックおよびサブトピック。見出しは、テキストのセクションを区切ったり編成したりするのに便利です。見出しテキストは、表示される際には強調表示され、独立した行になります。

テキストの強調表示: 以下のパネル・マークアップ・タグを使用して、テキストを強調表示できます。

機能	UIM タグ	使用法
強調表示句	:HP0 から :HP9 まで、および :EHP0 から :EHP9 まで	強調表示される語または句。テキストを強調表示して、周囲のテキストより目立たせることができます。タグに応じて、テキストを強調表示する際には、通常表示、下線付き表示、輝度や色の変更表示、反転表示、またはこれらの効果を組み合わせた表示が可能です。
タイトルの引用	:CIT および :ECIT	資料のタイトル。資料のタイトルが表示される際には、下線が付けられます。

リストの作成: 以下のパネル・マークアップ・タグを使用して、テキストをリスト形式に編成できます。

機能	UIM タグ	使用法
定義リスト	:DL および :EDL (リストの先頭と末尾)、:DTHD および :DDHD (列見出しの提供)、:DT および :DD (用語とその定義の識別)	語または句のリストと、対応する定義、記述、または説明。定義リストが表示される際には、個々の用語は左側の列に表示され、その定義は向かいの右側の列に表示されます。
リストの一部	:LP	リストの一部に適用されるコメントまたは説明。リストの一部を使用すると、その前のリスト項目のテキスト部分を作成したりリストに割り込んだりせずに、そのリスト項目の後にテキストを入力できます。リストの一部のテキストが表示される際には、字下げされません。

機能	UIM タグ	使用法
番号付き項目リスト	:OL および :LI および :EOL	番号付きの項目のリスト。番号付きリスト内の項目の前には、順序または番号を示す数値か英字が付きます。番号付きリスト内に表示されるリスト項目は字下げされます。
パラメーター・リスト	:PARML および :EPARML	パラメーターの用語と説明。パラメーター・リストは、普通はプログラミング・キーワードやプログラミング変数の定義に使用されます。パラメーター用語は左側の列に表示されます。その定義は用語の次の行に字下げされて表示されます。
単純リスト	:SL および :LI および :ESL	項目のリスト。単純リストは、普通は複数のリスト項目が 1 行に含まれ、順序が必要ない場合に使用されます。単純リスト内のリスト項目は、字下げされますが、前に数値、文字、ハイフン、またはダッシュは付けられません。
順不同リスト	:UL および :LI および :EUL	順不同の項目のリスト。順不同リストは、普通はリスト内の項目に順序が必要ない場合に使用されます。順不同リスト内のリスト項目は字下げされ、前に小文字の o、ハイフン、またはダッシュが付けられます。

プログラミング・キーワードおよびプログラミング変数の識別: 以下のパネル・マークアップ・タグを使用して、プログラミング・キーワードやプログラミング変数を強調表示できます。多くの場合、これらのタグはパラメーター・リスト・タグと併用されます。

機能	UIM タグ	使用法
プログラミング・キーワード	:PK および :EPK	プログラミング・キーワード。プログラミング・キーワードは、プログラミング構文の要素を説明するのに使用します。プログラミング・キーワードは強調表示され、デフォルトとして指定されると、下線も付けられます。
プログラミング変数	:PV および :EPV	プログラミング変数。プログラミング変数は、プログラミング構文の要素を説明するのに使用します。プログラミング変数が表示される際には、下線が付けられます。

構造化テキストの指示: テキストを入力されたとおりに表示するには、以下のパネル・マークアップ・タグを使用できます。

機能	UIM タグ	使用法
例	:XMP および :EXMP	入力したとおり正確に表示されるテキストの領域。例は、サンプルのコンピューター入力や出力を示すのによく使用されます。例のテキストは字下げして表示されます。
図	:FIG および :EFIG	一覧図、グラフ、またはその他の図。希望する場合は、表題を付けて図を表示できます。
不定形式行	:LINES および :ELINES	入力したとおり正確に表示されるテキストの領域。不定形式行は、連結せずに表示する必要のあるテキストの場合に使用されます。

言語タグについて詳しくは、511 ページの『付録 A. UIM パネル・グループ定義言語』を参照してください。

ヘルプ・パネル・グループへの索引検索タグの追加

ヘルプ・パネル・グループには索引検索モジュールが含まれていることもあります。索引検索は、画面ごとに提供されるヘルプ情報を補足します。ヘルプ・パネル・グループ中の索引検索機能に関する情報を使用するには、該当する UIM タグをヘルプ・モジュールに追加する必要があります。

索引検索の働き： ユーザーは、索引検索機能が使用可能であることを指定しているディスプレイ・ヘルプから、その索引検索機能にアクセスできます。

ユーザーが作業画面からヘルプ・キーを押すと、ヘルプ情報が表示されます。索引検索機能をサポートするすべてのヘルプ画面で、「F11=Index search」がアクティブになります。ユーザーが F11 を押すと、「Index Search」画面が表示されます。この画面では、メイン・トピックのリストがトピック階層の形式で表示されます。下部に検索語の入力フィールドがあります。

ユーザーは、リスト内のトピックのブラウズや印刷を行ったり、入力フィールドに 1 つ以上の語を入力したりできます。1 つ以上のトピックを表示するには、ユーザーはトピックのタイトルの隣にあるオプション・フィールドに 5 と入力して、Enter キーを押します。1 つ以上のトピックを印刷するには、ユーザーはオプション・フィールドに 6 と入力します。

トピックの隣の正符号 (+) は、ユーザーがオプション 7 (トピック拡張) を使用して、トピックを展開し、階層中のサブトピックを表示できることを示します。または、ユーザーはオプション 8 (トピック圧縮) を使用して、トピックを圧縮し、サブトピックが表示されなくなるようにすることもできます。

ユーザーが入力フィールドに 1 つ以上の語を入力すると、索引検索機能はその語を同義語表と突き合わせ、入力した検索語と一致するトピックのリストを示します。

以下の図は、ディスプレイ・ヘルプから索引検索機能にアクセスする方法を示しています。

索引検索に挿入するトピックの決定: 索引検索のトピックは、普通は概念情報、手順情報、および参照情報から成ります。

概念情報 は、普通は質問 『Why?』 (なぜ ?) に応答します。概念情報には概要、関係、利点、比較と対比、および図が含まれています。

手順情報 は、質問 『How do I . . . ?』 (どのように ?) に応答します。手順情報は常に一連のステップから成ります。

参照情報 は、普通は質問 『What is . . . ?』 (何 ?) に応答します。参照情報には、定義、例、コマンド、表、前提条件、および注意が含まれます。

索引検索トピックとルート・ワードの定義: :ISCH タグは、索引中のトピックのタイトルを定義し、ユーザーが入力した検索語 (同義語) とトピックの間のリンクの働きをするルート・ワードを指定します。このタグは、このタグが参照する :HELP タグの直後にあります。:ISCH タグは、1 つのヘルプ・モジュール内で 1 つのみ使用できます。

:ルート・ワードの総数が 50 以下の場合、ISCH タグごとにルート・ワードの行が複数あってもかまいません。複数のルート・ワードの行を使用する場合は、2 行目以降の先頭に ROOTS= を繰り返し使用しなければなりません。

```
:PNLGRP.  
:HELP name=entry1.  
:ISCH ROOTS='root1 root2 root3 root4 root5'  
ROOTS='root6 root7 root8 root9 root10'  
ROOTS='root11 root12 root13 ... root50'.  
Title of First Topic  
:P.  
This is the first index search module in this panel group.  
:EHELP.  
:EPNLGRP.
```

すべての行のルート・ワードをアポストロフィで囲まなければならない、ルート・ワードの最後の行の末尾のみにピリオドを使用しなければなりません。:ISCH タグのピリオドの後にトピックのタイトルが続きますが、このタイトルはピリオドの直後の行にすることができます。

ルート・ワードの同義語の指定: :ISCHSYN タグは、ユーザーが入力する語 (同義語) が、特定のルート・ワードと一致することを定義します。ユーザーが入力した語がルート・ワードの同義語である場合は、:ISCH タグにこのルートが含まれるトピックごとに一致が検索されます。

ルート・ワードとして使用されている語が同義語としても使用されるようにするには、その語を :ISCHSYN タグに同義語として組み込まなければなりません。以下に例を示します。

```
:ISCHSYN ROOT='ocean'.ocean water sea
```

:ISCHSYN タグの同義語は 1 行に入力しなければならない、ルート・ワードごとに 1 つ以上の :ISCHSYN タグがなければなりません。複数の行が必要な場合は、同じルート・ワードに関する追加の :ISCHSYN タグを入力できます。

大文字、小文字、または大/小文字混合で入力された同義語の間に UIM は区別を設けません。この理由により、同義語を繰り返し指定して、大/小文字の違いに関するすべてのケースを考慮する必要はありません。

同義語には英字または数字を使用できます。しかし、以下の文字 (16 進数の同等の文字を含む) を同義語またはその一部として使用することはできません。

- ・ (ピリオド)
- ((左括弧)
-) (右括弧)
- ; (セミコロン)
- ,
- ? (疑問符)
- :

:ISCHSYN タグはパネル・グループのどこでも挿入できますが、保守を容易にするには、すべて 1 つの領域に挿入してください (パネル・グループの先頭や、:ISCHSYN タグのみ含まれるパネル・グループ・オブジェクト中など)。

以下の例は、:ISCHSYN タグと、それらのタグを使用する :ISCH タグを示しています。

```
:PNLGRP.
:ISCHSYN ROOT='ocean'.ocean water sea
:ISCHSYN ROOT='lake'.lake water pond
:ISCHSYN ROOT='definition'.definition define description what
:ISCHSYN ROOT='definition'.summary concept information explanation
:HELP name='defocean'.
:ISCH ROOTS='definition ocean'.
Definition of ocean
:P.
An ocean is one of the five large bodies of salt water, which
together cover nearly three-fourths of the world.
:EHELP.
:HELP name='deflake'.
:ISCH ROOTS='definition lake'.
Definition of lake
:P.
A lake is a body of standing water that is enclosed by land.
:EHELP.
:EPNLGRP.
```

索引検索トピックのルート・ワードと同義語の選択: 以下のヒントは、ルート・ワードとして使用する語を決めたり、索引検索トピックの同義語を決めたりするのに役立ちます。

- ・ さまざまなトピックのさまざまな組み合わせをまとめることができるビルディング・ブロックとしてルート・ワードを設計してください。
- ・ トピックのタイトル中の重要な語は、そのトピックの適切なルート・ワードの集合を決めるのに良い出発点になります。
- ・ ある語のすべての変化形の同義語が同じ場合は、その語の 1 つの変化形のみルートとして使用してください。
- ・ 2 つの類似語の同義語が同じ場合は、両方ではなく一方の語のみルート・ワードとして使用してください。
- ・ 共通の同義語が多数ある 2 つの異なるルート・ワードを使用するのが適切な場合があります。
- ・ ルート・ワードとして妥当と思われる同義語をすべて組み込んでください。
- ・ ルート・ワードは同義語とヘルプ・トピックとの接続のみに使用されることに留意してください。
- ・ 1 つのトピックや 2、3 の特定のトピックのみに適用される特殊なルート・ワードを作成する必要があることもありますが、これらの特殊なルート・ワードは規則ではなく例外にする必要があります。
- ・ タイトルに省略語が含まれているトピックの場合、省略語とその省略語を構成する主要な語をすべてルート・ワードとして使用する必要があります。

- 大文字または小文字のいずれかを使用できます。
- コメント行を自由に使用して、特定のルート・ワードが使用されるコンテキストを明確にしてください。
- ルート・ワードが実際の語である場合は、そのルート・ワードの同義語と同じ語を含めてください。
- 一般的につづりを誤りがちな語の、ミススペルを同義語として含めることもできます。

索引検索階層の定義: 索引検索サブトピック (ISCHSUBT) タグは、索引検索 (ISCH) タグに指定された先行するトピックの下にあるサブトピックであるヘルプ・モジュールを、同じパネル・グループ内で識別します。ISCHSUBT タグは ISCH タグの後に出現する必要があります。ISCHSUBT タグで識別されないヘルプ・モジュールは、索引検索階層の主トピックになります。したがって、ISCHSUBT タグを使用しないと、すべてのヘルプ・モジュールが主トピックになり、階層はありません。

ISCHSUBT タグの TOPIC 属性は、トピックのサブトピックを定義するのに使用します。TOPICS 属性上のヘルプ・モジュールの順序が、索引検索階層に表示される順序になります。ヘルプ・モジュール名の規則について詳しくは、515 ページの『名前構文』を参照してください。

1 つのトピックを複数のトピックのサブトピックにすることができます。

トピックを 16 までのレベルにネストできます。

以下の例は、ISCH タグと ISCHSUBT タグを併用して、索引検索階層を形成する方法を示しています。

```
:PNLGRP.
:HELP name=mainhelp.
:ISCH roots='root words'.
Main Help Topic
:ISCHSUBT topics='help1'
           topics='help2'.
:
:EHELP.
:HELP name=help1.
:ISCH roots='root words'.
Help number 1
:ISCHSUBT topics='help3 help4'.
:
:EHELP.
:HELP name=help2.
:ISCH roots='root words'.
Help number 2
:ISCHSUBT topics='help3'
:
:EHELP.
:HELP name=help3.
:ISCH roots='root words'.
Help number 3
:
:EHELP.
:HELP name=help4.
:ISCH roots='root words'.
Help number 4
:
:EHELP.
:EPNLGRP.
```

この UIM ソースは、以下の索引検索階層を作成します。

検索見出し

オプションを入力し、実行キーを押してください（+ は拡張可能トピック）。

5= トピックの表示 6= トピックの印刷 7= トピックの拡大
8= トピックの圧縮

OPT トピック
見出しのタイトル
- 主要ヘルプ・トピック
- ヘルプ番号 1
- ヘルプ番号 3
- ヘルプ番号 4
- ヘルプ番号 2
- ヘルプ番号 3

終わり

または、検索語を入力し実行キーを押してください（* は一致するトピック）。

F3= ヘルプ終了 F5= 全トピック F6= 主要トピック F10= 最初へ移動
F11= 構造の隠蔽 F12= 取消し F13= 情報援助 F18= 見出しの追加

各国語に関する考慮事項: 索引検索機能を 2 バイト文字サポート (DBCS) または 1 バイト文字サポート (SBCS) データで使用できます。DBCS データを使用する際には、要求元の装置がデータを DBCS で入力したり表示したりできるようになっていなければなりません。該当する際には、索引検索データを含むオブジェクトは、DBCS データを含むものとしてマークされます。システムは、装置が DBCS データを処理できるかどうかを判別します。

DBCS 形式のデータを準備しており、索引検索機能をこのデータで使用する場合は、以下の点を考慮してください。

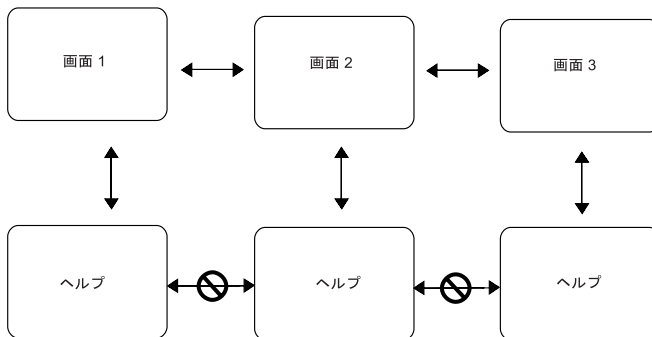
- DBCS システム用の索引検索データを準備する際には、ISCHSYN タグに入力する同義語は 2 バイト文字モードでなければなりません。つまり、タグの後の最初のバイトはシフトアウト文字でなければならず、データの最後のバイトはシフトイン文字でなければなりません。システムは ISCHSYN タグ上のデータを 2 バイトに変換しません。
- 1 バイトのブランクで語を区切らなければなりません。1 個から 19 個までの 2 バイト文字を結合して 1 つの語を形成することができます。間にシフトアウト文字とシフトイン文字があってもかまいませんが、索引検索には無視されます。
- ISCH タグと ISCHSYN タグ (ISCH の ROOTS 属性と ISCHSYN タグの ROOT 属性) のリンクに使用する語は同一でなければならず、DBCS で入力することはできません。
- 検索語は、1 バイト・モードまたは 2 バイト・モードで入力できます。1 バイトのブランクを入力して語を区切ることができます。

1 検索語が画面に表示される際には、2 バイト文字の表記 (実際に検索で使用された文字) が表示されます。
1 索引検索が大/小文字を区別しないように特殊な処理が行われます。PNLGRP タグの TXTCHRID 属性で指定
1 されたコード・ページの変換テーブルを使用して、ISCHSYN タグ上の検索語が大文字にされます。検索語
1 が DBCS の場合は、大文字になりません。構文解析中にシフトアウト文字やシフトイン文字はブランクと
1 して扱われます。前後のブランクは除去されます。装置記述のコード・ページの変換表を使用して、すべて
1 の SBCS 語は大文字にされます。DBCS についての詳細は、iSeries Information Center の 2 バイト文字セ
1 ットのサポートを参照してください。

ヘルプ・モジュールのリンク

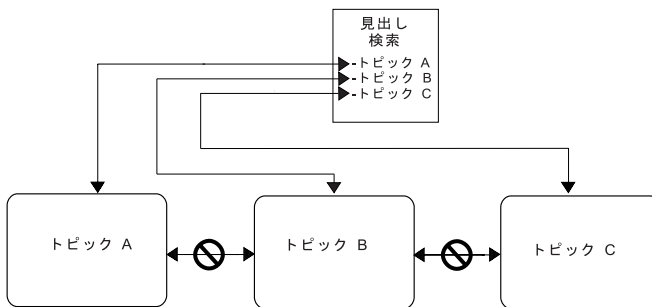
ハイパーテキストとは、共通域で一緒にリンクされている関連したヘルプ・モジュールの構造のことです。ユーザーはリンクを使用して 1 つ以上の興味のあるヘルプ・モジュールを別のヘルプ・モジュールから選択できます。1 つ以上のヘルプ・モジュールにリンクしているヘルプ・モジュールのことを、情報ノードと呼びます。

ハイパーテキストがなければ、ヘルプにアクセスするには関連したディスプレイやコマンド・プロンプターを使用する方法しかなくなります。ハイパーテキスト・ネットワークで個々のヘルプ・モジュールを 1 つのノードにするリンク (2 つの情報ノードの間の関連) が存在しないと、ヘルプ・モジュール間を直接移動することはできません。



RV2W014-2

同様に、索引検索トピックにアクセスするには、索引検索機能を使用する方法しかなくなります。索引検索トピック間を直接移動することはできません。



RV3W131-0

これらの制約事項により、まずどんな関連情報があるか判別すること、次に情報に容易にアクセスすることが難しくなります。ハイパーテキストは、情報ノードの間の関係を識別して、読者や他のユーザーが必要な情報に容易にアクセスできるようにします。

リンクの設計: ハイパーテキスト・ノードの構造により、さまざまなノード間の関係が決まります。

組織図が管理の構造を表すのと同じように、関係している階層の関連 (タスクとそのサブタスクの間の関連やコマンドとそのパラメーターの間の関連など) を、大きなノードとその下の関連する従属ノードとして表すことができます。

関係のない階層の関連 (手順ノードと参照ノードの間の関連や 2 つの類似したプロシージャー間の関連など) は、ノードのクラスターとして表すことができます。

複雑なハイパーテキスト構造を設計しようとしている場合、以下の質問を考慮すると、何を何にリンクするか決定するのに役立ちます。

- 『自分が理解していないことは何か ?』
- 『精通していない語は何か ?』
- 『前提になっている概念情報は何か ?』
- 『自分の作業を完了するのに、知っている必要のあることが他に何かあるか ?』
- 『この作業の後に行う次の作業は何か ?』
- 『この作業に似ている他の作業は何か ?』
- 『この情報のグラフィカル表現の外観はどうか ?』
- 『この一般情報の特定の例は何か ?』
- 『この画面またはコマンドに関連している他の画面またはコマンドは何か ?』

1 つの情報ノードから希望する数だけリンクを作成できます。しかし、個々の情報ノードが別のノードにリンクされていることは必須ではありません。リンクは明確な論理関係を暗示します。関連が明確でなく論理的でもない場合は、リンクは存在しないはずで

リンクの作成: ハイパーテキスト参照句とは、選択可能なハイパーテキスト・リンクを識別する語または句のことです。参照句は (ディスプレイ装置に応じて、高輝度、色、下線、またはこれらの組み合わせによって) 強調されます。ハイパーテキスト参照句は、リンク先に詳細情報があることをユーザーに知らせます。

ハイパーテキスト・リンクを作成するには、:LINK タグを使用して、参照句の先頭にマークを付けます。対応する :ELINK タグを使用して、参照句の末尾にマークを付けます。

ノード A からノード B への片方向のみのリンクを定義します。ノード B からノード A に戻るリンクは :LINK タグでは定義されません。しかし、ユーザーは F12 を押すとノード B からノード A に戻ることができます。またユーザーは、F6 を押して以前に表示されたノードのタイトルのリストを表示してから、リスト内のタイトルの隣にカーソルを置いて Enter キーを押し、選択したノードに再び戻って表示することもできます。

以下の例では、:LINK タグを使用して、ヘルプ・モジュール間のリンクを作成します。

```
:HELP NAME='wrkjob'.Work with Jobs - Help
:XH3.Work with jobs
:P.
The Work with Jobs display shows you the status of your
:LINK PERFORM='DSPHELP job pn1grp1'.
jobs.
:ELINK.
:EHELP.
```

上記の例がユーザーに対して表示される際の画面を以下に示します。

```
Work with Jobs - Help
The Work with Jobs display shows you the status of your jobs.
```

参照句が表示される際には強調されることに注意してください。また、ある属性バイト、1 文字のフィールド、および別の属性バイトの 3 つのブランクが前に挿入されます。ユーザーは、タブ・キーを使用してこのフィールド上にカーソルを置き Enter キーを押すと、強調表示された語または句に関する追加情報を参照できます。

パネル・グループの作成および削除

パネル・グループ作成 (CRTPNLGRP) コマンドは、ヘルプ画面のパネル・グループを作成します。以下の例では、ライブラリー LIBSAM 中のソース・ファイル・メンバー HDMSAM を使用して、同じライブラリー中に PNL SAM という名前のパネル・グループが作成されます。

```
CRTPNLGRP PNLGRP(LIBSAM/PNL SAM) SRCFILE(LIBSAM/SRCSAM) SRCMBR(HDMSAM)
```

パネル・グループ削除 (DLTPNLGRP) コマンドは、システムからパネル・グループを削除します。以下の例では、ライブラリー LIBSAM 中の PNL SAM という名前のパネル・グループが削除されます。

```
DLTPNLGRP PNLGRP(LIBSAM/PNL SAM)
```

パネル・グループをコマンドのヘルプとして割り当てる

プロンプトを出すことができるコマンド・パラメーターのヘルプとしてパネル・グループを作成できます。

最初のヘルプ・モジュールには、コマンドの拡張ヘルプが含まれ、コマンド作成 (CRTCMD) コマンドのヘルプ ID パラメーター用に指定されます。パネル・グループ中の個々の追加ヘルプ項目が、コマンド中のパラメーターに対応します。使用できるヘルプ・モジュールは、プロンプトを出すことができるパラメーターごとに 1 つのみです。

ヘルプ・モジュールをパラメーターと関連付けるには、:HELP 名に以下の規則を使用しなければなりません。

```
:HELP name='help-identifier/parameter-name'.
```

help-identifier はコマンド作成 (CRTCMD) コマンドのヘルプ ID パラメーター用に指定する名前で、parameter-name はヘルプ・モジュールで説明されているパラメーターの名前です。

コマンド・ヘルプのパネル・グループを編成する方法の一例を以下に示します。

```
:PNLGRP.  
:HELP name=startcmd.  
:P.  
The text for this help module is used as the  
extended help for the command.  
:EHELP.  
:HELP name='startcmd/parameter1'.  
:P.  
This help item is used for information about the first  
parameter.  
:EHELP.  
:HELP name='startcmd/parameter2'.  
:P.  
This help item is used for information about the second  
parameter.  
:EHELP.  
:EPNLGRP.
```

検索索引でのパネル・グループの使用法

:ISCH、:ISCHSYN、および :ISCHSUBT タグを含むパネル・グループを検索索引中に項目として追加できます。索引検索機能にアクセスする方法については、442 ページの『索引検索の働き』を参照してください。

検索索引の作成

検索索引作成 (CRTSCHIDX) コマンドは、検索索引オブジェクトを作成します。1 つ以上のパネル・グループ・オブジェクトに含まれるオンライン・ヘルプ情報への参照を提供する検索索引項目をこのオブジェクトに追加できます。これらの参照されるパネル・グループを検索索引に追加したり、検索索引から除去したりできます。

以下の例では、ACCOUNTING という名前の検索索引オブジェクトが現行ライブラリー中に作成されます。

```
CRTSCHIDX SCHIDX(ACCOUNTING)
          TITLE('Accounting Help Index')
          TEXT('Accounting Help Index')
```

検索索引への項目の追加

検索索引項目追加 (ADDSCHIDX) コマンドを使用すると、検索索引オブジェクトで、ISCH タグを含むパネル・グループ中のヘルプ・モジュールごとにルート・ワードと同義語への参照を構築できます。構築後、パネル・グループ中のヘルプ・モジュールは検索索引オブジェクト中の項目になります。

検索索引オブジェクトが複数のパネル・グループを項目として参照する際には、パネル・グループが検索索引オブジェクトに追加される順序により、検索索引機能の表示時に表示される項目の順序が決まります。同様に、パネル・グループ中のヘルプ・モジュールの順序により、検索索引にこれらのトピックが表示される順序が定義されます。パネル・グループで階層を定義すると、階層で定義されている順序でトピックが表示されます。

検索索引オブジェクトに追加できるパネル・グループの限度は 1000 です。検索索引オブジェクトに追加できるパネル・グループのタイプは 1 つのみです。つまり、検索索引オブジェクトに ISCHSUBT タグを使用しているパネル・グループと ISCHSUBT タグを使用していないパネル・グループを含めることはできません。

以下の例では、PAYROLL という名前のパネル・グループが、ACCOUNTING という名前の検索索引に追加されます。ライブラリー・リスト中に、パネル・グループ・オブジェクトと検索索引オブジェクトの両方とも存在しなければなりません。

```
ADDSCHIDX SCHIDX(ACCOUNTING) PNLGRP(PAYROLL)
```

検索索引からの項目の除去

検索索引項目除去 (RMVSCHIDX) コマンドは、パネル・グループへの参照を検索索引オブジェクトから除去します。RMVSCHIDX コマンドは、ADDSCHIDX コマンドを使用して追加したパネル・グループ・オブジェクトへの参照を除去します。

以下の例では、パネル・グループ PAYROLL の項目が検索索引 ACCOUNTING から除去されます。ライブラリー・リストを検索して、検索索引オブジェクトを検出します。

```
RMVSCHIDX SCHIDX(ACCOUNTING) PNLGRP(PAYROLL)
```

検索索引の削除

検索索引削除 (DLTSCHIDX) コマンドは、検索索引オブジェクトをシステムから削除します。DLTSCHIDX コマンドは、検索索引オブジェクトによって参照されるパネル・グループを削除しません。

以下の例では、ACCOUNTING という名前の検索索引オブジェクトを、ライブラリー・リスト中のこのオブジェクトが最初に検出されたライブラリーから削除します。

```
DLTSCHIDX SCHIDX(ACCOUNTING)
```

パネル・グループ中のヘルプを定義する QUSRTOOL 例のコピー

QUSRTOOL ライブラリーは、パネル・グループを使用してオンライン・ヘルプ情報にアクセスする DDS 記述の画面のサンプルを提供します。選択したライブラリーにこれらの例をコピーしてから、独自の使用方法に合うように調整できます。これらのサンプル画面についての詳細は、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

DDS レコードでのオンライン・ヘルプ情報の定義

画面に関するオンライン・ヘルプ情報に、DDS レコードを使用できます。

実際のオンライン・ヘルプ情報を提供するレコード形式を、アプリケーション画面の DDS ソースと同じメンバーに組み込むことができます。別のディスプレイ・ファイルにレコード形式を含めることもできます。オンライン・ヘルプ情報用に使用される DDS ソースの例は、413 ページの『ヘルプ情報が含まれるレコードの入力』に示されています。

第 5 部 IBM i5/OS-Style 画面の指針

第 21 章 IBM i5/OS スタイルの画面の設計	455	拡張処置項目域の操作	481
QUSRTOOL ライブラリーにおける画面例の使用	455	オプションが指定されている場合のリスト操	
サンプル・オブジェクトの認識	455	作	483
サンプル・オブジェクトのインストール	457	カーソル位置の規則	484
サンプル画面、コマンド、およびオンライン・ヘル		エラー条件の規則	485
プ情報の表示	457	1 つの項目のみが選択可能なリスト	486
独自に使用するためのサンプル・オブジェクトの		リストが空の状態でのリスト形式	486
ソースのコピー	461	すべての画面用の機能キー域の定義	486
すべての画面に対する特殊機能および属性の定義	462	任意指定のコマンド行および ID フィールド	487
単一選択メニュー表示の設計	462	使用可能なコマンド行ツール	490
タイトル	463	共通キーの割り当て	490
指示行	463	すべての画面のヘルプ情報の定義	495
メニュー・オプション	463	メニュー画面のヘルプ	495
メニュー選択項目フィールド	464	入力画面のヘルプ	496
機能キー	464	情報画面のヘルプ	497
オンライン・ヘルプ情報	464	リスト画面のヘルプ	498
一般メニュー画面の操作	464	メッセージの定義と表示	499
入力画面の設計	465	共通ユーザー・アクセス (CUA) 基本レベル・モデ	
タイトル	465	ルの設計	501
指示行	465	基本ダイアログ処置	501
プロンプト域	466	機能キー域とメッセージ行の関係	502
フィールド・プロンプト	466	単一項目選択 (メニュー)	502
機能キー	470	選択項目および選択項目入力フィールド	503
オンライン・ヘルプ情報	470	単一選択フィールド操作のガイドライン	503
一般入力画面の操作	470	入力画面	504
情報画面の設計	471	入力フィールド	504
タイトル	472	情報画面	505
位置情報	472	リスト画面	506
プロンプト域 1	473	ヘルプ情報	507
プロンプト域 2	474		
指示行	474		
機能キー	474		
オンライン・ヘルプ情報	474		
一般情報画面の操作	474		
リスト画面の設計	474		
タイトル	475		
プロンプト域	476		
指示行	476		
オプション行	476		
列見出し	477		
拡張処置項目域	478		
リスト・フィールド	478		
ページング位置情報	478		
機能キー	479		
オンライン・ヘルプ情報	479		
一般リスト画面の操作	479		
リスト制御フィールドの操作	480		
リストの位置指定	480		
リストでの小文字名の位置指定	481		
リスト制御フィールドの変更およびリストの			
位置指定	481		

第 21 章 IBM i5/OS スタイルの画面の設計

データ記述仕様 (DDS) を使用してディスプレイをインプリメントする場合、この章では、それぞれの i5/OS システム表示タイプごとのフォーマットを示し、すべてのキー表示エレメント (定数およびデータ・フィールドの両方) の位置および特性について説明します。さらに、この章では、システム間で機能キーに割り当てられた共通のアクションについて説明します。ただし、ユーザー・インターフェース管理機能 (UIM) を使用してディスプレイをインプリメントする場合、UIM は正しいフォーマットおよび配置を提供します。UIM について詳しくは、299 ページの『第 3 部 パネル・グループを使用したアプリケーション表示のプログラミング』を参照してください。

この章のガイドラインを使用することによって、ご使用の画面は、システム・アプリケーション体系[®] (SAA[®]) 環境と互換性を持つようになります。これは、ディスプレイ装置に適用される IBM システム・アプリケーション体系の共通ユーザー・アクセス (CUA) 規則およびガイドラインを有効に活用します。

QUSRTOOL ライブラリーにおける画面例の使用

QUSRTOOL ライブラリー (基本オペレーティング・システムにオプションでインストール可能) の目的は、システムのアプリケーション開発および管理に役立つ、さまざまなツールおよびプログラミング手法の例にアクセスできるようにすることです。

これらのプログラムの例のいずれかを使用する前に、パッケージされている保管ファイルをまずソース物理ファイルに変更しなければなりません。これはすでに行われている可能性があります。そうでない場合にこれを行うために UNPACKAGE ツールが提供されています。ファイルのアンパックに関する指示は、ソース物理ファイル QATTINFO のメンバー TTTINFO を参照してください。(DSPPFM コマンドを使用して、物理ファイル・メンバーを表示できます。)

ソース物理ファイル QATTINFO には、開始するのに必要な情報がすべて含まれています。QUSRTOOL ライブラリーにあるプログラム例ごとに、ツールおよびそのインストール方法について説明するメンバーが QATTINFO 内にあります。

QUSRTOOL ライブラリーにある画面の例は、4 つのサンプル画面およびサンプル・コマンドへのアクセスを提供します。オンライン情報は、画面およびコマンドで使用可能であり、これには索引検索機能およびハイパーテキスト・リンクが含まれます。

サンプル・オブジェクトの認識

QUSRTOOL にある表示ツールのサンプルは、例で使用されるオブジェクトのインストール、作成、および削除に役立ついくつかのメンバーで構成されます。

表 40. QUSRTOOL の画面例のソース・メンバー (インストール、作成、および削除)

ソース・メンバー	ソース・ファイル	オブジェクト・タイプ	説明
T0011INF	QATTINFO	適用外	選択したライブラリー内のサンプル・オブジェクトをインストールするのに必要な事柄を伝える基本文書メンバー。文書は、DSPPFM コマンド、プリンターへの CPYF、または SEU 表示機能を使用して表示できます。

表 40. QUSRTOOL の画面例のソース・メンバー (インストール、作成、および削除) (続き)

ソース・メンバー	ソース・ファイル	オブジェクト・タイプ	説明
T0011CRT	QATTCL	*PGM	すべてのサンプル #1 オブジェクトを作成するためのインストール・プログラム
T0011CR2	QATTCL	*PGM	すべてのサンプル #2 オブジェクトを作成するためのインストール・プログラム
T0011CR5	QATTCL	*PGM	すべてのサンプル #3 オブジェクトを作成するためのインストール・プログラム
T0011DCL	QATTCL	*PGM	4 つのサンプル画面、コマンド、およびオンライン・ヘルプ情報とともに、サンプル・ディスプレイ・ファイルを示すプログラム
T0011DLT	QATTCMD	*CMD	サンプル #1 のすべてのサンプル・オブジェクトを削除するコマンド
T0011DC2	QATTCMD	*CMD	サンプル #2 のすべてのサンプル・オブジェクトを削除するコマンド
T0011DC5	QATTCMD	*CMD	サンプル #3 のすべてのサンプル・オブジェクトを削除するコマンド
T0011DEL	QATTCL	*PGM	サンプル #1 のサンプル・オブジェクトを削除する T0011DLT コマンドによって使用されるプログラム
T0011DL2	QATTCL	*PGM	サンプル #2 のサンプル・オブジェクトを削除する T0011DC2 コマンドによって使用されるプログラム
T0011DL5	QATTCL	*PGM	サンプル #3 のサンプル・オブジェクトを削除する T0011DC5 コマンドによって使用されるプログラム

サンプル画面、コマンド、およびオンライン・ヘルプ情報は、QUSRTOOL の以下のメンバーに含まれています。

表 41. QUSRTOOL の画面例のソース・メンバー (サンプル画面、コマンド、およびオンライン・ヘルプ情報)

ソース・メンバー	ソース・ファイル	オブジェクト・タイプ	説明
T0011CMD T0011CM2 T0011CM3 T0011CM5	QATTCMD	*CMD	サンプル・コマンド
T0011CLP T0011CP2 T0011CP3 T0011CP5	QATTCL	*PGM	サンプル・コマンドによって使用されるコマンド処理プログラム (CPP)
T0011CHL	QATTUIM	*PNLGRP	サンプル・コマンドのオンライン・ヘルプ情報
TOO11DDS	QATTDDS	*FILE	サンプルの i5/OS スタイル・メニュー、入力画面、情報画面、およびリスト画面用の DDS ソースが含まれるサンプル・ディスプレイ・ファイル
T0011DHL	QATTUIM	*PNLGRP	サンプル #1 のサンプル・ディスプレイ・ファイルのオンライン・ヘルプ情報
T0011IDX	QATTUIM	*PNLGRP	索引検索オブジェクトによって参照されるオンライン・ヘルプ情報
T0011DD5	QATTDD5	*FILE	サンプル #3 の物理ファイルのソース。

表 41. QUSRTOOL の画面例のソース・メンバー (サンプル画面、コマンド、およびオンライン・ヘルプ情報) (続き)

ソース・メンバー	ソース・ファイル	オブジェクト・タイプ	説明
T0011HL2	QATTUIM	*PNLGRP	サンプル #2 のオンライン・ヘルプ。
T0011MN2	QATTUIM	*MENU	サンプル #2 のメニュー。
T0011PNI	QATTUIM	*PNLGRP	サンプル #3 のパネル・グループ (さまざまな宣言を含む)。
T0011PN2	QATTUIM	*PNLGRP	サンプル #2 のパネル・グループ。
T0011PN5	QATTUIM	*PNLGRP	サンプル #3 のパネル・グループ。
T0011PN6	QATTUIM	*PNLGRP	サンプル #3 のオンライン・ヘルプ。
T0011RA5	QATTRPG	*PGM	サンプル #3 のリスト・オプションの処理プログラム。
T0011RE5	QATTRPG	*PGM	サンプル #3 の一般パネル出口プログラム。
T0011RF5	QATTRPG	*PGM	サンプル #3 の F4 を処理するプログラム。
T0011RF6	QATTRPG	*PGM	サンプル #3 の機能キーの処理プログラム。
T0011RI6	QATTRPG	*PGM	サンプル #3 のプログラムによって使用される RPG PLIST。
T0011RI7	QATTRPG	*PGM	サンプル #3 のプログラムによって使用される VARRCD 定義。
T0011RL5	QATTRPG	*PGM	サンプル #3 の不完全リスト処理の出口プログラム。
T0011RP5	QATTRPG	*PGM	サンプル #3 のメインプログラム。
T0011RP6	QATTRPG	*PGM	サンプル #3 の F4 プロンプトの処理プログラム。

他のオブジェクトは QUSRTOOL ライブラリーに含まれていませんが、サンプル・オブジェクトをライブラリーに作成する際に作成されます。

表 42. サンプル・オブジェクトを作成する際に作成されるオブジェクト

ソース・メンバー	オブジェクト・タイプ	説明
T0011IDX	*SCHIDX	検索索引オブジェクト
T0011MSGFL	*MSGF	サンプル・コマンドおよびディスプレイ・ファイルの作成時に使用されるメッセージ・ファイル

サンプル・オブジェクトのインストール

画面例をインストールするには、T0011INF にあるインストール指示に従ってください。インストール・プログラムは、サンプル・オブジェクトをユーザーが選択したライブラリーに置きます。

サンプル画面、コマンド、およびオンライン・ヘルプ情報の表示

サンプル画面、コマンド、およびオンライン・ヘルプ情報は、サンプル・オブジェクトをインストールした後で表示できます。サンプルを表示するには、以下のコマンドを入力します。

- サンプル・メニューを表示するには、次のように入力します。

```
CALL PGM(your-library/T0011DCL) PARM(MENU)
```

次のサンプル・メニューが表示されます。

```

                                Go To Another List

Select one of the following:

    1. Work with documents in folder
    2. Work with documents to be printed
    3. Work with folders
    4. Work with nontext document data
    5. Work with text profiles

Selection
    -
F3=Exit   F12=Cancel
```

図 125. QUSRTOOL のサンプル・メニュー

画面のオンライン・ヘルプ情報を表示するにはヘルプ・キーを押します。表示されるオンライン・ヘルプ情報は、ヘルプ・キーを押した時にカーソルのある位置によって異なります。

- サンプル入力画面を表示するには、次のように入力します。

```
CALL PGM(your-library/T0011DCL) PARM(ENTRY)
```

次のサンプル入力画面が表示されます。

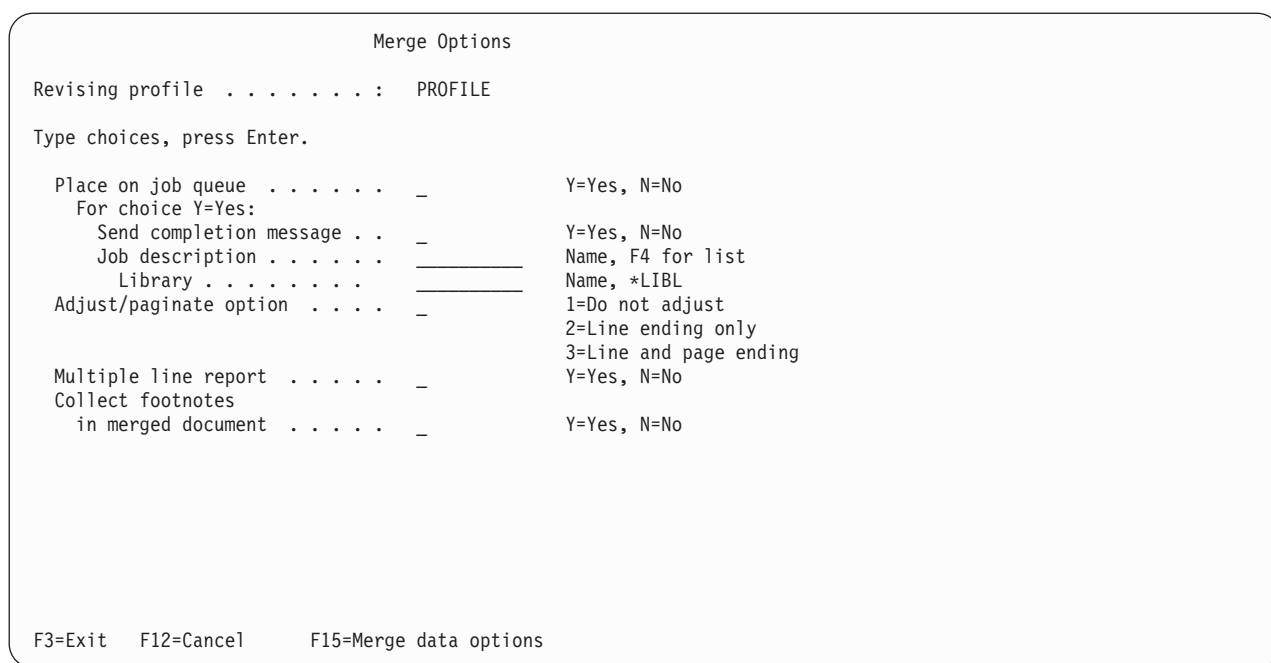


図 126. QUSRTOOL のサンプル入力画面

画面のオンライン・ヘルプ情報を表示するにはヘルプ・キーを押します。表示されるオンライン・ヘルプ情報は、ヘルプ・キーを押した時にカーソルのある位置によって異なります。

- サンプル情報画面を表示するには、次のように入力します。

```
CALL PGM(your-library/T0011DCL) PARM(INFO)
```

次のサンプル情報画面が表示されます。

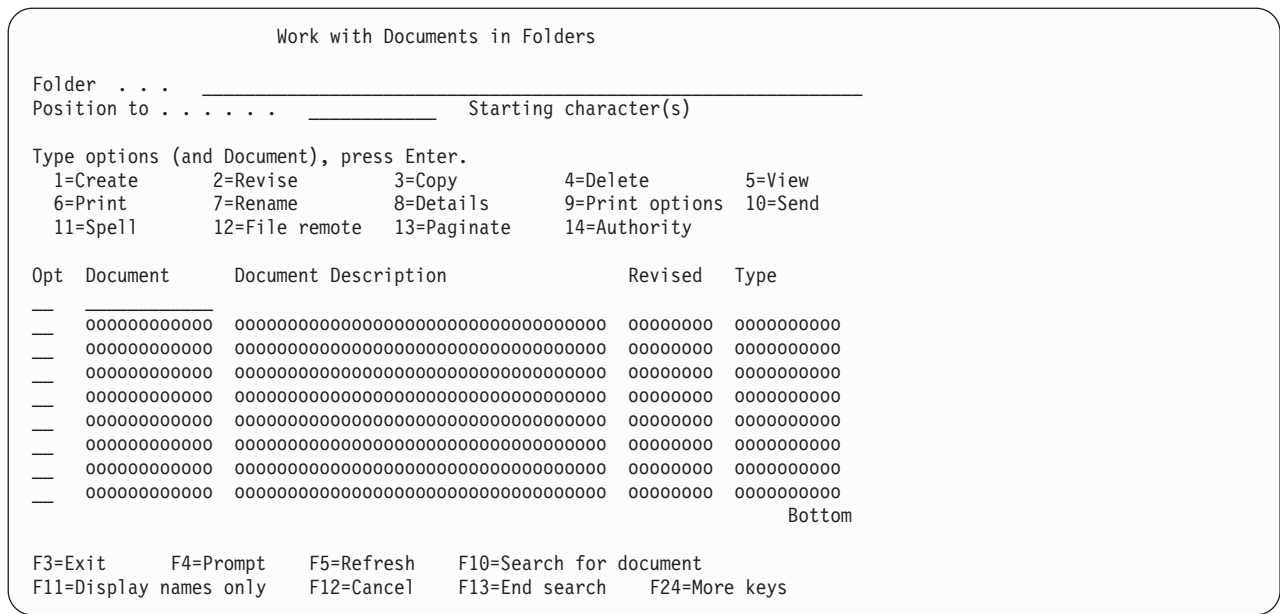


図 128. QUSRTOOL のサンプル・リスト画面

画面のオンライン・ヘルプ情報を表示するにはヘルプ・キーを押します。表示されるオンライン・ヘルプ情報は、ヘルプ・キーを押した時にカーソルのある位置によって異なります。

- 索引検索機能を表示するには、以下のようにしてください。
 1. コマンドを入力して、458 ページの図 125、459 ページの図 126、460 ページの図 127、または図 128 で示されている画面のいずれかを表示します。
 2. ヘルプ・キーを押して画面のヘルプを表示します。
 3. F11 を押して索引検索機能を使用します。
 4. F5 を押して索引検索で使用可能なトピックをすべて表示します。
 5. リストに表示されているすべてのトピックの表示を選択できます。リストの 2 番目のトピックにはハイパーテキスト・リンクが含まれます。リストの 4 番目のトピックは、コマンドのヘルプと同じパネル・グループを使用します。
- サンプル・コマンドを表示するには、T0011CMD を入力します。

コマンドのオンライン・ヘルプ情報を表示するにはヘルプ・キーを押します。表示されるオンライン・ヘルプ情報は、ヘルプ・キーを押した時にカーソルのある位置によって異なります。

独自に使用するためのサンプル・オブジェクトのソースのコピー

QUSRTOOL ソース・ファイルからソースをコピーした後で、独自に使用するためにサンプル・オブジェクトを調整することができます。QUSRTOOL ソース・ファイルからソースをコピーする方法には、基本的に 2 つの方法があります。

- SEU の走査検索/コピー・サービス機能を使用して一度に 1 つのメンバーをコピーできます。
- ソース・ファイルのコピー (CPYSRCF) コマンドを使用して、QUSRTOOL ソース・ファイルから一度にすべてのメンバー (たとえば、QATTCL に含まれるすべての CL ソース・メンバーなど) をコピーできます。

すべての画面に対する特殊機能および属性の定義

以下の機能および属性は、すべての i5/OS スタイルの画面において必須です。

表 43. すべての i5/OS スタイルの画面に必要な機能および属性

機能	説明
オンライン・ヘルプ情報	ヘルプ・キーおよび代替ヘルプ・キーとしてコマンド・アテンション (CAnn) キーを使用してすべての画面から使用可能です。代替ヘルプ・キーのデフォルトは CA01 です。これは、ヘルプ機能を F1 (CUA 指定のヘルプ・キー) に割り当てます。
カーソル明滅	レコードが表示画面に表示されている間、明滅します。
コマンド機能キー CF03 および CF12	対応する機能キー (F3 および F12) が画面でアクティブになるようにします。
ヘルプの消去	現行のヘルプのみが使用可能になるようにします。

以下の機能および属性は、すべての i5/OS スタイルの画面においてオプションです。

表 44. すべての i5/OS スタイルの画面に任意指定の機能および属性

機能	説明
ページ印刷キー	ユーザーが画面を印刷できるようにします。
前ページおよび次ページ (ロールアップまたはロールダウン) キー	ユーザーがメッセージ・サブファイルをページングできるようにします。
代替前ページおよび代替次ページ・キー (代替ロールアップおよび代替ロールダウン) キー	コマンド機能 (CFnn) キーを代替ページング・キーとして割り当てます (デフォルト・キーはそれぞれ CF07 および CF08 です)。
キーボード・ロック	システムが画面の書き込みを完了して入力の準備ができるまで、キーボードがアンロックしないようにします。

単一選択メニュー表示の設計

i5/OS システム・メニューは、ユーザーが 1 つの項目を選択できる選択項目のリストを表示します。これには常に、タイトル、指示、選択項目のリスト、および選択される項目の数を入力するためのラベル付きフィールドが含まれます。463 ページの図 129 は、メニューの例を示しています。



図 129. アプリケーション・メニューのサンプル

この画面の設計を続ける前に、462 ページの『すべての画面に対する特殊機能および属性の定義』にある必要な機能および属性を (必要に応じて、オプション機能および属性も) 定義します。

サンプル画面の表示または独自に使用するためのソースのコピーについて詳しくは、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

タイトル

タイトルは、1 行目の中央に大/小文字混合で、そして高輝度で表示すべきです。

指示行

指示行のガイドラインは以下のとおりです。

- 指示行は 3 行目にあり、2 桁目で始まります。これは、画面上にリストされるオプションの中から選択するようユーザーに通知します。
- メニューの指示行の書式は次のとおりです。

次の 1 つを選択してください:

指示行はコロンで終わることに注意してください。

- システム画面のすべての指示行は、カラー・ディスプレイでは青で指定されます。
- 4 行目はブランクにし、メニュー・オプションを 5 行目で始めます。

メニュー・オプション

メニュー・オプションのガイドラインは以下のとおりです。

- 指示行の後に 1 つのブランク行を続けてオプションを開始します。
- オプションをリストします (1 行に 1 つ)。オプション 1 から 9 は 7 桁目で、オプション 10 以上は 6 桁目で開始します。オプション番号の後にピリオドが続きますが、強調表示されません。
- 可能な場合は常に、オプションの数を最大 10 に制限します。

- オプションはシングル・スペースで、数字順に表示されます。数字順のギャップは、番号付けオプションの欠落数にかかわらず、1つのブランク行で示されます。
- オプション・テキストの最初の語の最初の文字は大文字にします。残りの語は文の必要に応じて大文字にしてください。
- オプション・テキストの各行の終わりに句読点はありません。
- メニューでサインオフがオプションになっている場合、そのオプション番号は 90 でなければなりません。
- 既存のオプション番号は、新規オプションを追加する際に変更してはなりません。

メニュー選択項目フィールド

メニュー選択項目フィールドのガイドラインは以下のとおりです。

- メニュー選択項目フィールドの長さは、オプション番号 1 から 9 の場合は 1 桁、オプション番号 10 以上 (最大 99 まで) の場合は 2 桁になります。
- メニュー選択項目フィールドは、機能キー域に関連した固定位置にあります。これは、機能キー域の先頭の (またはただ 1 つの) 行の 2 行上の 7 桁目で始まります。
- 選択プロンプトは、メニュー選択項目フィールドの上の行にあります。
- 選択プロンプトのテキストは次のとおりです。

```
Selection
```

- 機能キー域の先頭行のすぐ上の行はブランクです。

機能キー

機能キー域の定義に関するガイドラインは、486 ページの『すべての画面用の機能キー域の定義』に記載されています。

オンライン・ヘルプ情報

ユーザー・インターフェース管理機能 (UIM) を使用した、このサンプル画面のオンライン・ヘルプ情報は QUSRTOOL で入手できます。QUSRTOOL 内のソースの使用法については、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

オンライン・ヘルプ情報の定義に関するガイドラインは、495 ページの『メニュー画面のヘルプ』に記載されています。

一般メニュー画面の操作

以下は、メニューで項目を指定してもしなくても、ユーザーが Enter キーを押した際にどうなるかを要約したものです。

- 選択項目フィールドに入力がない場合でも、メニューは画面上に残り (ノーオペレーション命令)、キーボードがロックされていなければ、情報メッセージが表示されます。以下に例を示します。

```
Enter option number or command. Press Help for details.
```

- 入力された値が無効なメニュー・オプションであった場合、

```
Value entered is not a valid menu selection
```

(入力された値は有効なメニュー選択ではありません)

といったメッセージを出すべきです。

- 有効なオプションであると判断された項目がある場合、要求されたアクションが処理されます。

入力画面の設計

入力画面を使用して、ユーザーは入力フィールドに入力することができます。これは通常、アクション要求に関連したシステム・パラメーターおよびオプションを提供するのに使用されます。図 130 は入力画面の例を示しています。

```

Merge Options

Type choices, press Enter.

Place on job queue . . . . . -          Y=Yes, N=No
Send completion message . . . . . -     Y=Yes, N=No
Job description . . . . . _____   Name, F4 for list
Library . . . . . _____           Name, *LIBL
Adjust/paginate option . . . . . -      1=Do not adjust
                                           2=Line ending only
                                           3=Line and page ending

Multiple line report . . . . . -        Y=Yes, N=No
Collect footnotes
in merged document . . . . . -         Y=Yes, N=No

F3=Exit  F4=Prompt  F12=Cancel

```

図 130. サンプル入力画面

この画面の設計を続ける前に、462 ページの『すべての画面に対する特殊機能および属性の定義』にある必要な機能および属性を (必要に応じて、オプション機能および属性も) 定義します。

サンプル画面の表示または独自に使用するためのソースのコピーについては、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

タイトル

タイトルは、1 行目の中央に大/小文字混合で、そして高輝度で表示すべきです。

指示行

指示行のガイドラインは以下のとおりです。

- 各指示行の最初の文字は 2 桁目で始まります。
- 入力画面の指示行は、大/小文字混合で必須フィールドです。書式は、入力される内容に応じて以下のいずれかになります。

選択項目を入力し、実行キーを押してください。
 変更内容を入力し、実行キーを押してください。
 情報を入力し、実行キーを押してください。

- 入力画面の最上部指示行はピリオドで終わります。
- システム画面のすべての指示行は、カラー・ディスプレイでは青で指定されます。

- 単一の指示 (1 つの文) が推奨されていますが、2 つの指示が許可されています。各指示は、2 桁目で始まる各行の最初の文字とともに 1 行に収まらなければなりません。指示の間に空白行はありません。たとえば、次のとおりです。

選択項目を入力して、実行キーを押してください。
F16 を押して、送信してください。

- 1 つの指示の長さは最大 2 行とすることができます。折り返し行は 2 つのスペースで字下げされます (4 桁目で始まる)。

配布リストの.....
入力し、実行キーを押してください。

プロンプト域

一般に、フィールド・プロンプトおよび入力フィールドが入る欄は画面の左半分にあるようにし、選択可能な項目のリストが入る欄は画面の右半分にあるようにします。要求に対するすべてのプロンプトを単一画面のプロンプト域に表示できない場合、プロンプト域のページングを使用可能にしてください。

フィールド・プロンプト

フィールド・プロンプトのガイドラインは以下のとおりです。

- フィールド・プロンプトは、通常の文では大文字化され、プロンプトが識別するフィールドの左側にあります。
- プロンプトの最初の語の最初の文字は大文字にし、残りのテキストは、それが文法的に正しくない場合を除いて小文字にします。
- プロンプトの終わりに句読点はありません。
- 一連のピリオド (ドット) は、フィールド・プロンプトおよび入力フィールドに接続するのに使用されません。
- ドットは 1 文字おきにスペースが空けられます。最右端のドットは、フィールド・プロンプトに指定された欄の幅の最後の桁にあり、ドットはプロンプト・テキストの後ろに 1 桁おきに置かれます。プロンプトが識別する最右端のドットと入力フィールドの間には、空白・スペースが 3 つ必要です。
- プロンプト・テキストに最も近接しているドットは、プロンプトの 2 または 3 桁後ろにあります (プロンプト・テキストとドットの間には最小 1 つの空白がある)。
- 入力フィールドの場合、2 ドット以上の余裕がなければなりません。そうでない場合、ドットは表示されないはずですが。

プロンプト . . . _____ (少なくともドット 2 個分のスペースがあるとき)
長いプロンプト _____ (ドット 2 個分のスペースがないとき)

- フィールド・プロンプトおよび対応する入力フィールドの字下げにより、階層のレベルを表示することができます。
- フィールド・プロンプトの欄は、入力フィールドが含まれる欄から 3 つの文字位置 (空白) で区切られます。
- フィールド・プロンプトの複数の行が表示されている場合、プロンプト、ドット、および入力フィールドの開始桁は位置合わせされます。
- 入力画面のフィールド・プロンプトの前に、2 桁目で始まる指示行があります。フィールド・プロンプトは 4 桁目で始まります。

以下に例を示します。

変更内容を入力し、実行キーを押してください。
得意先名 _____
得意先住所 _____
電話番号 _____

指示行は、最初のフィールド・プロンプトから 1 つのブランク行で区切られます。

- フィールド・プロンプトのテキストは、2 行目に折り返される場合、追加の 2 つのスペースで字下げされます。入力フィールドは最後の行の後にあります。

ファイルの
レコード様式 _____

- 複数行に折り返される不定形式 (継続) の入力フィールドでの先行ドットの使用は、標準形式と同じです。ただし、フィールドが別の行に折り返される場合には、それを 80 桁まで拡張し、以下の行の 1 桁に折り返す必要があります。

原因 _____

- 各行ベースで形式設定された入力フィールドは、プロンプトに続く行で始まる個別の行を占有しているべきです。入力フィールドは、プロンプトの先頭から 2 桁字下げしてください。

プロンプトの後に先行ドットは置かれませんが、識別されるフィールドが同じ行にある場合を除き、先行ドットを決して使用しないでください。

メモ

- 入力される単一のデータ断片に複数パーツが含まれる場合、ユーザーによって認識され、単一のデータ断片と見なされるなら、そのパーツを 1 つの行に置くことができます。

社会保険番号 _ _ _
行/桁 _ _ _

フィールド・プロンプトの開始/終了日付は、以下のように 1 つの行に 2 つのフィールドとして形式設定すべきです。

申告期日 開始 _/_/_ 終了 _/_/_ MM/DD/YY
文書日付 開始 _/_/_ 終了 _/_/_ MM/DD/YY

- 値のタイプ (例、Name)、値の範囲 (例、1 から 99)、または特殊値 (例、Y=Yes、N=No) の指示は、フィールド・プロンプトとともに現れてはならず、選択可能な項目域でのみ表示すべきです。
- 修飾名階層の場合、プロンプト・テキストと入力フィールドの両方を 2 桁字下げしてください。
- グループ見出しの下のプロンプトを 2 桁字下げします。グループ見出しには対応する入力フィールドはありません。グループ見出しに続いてコロンを置くべきです。これは複数の行に拡張できますが、字下げレベル 1 のみが許可されます。
- 画面上の値が無意味である場合には、入力フィールドを表示しないでください。入力フィールドが条件付きで意味のあるものになる場合、フィールドが依存するプロンプトとは別に、画面上に有効なプロンプトを条件付きで発行してください。

代替案として、条件を定義するグループ見出しの下でプロンプトを字下げすることもできます。以下に例を示します。

ジョブ待ち行列への投入 _ Y=Yes, N=No
選択 Y=Yes の場合は:
完了メッセージの送信 _ Y=Yes, N=No

入力フィールド: 必須 入力フィールドでは、ユーザーが値を入力する必要があります。デフォルトでは、こうしたフィールドをサポートすることはできません。オプションの入力フィールドでは、ユーザーが値を入力する必要はありません。デフォルトを定義することによって、プログラムはオプションの入力フィールドの値を常に取得できます。

入力フィールドは 2 つの形式を取ります。最初の形式は、名前、記述テキスト、またはアドレスのようなユーザー提供の値を要求します。多くの場合、これらのフィールドは、可変の長さおよび内容の値のリストから、値を受け入れることができます。こうした有効な選択項目のリストが存在する場合、ユーザーがそのリストを要求し、それから単に選択できるように F4 がサポートされていなければなりません。

ファイル _____ 名前、リストは F4

入力フィールドの 2 番目のタイプは、固定された選択項目のセットからの選択をサポートします。CUA では、これを選択フィールドと呼びます。選択項目を 1 つだけ選択できる場合、その選択項目に番号を付ける必要があります。以下に例を示します。

書体 1 1=エリート (12 ピッチ)
2=クーリエ (10 ピッチ)
3=エッセイ・スタンダード

選択項目の番号付けの例外は、応答が Yes または No である場合、および選択値それ自体がユーザーにとって意味がある場合です。

値のリストが 5 行より長い場合、F4 がサポートされているべきです。

プロンプトで Yes または No の応答が必要な場合、Y および N が有効な項目であるべきです。以下に例を示します。

両面印刷 Y Y=Yes, N=No

画面が最初に表示された際に、ユーザーが入力する最初の入力フィールドにカーソルを配置する必要があります。これは通常、前の画面で指定されたフィールドがこの画面に繰り上がる場合を除き、最初のフィールドになります。

入力画面によっては、入力フィールドにおいて直前のユーザー入力を示すほうが望ましい場合もあります。たとえば、プロンプトを要求する前に、ユーザーが一部のコマンド値を入力した場合、それらユーザー提供の値にはプロンプトでフラグを立てます。

より大記号 (>) が標識として使用されます。入力画面様式では、最後の先行ドットと入力フィールドの先頭の間には 3 つのブランク・スペースがあります。> は、これら 3 つのブランク・スペースの真ん中のスペースに置かれます。

入力フィールドの規則: 以下の規則が入力フィールドに適用されます。

- 必須フィールドはデフォルト値を持つことができません。必須仕様のサブ値はデフォルト値を持つことができます。たとえば、ファイル名パラメーターは必須です。定義されているデフォルトのファイル名はありませんが、ファイル名のデフォルトのライブラリー修飾を指定することができます。
- 必須入力フィールドは、高輝度で表示されなければなりません。フィールド・プロンプトおよび選択可能な項目情報は、通常輝度で表示されます。エラーに値が入力されている場合、それは反転イメージおよび下線付きで再び表示されます。
- 任意指定の入力フィールドは、通常輝度で表示すべきです。
- 可能な場合にはいつでも、任意指定の入力フィールドには、定義および表示されたデフォルト値があるべきです。
- 入力フィールドの開始桁は、左寄せおよび位置合わせされている必要があります (階層フィールドがある場合を除く)。
- デフォルト値は、定義されている場合、入力フィールドに表示されるべきです。フィールドに表示される値 (文字値として表示される数字を含む) は左寄せされます (465 ページの図 130 を参照)。フィールドに表示される値が数値として指定されている場合、それらは右寄せされます。

- 入力フィールドに入力されたユーザー提供の値は、デフォルト値をオーバーライドします。
- 下線属性を使用して入力フィールドの長さを示します。指示される長さは、入力可能な文字の最大数と同じにしてください。
- 入力フィールドは、選択可能な項目情報に拡張でき、必要であれば、後続の行のフィールド・プロンプト域に移行することもできます。不定形式または継続のフィールドが別の行に移行される場合には、それを 80 桁まで拡張し、次の行の 1 桁に移行する必要があります。以下に例を示します。

```

メッセージ・テキスト . . . . . _____
_____
_____

```

選択可能な項目のリストが、拡張した長さの入力フィールドにとって適切である場合 (469 ページの『記述テキスト域 (選択可能な項目情報)』を参照)、その選択項目は入力フィールドの後ろに表示されるか、または F4 を使用してアクセスできます。

- エコー表示されたエラー (入力フィールドの正しくない値) は、反転イメージで再び表示されます。キーボードをアンロックして、ユーザーが、訂正を行う前にエラー・リセットを押さずにエラーを訂正できるようにする必要があります。カーソルは、エラーのある最初のフィールドに配置する必要があります。

注: 強調表示された入力フィールド (たとえば、必須フィールドなど) で反転イメージが使用されている場合、強調表示属性をオフにして (通常輝度に変更)、5250 ディスプレイ装置を使用した場合に何も表示されないという状態を避ける必要があります。

- フィールドが選択項目のリストで F4 をサポートしている場合、そのフィールドにエラーがあったとしても、F4 は使用可能なままでなければなりません。これは、ユーザーがそのエラーを訂正するのを助ける点で重要です。

入力画面がページング可能であり、最初のエラーが現在表示されているフィールドにある場合、特殊な位置指定を行わずに画面をそのまま表示します。最初のエラーが現在表示されているフィールドにない場合は、ページング可能な領域の上部に最初のエラーを配置してください。

記述テキスト域 (選択可能な項目情報): 選択可能な項目情報は、入力フィールドの右側の記述域に表示すべきです。ここでは、事前定義された値のセットまたは限定された範囲の値だけが許可されます。値の数が多すぎて表示できない場合、あるいは文書名のリストのように値が動的である場合、F4 がサポートされている必要があります。「リストは F4 キー」という句が、そのフィールドの選択可能な項目情報として表示されなければなりません。(F4 が画面のすべてのフィールドでサポートされている場合は、各フィールドごとに「リストは F4 キー」を表示する必要はありません。) 値に関する説明は、入力フィールドをサポートするヘルプ情報でも提供されているべきです。

以下の規則は、選択可能な項目情報に適用されます。

- 画面上のすべての入力フィールドの情報は、同じ開始位置を持っているべきです。
- 選択項目は、横方向の文字ストリングで表示される場合、1 つのコンマおよび 1 つのブランクで区切られます。

```

Y=Yes, N=No
QDKT, QTAPE1, QTAPE2, *SAVF
名前, *ALL

```

選択項目は、縦方向に表示される場合、左側および個々の行で位置合わせされます。どの行の終わりにもコンマや他の句読点は表示されません。

Y=Yes
N=No
1=エリート (12 ピッチ)
2=クーリエ (10 ピッチ)
3=エッセイ・スタンダード (プロポーショナル)
4=エッセイ・ボールド (プロポーショナル)

- それが最初の値、最後の値、または唯一の値であるかどうかにかかわらず、等号の各側の最初の語を大文字にしてください。以下に例を示します。

1-255, Blank=Entire instruction
Blank=Entire instruction, *NONE=None of instruction

- 入力できる特殊値は大文字で表示されます。

ファイル _____ 名前、*NONE

- Yes および No は Y=Yes、N=No で指定される必要があります。
- 'generic*' を使用して、総称名の後にアスタリスクを付ける必要があることを示します。

以下に例を示します。

ファイル _____ 名前、総称*

- キーの数值範囲を示します。以下に例を示します。

0-99

この範囲が重要でない場合は、それを含まないでください。たとえば、値に制限がない場合には何も表示しないでください。

- 数値のプロンプトが出され、その値が特定の単位でなければならない場合、単位のタイプを示します。以下に例を示します。

秒単位の時間。

- スペースなしですべての値を表示する一連の選択可能な項目では、最も有用な開始をデフォルト値で、終了を省略符号で表示します。
- 引用符で囲まれた語を表示するのは、プロンプトが出されている値を引用符で囲む必要がある場合のみです。

機能キー

機能キー域の定義に関するガイドラインは、486 ページの『すべての画面用の機能キー域の定義』に記載されています。

オンライン・ヘルプ情報

ユーザー・インターフェース管理機能 (UIM) を使用した、このサンプル画面のオンライン・ヘルプ情報は QUSRTOOL で入手できます。QUSRTOOL 内のソースの使用法について詳しくは、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

オンライン・ヘルプ情報の定義に関するガイドラインは、496 ページの『入力画面のヘルプ』に記載されています。

一般入力画面の操作

入力画面操作のガイドラインは以下のとおりです。

- ユーザーがデフォルト値を含むフィールドをブランクにする場合、他の検査が完了した後に再び画面を表示する際、デフォルト値をフィールドに戻してください。エラー・メッセージを発行しないでください。画面の機能は実行されず、ユーザーは戻されたデフォルト値を受け入れたり変更したりできます。

- ユーザーがデフォルト値を含まない任意指定のフィールドをブランクにする場合は、そのフィールドをブランクのままにしてください。エラー・メッセージを発行しないでください。処理は必要に応じて行うことができます。
- ユーザーが、デフォルト値を持たない必須フィールドをブランクにすると、
「ライブラリー名が必要である」

というメッセージが表示されます。

- フィールドがブランクにされ、Enter キーが押されると、表示画面の他のフィールドはエラーがないかどうか検査されます。デフォルト値を戻すと、表示画面の別のフィールドによってはエラーが発生する場合があります。これらのエラーが発生した場合、エラー・メッセージを表示する必要があります。
- いずれの場合にも、必要な値が提供されず、有効な項目が存在しない場合、何が必要かを示すエラー・メッセージとともに画面が再び表示されます。無効な入力を持つフィールドまたは入力が必要なフィールドは、反転イメージで表示されます。すべてのフィールドに有効な入力またはデフォルト値がある場合、画面は処理され、ダイアログは入力処置に対して定義された次の論理画面に進みます。

情報画面の設計

情報画面は保護情報を表示します。472 ページの図 131 は 2 つの部分で構成される情報画面を表示します。以下の例では、フィールド・プロンプトによって識別されるかまたはラベルが付けられる一連の出力フィールドを示します。

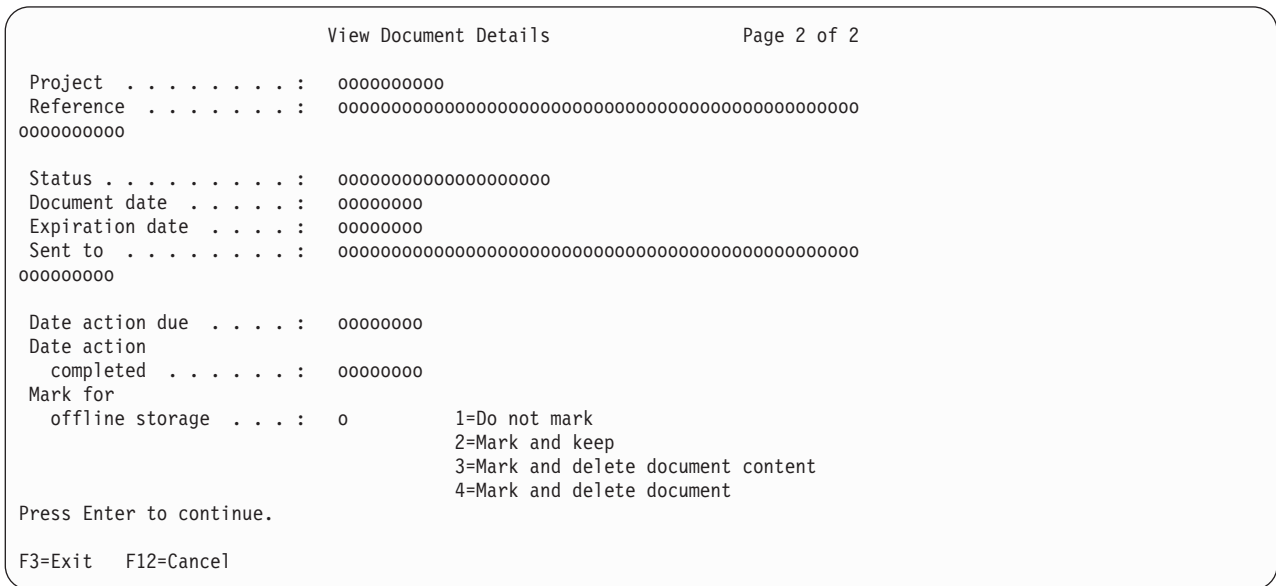
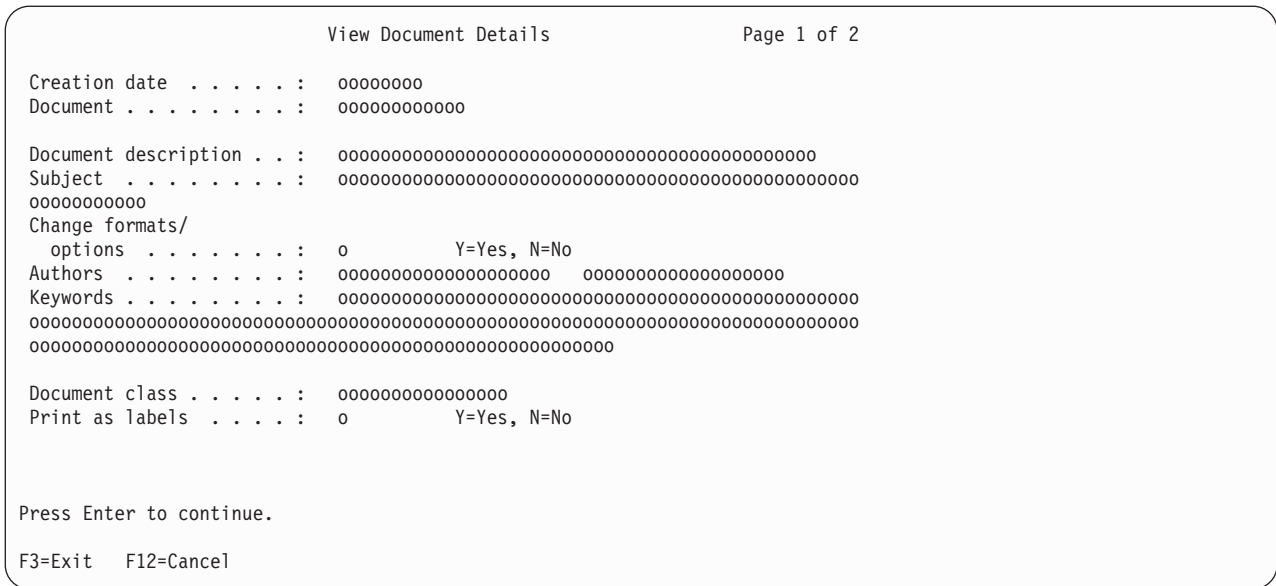


図 131. サンプル情報画面 (2 ページ)

この画面の設計を続ける前に、462 ページの『すべての画面に対する特殊機能および属性の定義』にある必要な機能および属性を (必要に応じて、オプション機能および属性も) 定義します。

サンプル画面の表示または独自に使用するためのソースのコピーについて詳しくは、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

タイトル

タイトルは、1 行目の中央に大/小文字混合で、そして高輝度で表示すべきです。

位置情報

複数の部分で構成される画面の位置情報は、1 行目で右寄せされ、次のような形式になります。

Page xx of xx

プロンプト域 1

フィールド・プロンプトのガイドラインは以下のとおりです。

- フィールド・プロンプトは、通常の文では大文字化され、プロンプトが識別するフィールドの左側にあります。
- 出力フィールドのフィールド・プロンプトは、その前に説明行は置かれず、2 桁目で開始されます。
- 一連のピリオド (ドット) およびコロンは、フィールド・プロンプトおよび出力フィールドに接続するのに使用されます。
- コロンは、フィールド・プロンプトに指定された欄の幅の最後の桁にあり、ドットは 1 桁おきにプロンプト・テキストの後ろに置かれます。プロンプト・テキストに最も近接しているドットは、プロンプトの 2 または 3 桁後ろにあります (プロンプト・テキストとドットの間には最小 1 つの空白がある)。
- フィールド・プロンプトの欄は、出力フィールドが含まれる欄から 3 つの文字位置 (空白) で区切られます。
- フィールド・プロンプトの複数の行が表示されている場合、各行のドットおよびコロンは縦方向に位置合わせされます。
- 出力フィールドは、常にコロンが先行し、フィールド長をマーク付けする下線属性はありません。
- プロンプトが複数の行を必要とする場合、2 行目を 2 桁字下げしてください。出力フィールドは最後の行の後にあります。

ファイルの

レコード様式 : _____

- 出力フィールドでは、コロンに加えて 1 つ以上の余裕がある場合にはドットを使用すべきです。コロンは必須です。出力フィールド形式では、コロンの前に 1 つの空白・スペースがなければなりません。

プロンプト : Xxxxxxx (複数のドットを置けるスペースがあるとき)
中位のプロンプト . : Xxxxxxx (1 つのドットを置けるスペースがあるとき)
長いプロンプト . . : Xxxxxxx (1 つのドットを置けるスペースがないとき)

- フィールド・プロンプトおよび対応する出力フィールドの字下げにより、階層のレベルを表示することができます。
- フィールド・プロンプトの複数の行が表示されている場合、プロンプト、ドットとコロン、および出力フィールドの開始桁は位置合わせされます。
- 複数行に移行される不定形式 (継続) の出力フィールドでの先行ドットおよびコロンの使用は、標準形式と同じです。出力フィールドの場合、2 行目および後続の行は、プロンプト・テキストの先頭から 2 桁字下げされます。

原因 : XXX
XXXXXXXXXXXXX. XXX.

- 各行ベースで形式設定された出力フィールドは、プロンプトに続く行で始まる個別の行を占有しているべきです。出力フィールドは、プロンプトの先頭から 2 桁字下げしてください。

プロンプトの後に空白なしでコロンが置かれます。プロンプトの後に先行ドットは置かれません。識別されるフィールドが同じ行にある場合を除き、先行ドットを決して使用しないでください。

連絡メモ :

XX
XX
XX
XX

- 表示される情報に複数パーツが含まれる場合、ユーザーによって認識され、単一のデータ断片と見なされるなら、そのパーツを 1 つの行に置くことができます。

社会保険番号 : XXX XX XXXX
行/桁 : 27 43

プロンプト域 2

情報画面の 2 番目の部分のプロンプトは、画面の最初の部分のプロンプトと同じ形式に従います。

指示行

複数パーツの情報画面では、指示行を使用して、画面でダイアログを継続または終了する方法をユーザーに伝えます。以下に例を示します。

続行するには、Enter キーを押してください。

この最下部指示行は、情報の最後の行の 1 ブランク行後ろに、または最初の機能キー行の 1 ブランク行上に置かれます。

機能キー

機能キー域の定義に関するガイドラインは、486 ページの『すべての画面用の機能キー域の定義』に記載されています。

オンライン・ヘルプ情報

ユーザー・インターフェース管理機能 (UIM) を使用した、このサンプル画面のオンライン・ヘルプ情報は QUSRTOOL で入手できます。QUSRTOOL 内のソースの使用法について詳しくは、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

オンライン・ヘルプ情報の定義に関するガイドラインは、497 ページの『情報画面のヘルプ』に記載されています。

一般情報画面の操作

この出力情報は、ユーザーが Enter キーを押すことによって進行できるように、個別の情報画面としてセットアップされています。ページング・モデルを使用することもできます。Enter キーが押されると、ダイアログは次の論理画面に進みます。Press Enter to continue という指示が画面に表示されるようにしてください。

リスト画面の設計

リスト画面は、ユーザーが 1 つ以上選択可能な項目のリストを表示し、それらの項目で実行する 1 つ以上の処置を指定します。475 ページの図 132 は、入力フィールドのあるリスト画面の例です。これは、混合画面と呼ばれます。混合画面は、異なるタイプの表示エレメントを結合した画面です。

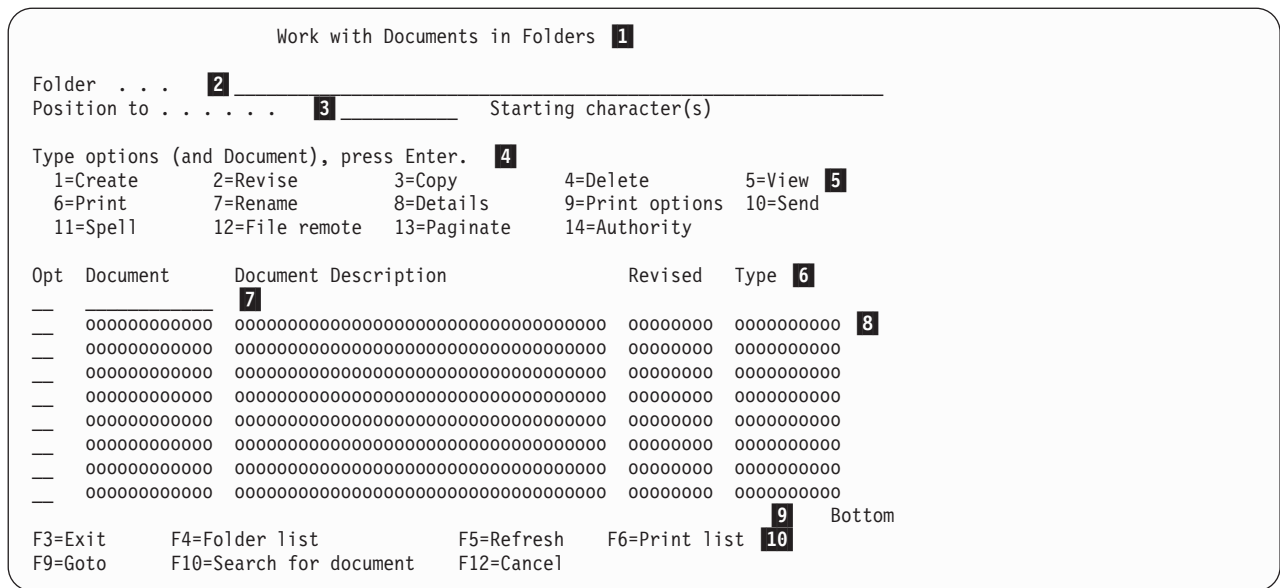


図 132. サンプル・リスト画面

参照キー

表示エレメント

- 1** タイトル
- 2** リスト制御フィールド
- 3** 位置指定 フィールド
- 4** 指示行
- 5** オプション行
- 6** 列見出し
- 7** 拡張処置項目域
- 8** リスト・フィールド
- 9** ページング位置情報
- 10** 機能キー

この画面の設計を続ける前に、462 ページの『すべての画面に対する特殊機能および属性の定義』にある必要な機能および属性を (必要に応じて、オプション機能および属性も) 定義します。

サンプル画面の表示または独自に使用するためのソースのコピーについて詳しくは、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

タイトル

タイトルは、1 行目の中央に大/小文字混合で、そして高輝度で表示すべきです。

プロンプト域

475 ページの図 132 では、プロンプト域はリスト制御フィールドおよび位置指定 フィールドで構成されます。

フィールド・プロンプトのガイドラインは以下のとおりです。

- フィールド・プロンプトは、通常の文では大文字化され、プロンプトが識別するフィールドの左側にあります。
- 一連のピリオド (ドット) は、フィールド・プロンプトおよび入力フィールドに接続するのに使用されます。最右端のドットは、フィールド・プロンプトに指定された欄の幅の最後の桁にあり、ドットはプロンプト・テキストの後ろに 1 桁おきに置かれます。プロンプト・テキストに最も近接しているドットは、プロンプトの 2 または 3 桁後ろにあります (プロンプト・テキストとドットの間には最小 1 つの空白がある)。
- フィールド・プロンプトの欄は、入力フィールドまたは出力フィールドが含まれる欄から 3 つの文字位置 (空白) で区切られます。
- フィールド・プロンプトの複数の行が表示されている場合、可能な場合には、最右端のドットは縦方向に位置合わせされます。
- その前に指示行が置かれないフィールド・プロンプト (一部の混合画面において) は、2 桁目で開始されます。
- 入力フィールドには、その長さをマーク付けする下線属性があります。

指示行

指示行のガイドラインは以下のとおりです。

- 指示行の最初の文字は 2 桁目で始まります。
- 指示行とそれに続くオプション行の間に空白行はありません。
- リスト画面の最上部指示行はピリオドで終わります。
- 指示行の前に入力フィールドなどの情報が置かれる場合、それを 1 つの空白行でその情報と区切る必要があります。
- 標準の処置リストの指示行は以下のとおりです。
オプションを入力して、Enter キーを押してください。
- 拡張処置項目を使用した処置リストの指示行は以下のとおりです。
Type options (and Xxxx), press Enter.

ここで、Xxxx は、リストに含まれる項目のタイプ (ディクショナリー、文書など) を示します。

- 指示行の後ろに置かれるオプション行がリストとともに使用される場合、指示とオプションの組み合わせは、リスト列見出しから 1 つの空白行で区切られます。
- システム画面のすべての指示行は、カラー・ディスプレイでは青で指定されます。

オプション行

オプション行のガイドラインは以下のとおりです。

- オプション行は 4 桁目で始まります。
- リストされる各オプションは、最初の文字は大文字で、その前にスペースなしで番号および等号が置かれます。リストされるオプションの間には、最小で 2 つのスペースを残さなければなりません。

- オプションの行は最大で 3 つです (オプション行を 2 行で保持することをお勧めします)。オプションを複数の行に拡張する場合、その数字は縦方向に位置合わせされます。リスト画面は、F23=More オプション機能に切り替えたほうが望ましい場合、最大で 3 つのオプション行を持つことができます。
- オプション・ラベルにはフルワード名を使用します (例、5=Display jobs、8=Save library など)。
- オプションをさらに表示しない場合、表示されている最後のオプションの後ろに省略符号を置き、F23=More オプションを表示してオプション説明の 2 番目のセットを指示域に切り替えます。
- 以下の標準化されたリスト処置コードを使用します。
 - 1 選択 (F4 キーを押すと画面で使用される)。F4 キーが押された入力フィールドに値を戻す際に選択します。処置を選択するのに使用しないでください。実際の処置には他の処置コードが使用される必要があります。これは、リスト内のオブジェクトまたは項目が、選択されている最中にのみ使用されます。
 - 1 作成。リストにある拡張処置項目を使用してリスト上で使用されます。「Create on all Work with... (すべての処理での作成...)」画面用に予約されています。
 - 2 変更または編集。
 - 3 コピーまたは保留。
 - 4 削除または終了。
 - 5 表示されている実際のオブジェクトとその内容を表示または処理します。それが内容を含むオブジェクトである場合、その内容を表示または処理します。それが説明である場合、説明される属性を表示または処理します。
 - 6 印刷または解放。
 - 7 名前変更。
 - 8 表示属性、または詳細 (2 番目の表示オプションの場合)。「Work with... (処理...)」属性、または詳細 (2 番目の処理の場合)。
 - 9 実行。
- 「Work with... (処理...)」処置は、それが表示処置に必要でない場合は、5 または 8 に割り当ててください。両方を使用する場合、処理オプションに 9 または 12 を使用できます。
- リスト画面でサポートされるオプション番号は、順序通りにリストされる必要がありますが、その順序番号ではギャップが許可されます。
- オプション・フィールドが 2 文字 (オプション番号のために) である場合、フィールドの桁への 1 桁の選択コードの入力が受け入れられる必要があります。

列見出し

列見出しのガイドラインは以下のとおりです。

- 列見出しはリスト画面で必須であり、大/小文字混合でなければなりません。
- 列見出しはデータが文字または文字と数値である場合、対応するデータに対して左寄せしてください。数値データに対する列見出しのみが右寄せされます。見出しがデータより長い場合、それをデータの中央に表示する必要があります。
- オプション・フィールドの列見出しは、*Option* とすべきです。(Option は、スペースが制限される場合、*Opt* と短縮することができます。)
- 列は、2 つ以上、7 つ以下のスペースで区切ってください。
- 列見出しは、最大で 3 行の垂直方向のサイズを占有することができます。

拡張処置項目域

拡張処置項目域では、列見出しの下の最初の行は、ページングできない項目域です。項目域にはオプション・フィールドと ID フィールドが含まれます。ID フィールドは入力フィールドであり、その長さは下線属性で示され、ページング可能なリストのそれらの行の対応する出力フィールドと同じ長さです。以下に例を示します。

OPT 文書 文書記述 改訂 タイプ

両方のフィールドには入力フィールドの属性があります。エラーの場合には、反転イメージおよびカーソル位置に下線が付きます。

リスト・フィールド

リスト域の最左端の列は、*Option* 列です。これには、使用可能なオプションの数に応じて、長さが 1 または 2 文字の入力フィールド (下線付き) が含まれます。2 文字入力フィールドのどの桁でも 1 文字項目が受け入れられます。

Option 列の右側にある列には、リスト項目の識別または順序付け (リストが配列されている場合) に関する情報が含まれているべきです。この列は通常、項目に名前を付けます。

リストは、サブファイルを使用して説明することができます (『第 4 章 サブファイルを使用したレコードのグループの表示』を参照)。しかし、サブファイルの使用は、ページングされる領域の下部にあるシステム・インターフェースによって使用される、記述ページング情報をサポートしていません。

475 ページの図 132 で表示されるサンプル・リスト画面では、使用可能なリスト行のフィールドは、1 行に 5 つのフィールドおよび各フィールドで 9 つの行を使用して記述されます。続いてプログラムは、連続する 9 行のデータ・ブロックをユーザー・ページとしてファイルを介してロードします。

Option 列のフィールドには、入力フィールドの属性があります。入出力 (B) として定義された、その他の列のフィールドには、ユーザーがアクセスできないようにする保護属性があります。

ページング位置情報

ユーザーが、ページング可能な領域の最後の部分以外の部分にある場合、最下部の区切り行の位置情報フィールドに、*More...* という句が強調表示 (高輝度) されて現れます。ユーザーが、ページング可能な領域の最後の画面にある場合、*More...* が *Bottom* によって置き換えられます。

位置情報フィールドは、最下部の区切り行の最右端の 7 表示桁として定義されます。 *More...* および *Bottom* はこのフィールドで右寄せされます。

混合画面の複数の領域でページングが可能な場合、それぞれの領域で独自のページング情報が必要になります。479 ページの図 133 は、位置情報を使用した画面のレイアウトを示します。

```

01                               画面表題
02  上部分離行
03  表示本体
04
05  - - - - - ページ送り行の始め - - - - -
06
07
08
09
10
11
16
12
13
14
15
17
18
19
20
21  - - - - - ページ送り行の終り - - - - -
22  末尾分離行                               続く...
23  機能キー域
24  メッセージ域

```

図 133. 位置情報を使用した画面のレイアウト

それぞれの許容定数は標識によって条件付けられます。プログラムは、ページングの状況に応じて、正しい定数を表示しなければなりません。

機能キー

機能キー域の定義に関するガイドラインは、486 ページの『すべての画面用の機能キー域の定義』に記載されています。

オンライン・ヘルプ情報

ユーザー・インターフェース管理機能 (UIM) を使用した、このサンプル画面のオンライン・ヘルプ情報は QUSRTOOL で入手できます。QUSRTOOL のソースの使用法について詳しくは、455 ページの『QUSRTOOL ライブラリーにおける画面例の使用』を参照してください。

オンライン・ヘルプ情報の定義に関するガイドラインは、497 ページの『情報画面のヘルプ』に記載されています。

一般リスト画面の操作

Enter キーが押された際の、混合リスト画面の処理の優先順位は以下のとおりです。

- 画面の上部にあるリスト制御フィールド (フォルダー、ディクショナリーなど)
- 位置指定プロンプト
- 拡張処置項目域
- リスト・フィールド内のオプション

エラー診断はこの同じ順序に従います。

このタイプの混合リスト画面では、画面の上部にあるリスト制御フィールドには、内容が表示されているオブジェクト (フォルダー、ディクショナリーなど) の名前が含まれます。名前を変更する (たとえば、別の

有効なフォルダー名にする) ことによって、ユーザーは本質的に、新規に指定されたフォルダーの内容に対して表示コマンドを発行していることになります。ユーザーが **Enter** キーを押すと、リスト画面域は新規のリスト画面で置き換えられます。

リスト制御フィールドの操作

リスト制御フィールドは、リストに含まれるものを指定します。たとえば、リスト制御フィールドを FOLDER1 から FOLDER2 に変更して、**Enter** キーを押すことによって、ユーザーは FOLDER2 の内容を表示する新規リストを取得します。

リスト制御フィールドの操作に関するガイドラインは以下のとおりです。

- 項目のリストが表示されると、リスト内の選択の保留中にリスト制御フィールドのリストの名前を変更することによって、エラー・メッセージが表示されます。保留中の選択が保管され、変更された名前をもつリスト制御フィールドは反転イメージで表示されます。エラー・メッセージには、変更された名前が含まれており、必要に応じてユーザーはその名前を元に戻すことができます。

名前をブランクにすることによって、保留中の操作の完了後にリストが再び表示されると、名前はリストアされます。

- フォルダー A の項目のリストが表示中であり、リスト内に入力された保留中の選択がない場合、ユーザーは、リスト制御フィールドのフォルダー A をフォルダー B に変更して **Enter** キーを押すことによって、フォルダー B から項目のリストを取得することができます。
- フォルダー A の項目のリストが表示中であり、保留中の要求がない場合、ユーザーは、フォルダー A をフォルダー B に変更し、選択されたオプション番号および ID を拡張処置項目域に入力し、そして **Enter** キーを押すことによって、フォルダー B の特定の項目で特定の機能を実行できます。オプションを実行すると、フォルダー B のリストが表示されます。
- フォルダー A をリスト制御フィールドでフォルダー B に変更し、**Enter** キーを押さずに、ページング/ロール操作が試行されると (保留中の選択があってもなくても)、ページング/ロールは実行されず、エラー・メッセージが表示され、リスト制御フィールドは反転イメージで表示されます。

リストの位置指定

リストの位置指定に関するガイドラインは以下のとおりです。

- 位置指定機能は、リストの迅速な位置変更に使用されます。
 - ユーザーが文字 S を開始点として指定した場合、結果リストは、最初の S 項目で開始され、S で始まる項目だけでなくその後のすべての項目も組み込まれます。S に位置指定されると、リストに AA、BB、SA、SB が含まれる場合、リストは SA に位置指定されます。
 - リストが位置変更されると、前ページおよび次ページ・キーを使用して、リスト内のその他の項目にページングできます。
 - リストの始めまたは終わりに位置指定できるように、特殊値 *TOP および *BOT がサポートされています。
 - 入力された文字で始まる項目が見つからない場合、リストは、要求された位置指定の直前の項目の始めに位置指定されます。
- 位置指定機能の後、リストが位置指定される項目はリスト域の上部になければならず、カーソルは入力可能フィールド (もしあれば) に従う最初のオプション・フィールドになければなりません。また、位置指定フィールドはブランクになっている必要があります。

- 位置指定フィールドでは、有効な構文名があるかどうか、またはリスト内の名前に直接に一致するものがあるかどうかは検査されません。その代わりに、ユーザーは照合順序規則に基づいて、入力されたストリングが適合するリストに位置指定されます (ユーザーは、A& を入力して、照合順序内で A& がある場所に位置指定されます)。
- リスト・オプション列の位置指定フィールド項目は無視されます。ユーザーがこのプロンプトに文字を入力し、リストのリスト域で操作が保留中の間に Enter キーを押すと、ページ/ロール・キーが押されたかのように、リストは位置変更され、選択は保管されます。

リストでの小文字名の位置指定

リストの位置指定プロンプトをサポートしており、そのリストの項目に小文字が含まれる場合、これらの規則に従ってユーザーがリストをそれらの項目に位置指定できるようにする必要があります。

- 位置指定フィールドの入力が二重引用符 (") で開始されない場合、その入力は大文字に変換されなければならない、その大文字名に最も近い名前に位置指定される必要があります。
- 位置指定フィールドの入力が二重引用符 ("b または "b") で開始される場合、そのリストは "b に最も近い名前に位置指定される必要があります。

ユーザーが終了引用符 ("b") を入力した場合、リストが二重引用符 ("b) で始まる名前に位置指定されるように、それらを除去する必要があります。そうしないと、照合順序で "b" が検索されることとなります。

- 引用符を必要とする特殊文字または小文字が含まれていないストリングで、二重引用符が使用されている場合、それらは除去され、リストは引用符によってマーク付けされていない名前に位置指定されます。

以下のリストを使用すると、

```
"aa"
"bb"
AAAAA
BBBBB
```

- 位置指定 "b および位置指定 "b" は、両方ともリストの最上部に "bb" を置くこととなります。
- 位置指定 "AAA"、位置指定 "AAA"、および位置指定 AAA はすべて、リストの最上部に AAAAA を置くこととなります。

リスト制御フィールドの変更およびリストの位置指定

リストの位置指定のオプションには以下のものが含まれます。

- リスト制御フィールドが有効な値に変更され (新規リストを表示可能)、位置指定フィールドに値が入力されると、位置指定フィールドに従って位置指定された新規のリストを表示します。
- リスト制御フィールドが無効な値に変更され、位置指定フィールドに値が入力されると、以下のようになります。
 - 同じリストを表示する
 - リスト制御フィールドを反転表示し、メッセージを表示する
 - 位置指定フィールドは入力されたままになり、リストは位置指定されない

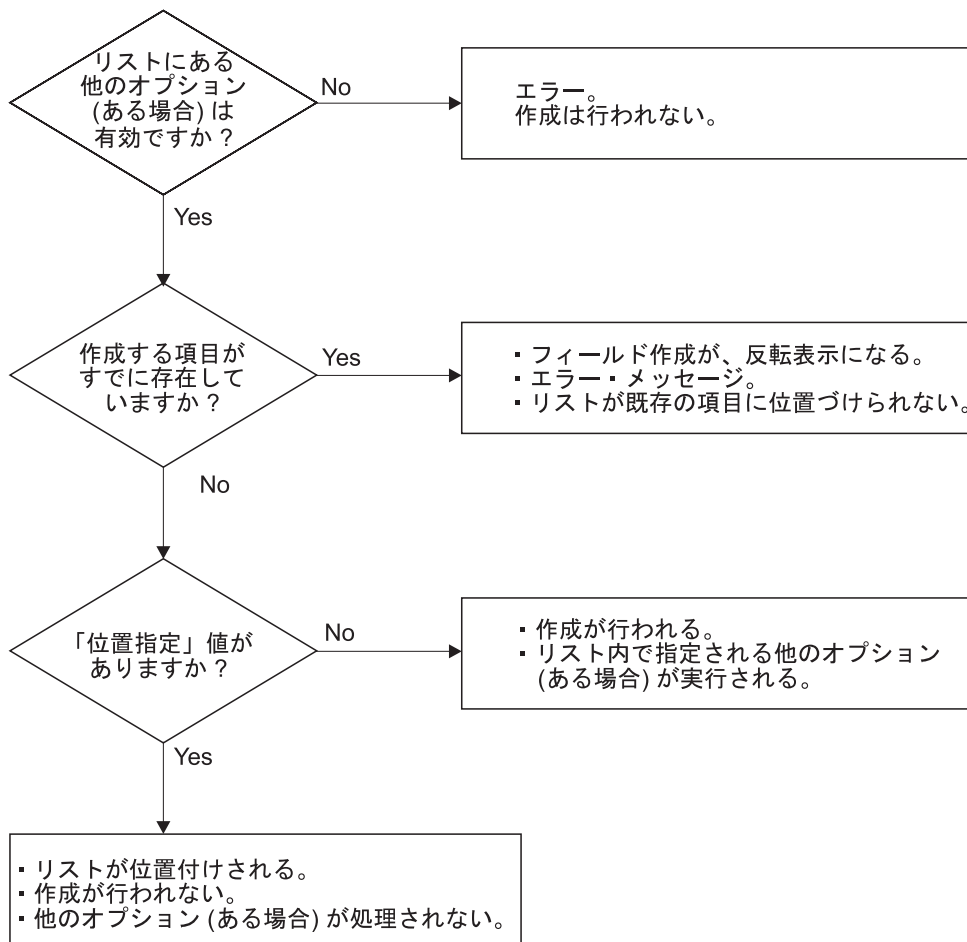
拡張処置項目域の操作

拡張処置項目域の操作に関するガイドラインは以下のとおりです。

- カーソルは、最初は拡張処置項目域のオプション・フィールドに位置指定されます。カーソルは、以下の操作のいずれかの後にこの位置に戻ります。
 - 位置指定

– ページング操作

- 図 134 は、拡張処置項目域で作成機能を使用した際にシステムによって使用される処理の優先順位を示します。



RV3W066-0

図 134. リスト画面を使用した処理の優先順位の例

- 拡張処置項目域以外のリスト行で作成オプションを指定すると、エラーが発生します。
- 作成オプションを指定すると、ユーザーによって識別された単一の新規リスト項目が作成されます (拡張処置項目域を使用)。プログラムが、リスト項目を作成するのに他のユーザー提供の情報を必要とする場合、プログラムによってその情報を提供する入力画面がユーザーに提供される必要があります。
- ユーザーは、拡張処置項目域を使用して、そのリストに有効な他のオプションを入力することができます。ユーザーは、処置を実行するリスト項目の名前を拡張処置項目域の入力フィールドに入力することによって、その項目を識別しなければなりません。
- 拡張処置項目域に入力されたオプションは、リスト・プロパティ内の任意項目に対して有効です。ユーザーは、リスト・プロパティで識別された項目に対してオプションを実行できます。
- 拡張処置項目域において、オプションを入力して ID を入力しなかった場合、ID を指定しないと機能が実行できないことを示すエラー・メッセージが表示されます。例外が 2 つあります。
 - ID が、一緒に表示される入力画面 (ユーザーに対して常に表示される) でプロンプトが出される項目の 1 つである
 - プログラムが、後で指定可能な一時オブジェクト (照会など) の作成をサポートしている

- ID が入力されておりオプションが入力されていない場合、ID フィールドの内容に対する操作は実行されません。
- ID フィールドにはデフォルト値を含めることができます。リスト処理が実行されると、拡張処置項目域のオプション・フィールドはブランクに復元されます。ただし、ID フィールドはデフォルト値を保持できます。

オプションが指定されている場合のリスト操作

以下の規則は、リスト内で項目 (オブジェクト) に対して指定されたオプション (処置) が含まれる一般的な操作をカバーしています。

リスト処理では次の一般規則に従ってください。

- リスト処理の順序は、リスト項目が表示画面に表示される順序と一致しているべきです。つまり、最上部行から最下部行 (1 行に複数の項目がある場合には左から右) までが一致している必要があります。
- 処理後には、処理前に表示されていたのと同じリストのビューが再び表示されるはずですが (エラーが発生した場合を除く)。

リスト操作の特定の規則は以下のとおりです。

- ユーザーは、複数のリスト項目に同じまたは異なるオプションを選択することができます。選択はページング可能などの項目画面でも行うことができます。オプションは、(リスト画面またはプロンプト・キーが押された後に表示された入力画面で) Enter キーが押されるまで処理されません。
- リストで入力可能なオプション数の制限をもうけるべきではありません (選択オプションを除く)。
- ユーザーが操作を実行するための適切な権限を持っていない場合、メッセージ行でそのようなメッセージを表示します。
- 操作はリスト域で表示されている順に実行されます。

注: リスト画面でサポートされる一部の操作 (削除など) は、確認用に表示される際に別の画面でグループ化することができます。確認は操作のグループに対して行われますが、各操作はリストで表示されている順に実行されます。

- リスト処理 (最初のページ以外のページから開始) が完了すると、ユーザーはリストの同じページに戻り、リスト位置は変更されません (エラーが発生した場合を除く)。ただし、ページの最上部の項目が削除操作の結果として除去されている場合、リストは、削除された項目の前の残っている次の項目がリストの最上部にくるように位置指定されます。

処理 (最初のページから開始) が完了すると、リストは、項目がリストの始めに追加または削除された場合でも、リスト内の現在の最上部項目で位置指定されます。

- ページ/ロール・キーは、リスト全体で前方および後方に移動するのに使用されます。ページ/ロール・キーを押すとページ全体が移動します (すべての項目が置き換えられます)。
- リストがユーザーに 1 つの操作の実行のみを許可している場合でも、Enter キーが押されるまで処理は行われなため、ユーザーは、1 つの項目が選択された後でもロールを許可されているべきです。
- ページ/ロール・キーが押されたときに、行われた選択はすべて保管されるべきです (操作は実行しません)。
- ページング可能なリストは、リストの先頭または終わりで循環しません。
- 複数の画面操作を実行する場合、それらは一度に 1 つずつ行われるべきであり、それぞれの操作の完了はユーザーが最後の画面で Enter キーを押すことによって示されます。
- ユーザーがリスト処理を完了前に中断した場合 (たとえば、中間画面で F3 キーまたは F12 キーを押すことによって)、リストは、処理されたすべてのオプションをブランクにしたまま表示されます (*Option*

列で)。未処理のオプションはすべてマーク付けされたままです。F3 キーまたは F12 キーが押された (カーソルが位置指定された位置で) 際に処理中であったオプションは、処理されたと見なされ、マークは付けられません。

- 処理エラーが起これなければ、リストが再び表示される前にすべての処理は完了します。
- 処理エラーが検出されると、処理はその時点で中断され、ユーザーはそのエラーを処理することができません。エラーに対応するオプション・フィールドを反転イメージで表示し、そのカーソルをそのフィールドに位置指定した状態で、リストを再び表示できます。ユーザーがエラーを処理できるようにする個別の画面も表示されます。該当するエラー・メッセージは常にメッセージ行に表示されます。ユーザーがエラー状態を解決すると、リスト処理は再び開始されます。
- 複数のエラーが検出されると、エラーに対応するすべてのオプション・フィールドを反転イメージで表示したまま、リストが表示されます。カーソルは、エラーのある最初のフィールドに位置指定されます。

リスト画面がページング可能であり、最初のエラーが現在表示されているフィールドにある場合、特殊な位置指定を行わずに画面をそのまま表示します。最初のエラーが現在表示されているフィールドにない場合は、ページング可能な領域の上部に最初のエラーを配置してください。

- *Option* 列の操作が実行される際にアプリケーションがリストを動的に更新するか、またはユーザーが F5=Refresh を押すまで構成されたリストの表示を待機するかのいずれかを選択することができます。各オプションは、次のように説明されます。
 - リストの動的更新を行うことができるのは、リスト内の項目数が多くなく、更新を操作の実行時に大きな遅延なしで行うことができる場合です。
 - 項目数が多く、リストの更新によって大きな遅延が生じた場合、リストはユーザーが F5=Refresh を押した場合にのみ更新されます。動的更新を使用しない場合、ユーザーが操作して (1 つ以上の項目の横にあるオプションを入力することによって) その結果変更された項目は、どの操作が項目に対して行われたかに関して注釈を付ける必要があります。注釈は、項目の右側にある状況フィールドで行ってください。注釈は操作のタイプによって行うべきです (たとえば、削除または変更などの、項目またはその状況を変更する操作には注釈を付ける必要があります。表示などの操作には注釈を付ける必要はありません)。操作が失敗した項目には注釈を付けないでください。注釈は、項目で実行された操作のタイプを示している必要があります。項目が終了した場合には (終了)、保持された場合は (保留)、変更された場合は (変更)、または解放された場合には (解放) を表示します。実行される操作に対応する省略語 (Cn1、H1d、Chg、R1s) は、必要に応じて使用できます。

カーソル位置の規則

カーソル位置の規則は以下のとおりです。

- リスト画面が最初に表示されるとき、リスト内の最初の項目のオプション・フィールドにカーソルを位置指定します。画面の最上部になんらかの 2 次フィールド (例、位置指定 など) がある場合でもこの規則に従ってください。
- リストの位置指定またはページ/ロール機能の後にリストが表示されたら、カーソルをリスト域の最初のオプション・フィールドに位置指定します。リスト位置指定機能の後、リストが位置指定される項目はリスト域の最上部にあり、カーソルはその項目のオプション・フィールド内にあるべきです。
- リストの別のビューが表示された場合でも、カーソルは現在の項目にとどまっているべきです (カーソルが始めからリスト域になかった場合を除き、カーソルはリスト内の最初の項目に位置指定されています)。
- 選択された操作が実行され、リストが再び現れると、以下の規則が適用されます。
 - ユーザーは常に、Enter キーが押されたリストの同じページに戻ります。これは基本的な規則です。

- 選択された最後の項目の近くにカーソルを置きます。その項目が現在の画面上になれば、現在の画面上の最初の項目の近くにカーソルを置きます。
- Enter キーが押される前にリストが最上部に位置指定されていた場合、処理の完了時にリストは最上部にとどまります。たとえば、ある項目がリストの先頭に追加された場合、新規の項目で開始するようにリストを位置指定します。ユーザーは、最上部に位置指定された場合にリストの最上部が表示されると期待します。
- エラーが検出されたら、エラーがある最初のオプションにカーソルを位置指定します (エラーがあるオプションは反転イメージで表示されているはずです)。リストを再び位置指定する必要があるのは、エラーの原因となっている項目が現在の画面ページにない場合のみです。

エラー条件の規則

エラー条件の規則は以下のとおりです。

- ページ/ロール・キーが押されると、オプション番号フィールドは有効な入力について検査されます。
- 終了機能または最新表示機能が実行されると、画面上の項目は処理されず、ユーザーにエラーは戻されません。
- エラーが発生すると、エラーに対応するオプション・フィールドは反転イメージで表示され、カーソルはそのフィールドに位置指定されます。また、該当するエラー・メッセージがメッセージ行に現れます。

リスト画面がページング可能であり、最初のエラーが現在表示されているフィールドにある場合、特殊な位置指定を行わずに画面をそのまま表示します。最初のエラーが現在表示されているフィールドにない場合は、ページング可能な領域の上部に最初のエラーを配置してください。

- 最初のエラーが現在表示されているフィールドにある場合、特殊な位置指定を行わずに画面をそのまま表示します。最初のエラーが現在表示されているフィールドにない場合は、ページング可能な領域の上部に最初のエラーを配置してください。
- 各製品は、一度に 1 つのリストで作成できる選択数についての内部的な限界を持つことができます。ユーザーがこの限界を超えると、エラー・メッセージが表示されます。このエラー・メッセージには、以下の項目を含めます。
 - 許可されている最大選択数を超過しているという事実。
 - この最大数を超過した量 (除去する選択数をユーザーが知るため)。

このメッセージが表示されたら、保留中の選択は反転イメージで表示しないでください。

- ユーザーがページング可能な領域の最上部または最下部を超えてページングしようとしたら、以下のエラーを発行します。
 - 最上部メッセージ

メッセージ・テキスト: すでに区域の先頭が表示されています。

原因: 領域を逆方向に戻すキーを押しています。ただし、すでにその領域の先頭が表示されているので、その方向に移動することはできません。

リカバリー: 表示された情報を別の領域で逆方向に移動するには、カーソルをその領域に移動して、もう一度前ページ・キーを押してください。

- 最下部メッセージ

メッセージ・テキスト: すでに領域の末尾が表示されています。

原因: 領域を正方向に移動するキーを押しています。ただし、すでにその領域の最後が表示されているので、その方向に移動することはできません。

リカバリー: 表示された情報を別の領域で正方向に移動するには、カーソルをその領域に移動して、もう一度次ページ・キーを押してください。

1 つの項目のみが選択可能なリスト

複数の項目が選択され、Enter キーが押されると以下ようになります。

- すべての選択にマークが付けられた状態でリストが再び表示される
- すべての選択が反転イメージで表示される
- 次のメッセージが発行されます。1 つの選択項目しか使用することができません。

1 つ以上の項目が選択され、ロール・キーが押された場合は以下ようになります。

- メッセージは出されない
- ロールが実行される

リストが空の状態でのリスト形式

ユーザーが項目のリストを要求し、使用可能な項目がなかった場合、規則は次のとおりです。

1. リスト域の列見出しに従って、テキストを 2 行目の 4 桁目で開始します (1 つのブランク行を置く)。1 つのリスト行だけが使用可能な場合、列見出しの次の行にテキストを置いてください。

```
OPT   文書       日付   テキスト
```

```
  -   - ...ブランク行...  
      (フォルダーに文書がありません)
```

2. テキストの属性は、通常の強調およびカラー (緑) で表示してください。
3. カーソルがリスト域内にある間にヘルプ・キーが押されると、リストが空ではないと見なされ、処理が実行されます (オンライン・ヘルプ情報は列で使用可能)。
4. テキストは汎用で、リストが空であることを示すべきですが、その理由について示す必要はありません。テキストは括弧で囲んでください。
5. テキストは、No xxxxxxxx で始めてください。ここで、xxxxxxx は表示されない内容を示します。句は、完全な文ではないためピリオドを続けなくてください。以下に例を示します。
(ライブラリーにオブジェクトがありません)
6. リストが空である理由に関する情報を示したほうが望ましい場合、メッセージ行のメッセージ内でそれを表示します。

すべての画面用の機能キー域の定義

機能キー域のガイドラインは以下のとおりです。

- アクティブなキーの割り当てを、数字順に左から右に、2 桁目からリストします。アクティブな F キーは、キーの続き機能によって別のセットのキーで表示される場合を除き、省略することはできません (F1 を除く)。機能の説明の最初の語の最初の文字を大文字にしてください。
- 位置合わせに必要な場合を除き、キー割り当ての間に 3 つのスペースを置いてください。
- 機能キーを 2 行に表示する場合、キー割り当てを左側に (F の上で) 位置合わせします (可能な場合)。

```
F3=XXXXXXXXX  F4=XXXXX    F5=XXXXXXXXXXXX  F6=XXXXXXXXXXXX  
F7=XXXXXXX   F12=XXXXXXX F13=XXXXXXX
```

- キーを数字の昇順で表示します (順序のギャップは存在できます)。

- 2 行の機能キー域の 1 行に非常に長い機能キー域が含まれる場合 (つまり、最大 7 つのブランクのギャップが強制される)、この説明は、位置合わせの目的で別の行の複数の機能キー説明にまたがる場合があります。以下に例を示します。

F3=終了 F4=プロンプト F9=コマンドの複写 F12=取り消し
F13=情報援助 F16=システム・メインメニュー

- すべてのアクティブ・キーを 2 行で表示できない場合、「F24=キーの続き」をサポートします。表示される内容のガイドラインは以下のとおりです。

- Enter キー、Help キー、およびロールまたはページ・キーは表示しないでください。
- アクティブな機能キーを、複数の 1 行または 2 行セットに表示します。「F24=キーの続き」キーを使用して、機能キーの次のセットを画面の機能キー域に表示します。すべてのセットは、同じ行数 (1 または 2) を確保する必要があります。

F3=XXXXXXXXX F4=XXXXX F5=XXXXXXXXXXXXX F6=XXXXXXXXXXXXX
F7=XXXXXXX F12=XXXXXXX F24=キーの続き

- 「F3=終了」および「F12=取り消し」は 1 次セットに表示しなければなりません (およびそのセット内でのみ)。F24 だけがすべてのセットで表示されます。
- 機能キーは各セットで数字順に表示されている必要がありますが、アプリケーションは各セットにどのキーが存在するかを判別することができます。1 次セットに表示される機能は、最もよく使用される機能にすべきです。
- システム画面に表示される機能キーは、カラー・ディスプレイでは青で指定されます。

任意指定のコマンド行および ID フィールド

形式が少し異なる他のシステム・メニューを見たことがあるかもしれません。463 ページの図 129 は、アプリケーション内のメニューを説明したもので、作成すべきメニューのタイプです。システム・メニューでは、左上隅に追加のシステム ID があります。これは、GO コマンドと一緒に名前として使用されます。こうした名前は、システム・テーブルでオブジェクト名として提示される必要があります。

システムは、GO コマンドによってアクセスされるメニュー上でのみ、この ID フィールドの名前を使用します。アプリケーション・メニューでこの ID フィールドを使用すると、名前は GO コマンドとともに機能しないためユーザーが混同してしまう可能性があります。ユーザーはシステム・メニューにアクセスでき、メニュー ID が与えられます。GO コマンドを使用して、ユーザーは特定のメニューをメニュー ID で指定できます。または、特定のメニューが不明である場合には、総称 ID を使用できます。この場合、「メニューの処理」画面が表示され、ユーザーはこのリストから実行するメニューを指定することができます。ディスプレイ・ファイル・メニューの作成方法について詳しくは、261 ページの『第 9 章 ディスプレイ・ファイルを使用したメニューの作成およびアクセス』を参照してください。

システム・メニューはまた、コマンド行をメニュー選択の入力フィールドとして使用します。DDS が提供する唯一のコマンド域のサポートは、コマンドを入力できる入力フィールドを (フィールド・プロンプトとともに) 定義することです。この項目域をコマンド域としてサポートするすべてのコードを書いてください。これには、システムへコマンドを渡すこと、プロンプト機能の処理、検索機能の処理、およびコマンドのヘルプの処理が含まれます。490 ページの『使用可能なコマンド行ツール』では、コマンド行をアプリケーション画面に書き込むのに役立つ、QUSRTOOL ライブラリーで使用可能なコマンド行機能に関する情報を提供しています。

アプリケーション内からコマンドをサポートするには、使用可能なコマンド行を使用して 1 つ以上のシステム画面を作成するか、1 つ以上のアプリケーション画面でコマンド行を提供します。

注: コマンドによってアプリケーションが従属する何かの変更されると、予測不能な結果が起こる可能性があります。

i5/OS システムを使用して、希望するシステム・コマンドおよびアプリケーション・コマンドに、コマンド入力を制限することができます。通常、これはユーザーにとって最良の方法です。図 135 にあるように、ユーザー・プロファイル作成 (CRTUSRPRF) 画面の「機能の制限」プロンプトで *YES を指定することによって特殊コマンドを制限できます。続いて、コマンド作成 (CRTCMD) コマンドを使用して、コマンドを作成することができます。489 ページの図 136 および 489 ページの図 137 にあるように、「制限ユーザー可能」プロンプトで *YES を指定して、コマンドを制限ユーザーに対して使用可能にします。

注: 大半のコマンドは ALWLMTUSR(*NO) で出荷されます。ALWLMTUSR(*YES) コマンドのリストについて詳しくは、ユーザー・プロファイル作成 (CRTUSRPRF) 画面の「機能の制限」(LMTCPB) プロンプトを参照してください。コマンド変更 (CHGCMD) コマンドを使用して IBM 提供のコマンドを変更できます。IBM 提供のコマンドを変更する場合、コマンドはそれぞれのリリースで置き換えられるため、制御言語 (CL) プログラム内での変更を記録しておく必要があります。

ユーザー・プロファイル作成 (CRTUSRPRF)

選択項目を入力して、実行キーを押してください。

ユーザー・プロファイル		名前
ユーザー・パスワード	*USRPRF	文字値 , *USRPRF, *NONE
パスワードを満了にセット	*NO	*NO, *YES
状況	*ENABLED	*ENABLED, *DISABLED
ユーザー・クラス	*USER	*USER, *SYSOPR, *PGMR...
援助レベル	*SYSVAL	*SYSVAL, *BASIC, *INTERMED...
現行ライブラリー	*CRTDFT	名前 , *CRTDFT
呼び出す初期プログラム	*NONE	名前 , *NONE
ライブラリー		名前 , *LIBL, *CURLIB
初期メニュー	MAIN	名前 , *SIGNOFF
ライブラリー	*LIBL	名前 , *LIBL, *CURLIB
制限機能	*YES	*NO, *PARTIAL, *YES
テキスト ' 記述 '	*BLANK	

終り

F3= 終了 F4=プロンプト F5= 最新表示 F10= 追加のパラメーター
 F12= 取り消し F13= この画面の使用法 F24= キーの続き

図 135. ユーザー・プロファイル入力画面の作成

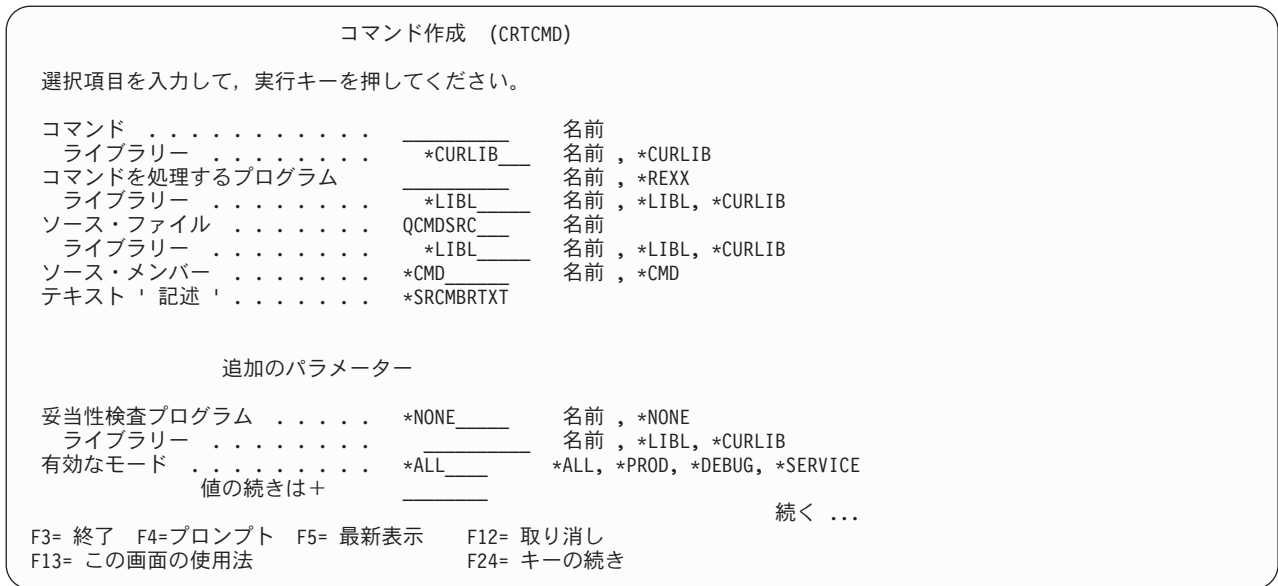


図 136. 選択される追加パラメーターを使用したコマンド作成画面

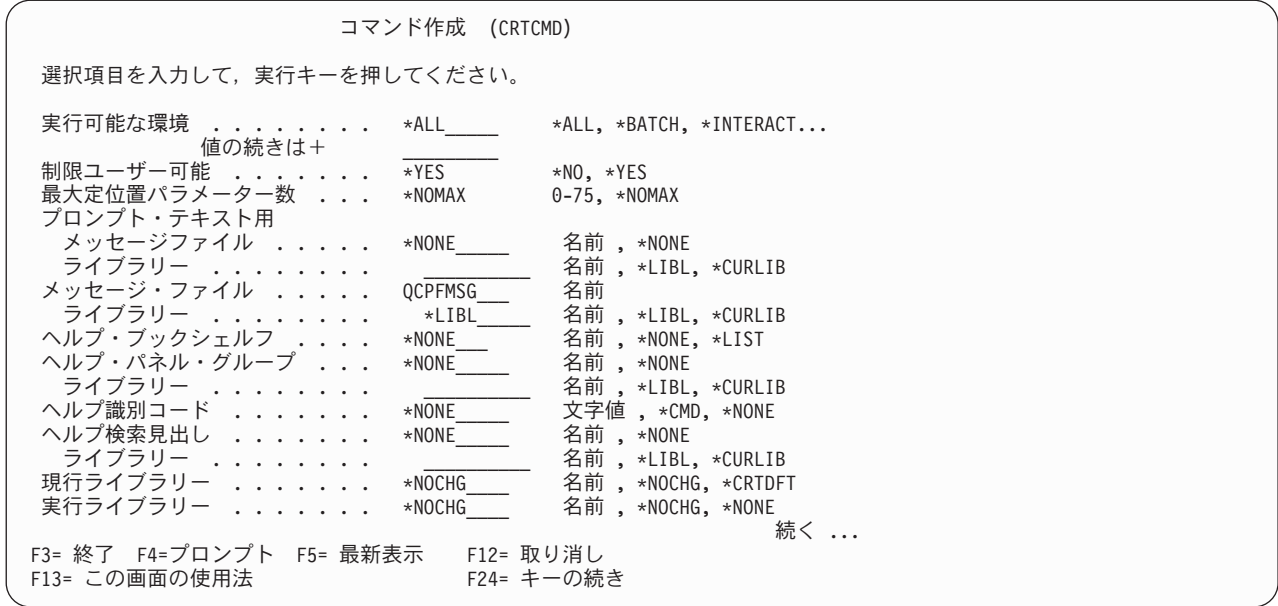


図 137. 追加パラメーターの 2 番目の画面

アプリケーションがコマンドを使用する場合、画面上で入力フィールドを (フィールド・プロンプトとともに) 定義する必要があります。i5/OS システムに一致させるには、コマンド行が次のように表示されるようにします。

```
Selection or command
====> _____
```

- 1 行のコマンド行は、機能キー域のすぐ上の行の 7 桁目で始まり、同じ行の 79 桁目で終わります。

- コマンド行は、コマンド行プロンプトで識別され、2 桁目で始まります。コマンド行を使用してメニュー選択またはコマンドのいずれかを入力する場合、プロンプトのテキストは次のようになります。

Selection or command

- コマンド入力矢印 (====>) は、コマンド行の最初の行の 2 桁目から 5 桁目にあります。
- コマンド入力矢印 (====>) はコマンド行のように通常の輝度です。

使用可能なコマンド行ツール

汎用プログラミング・インターフェース

QUSCMDLN プログラムは、コマンド行が含まれるポップアップ・ウィンドウを表示するのに使用できます。コマンド行はシステム・コマンドを入力するのに使用できます。QUSCMDLN はどのユーザー・プログラムでも呼び出すことができます。QUSCMDLN についての詳細情報は、「API」トピックにあります。

汎用プログラミング・インターフェースの終了

i5/OS システム上の QUSRTOOL ライブラリーには、アプリケーション画面上でコマンド行機能 (ファイル QATTINFO、QUSRTOOL のメンバー CMDLINE) を書き込むのに役立つ、コマンド行機能と文書が含まれています。この機能にアクセスする 1 つの方法は、図 138 にあるように PDM を使用したメンバーの処理 (WRKMBRPDM) 画面を使用することです。

PDM 使用のメンバーの処理 (WRKMBRPDM)

選択項目を入力して、実行キーを押してください。

ファイル	QATTINFO__	*PRV, 名前
ライブラリー	QUSRTOOL__	*PRV, 名前, *LIBL, *CURLIB
メンバー	CMDLINE__	*ALL, 名前, * 総称 *...
メンバー・タイプ	*ALL_____	*ALL, タイプ, * 総称 *...

終了

F3= 終了 F4=プロンプト F5= 最新表示 F12= 取り消し
F13= この画面の使用法 F24= キーの続き

図 138. 「PDM を使用したメンバーの処理」入力画面

共通キーの割り当て

491 ページの表 45 は、共通機能キーの割り当てを要約したもので、それぞれのキーがいつ使用されるかを示しています。CUA として識別されるキーは、定義された機能に対してのみ使用できます。その機能が使用できない場合、キーを別の機能に使用しないでください。

490 アプリケーション表示プログラミング

一般的に、CUA 以外のキーはシステム上で特定の機能に使用されます。機能が画面で使用可能な場合、キー割り当てはシステム・キー割り当てとの互換性のために使用する必要があります。機能が使用されていない場合、アプリケーション固有の機能のためにキー割り当てを使用できます。

表 45. 機能キーの割り当て

名前	キー	CUA	使用法/機能
ヘルプ	F1	はい	すべての画面でアクティブですが、画面上に表示されません。F1 もヘルプ機能にマップされますが、機能キー域には表示されません。
セット 1/セット 2	F2	はい	24 個のキーに割り当てることができる機能キーよりも多くの機能キーをアプリケーションが必要とする場合に必要です。 F1=Help、F2=Set n、F3=Exit、F9=Retrieve/command、および F12=Cancel は両方の機能キーのセットで必要です。
終了	F3	はい	通常はアクティブであり、すべての画面で表示されます。F3 は、アプリケーションまたはアプリケーション内の作業単位を終了します。アプリケーションを終了すると、ユーザーはアプリケーションが要求された画面に戻ります。アプリケーション内の作業単位が終了すると、ユーザーは、要求中の作業 (通常は基本メニュー) のフォーカス点として機能するアプリケーション内の事前定義ポイントに戻ります。ユーザー入力は通常廃棄されますが、Exit キーを押すとデータが失われたり、拡張ユーザーによる検索または再構成の試みが必要になる可能性がある場合、終了画面を指定して終了前にデータを保管するオプションをユーザーに提供する必要があります。 終了機能は、次の例外を除き、すべての画面で使用可能でなければなりません。 <ul style="list-style-type: none"> 削除処置の確認に使用される画面 F4 を押した結果として表示されるリスト画面 終了前にデータを保管するオプションをユーザーに提供するのに指定される終了画面 以下は、特定の画面タイプでの Exit キーの使用について説明したものです。 <ul style="list-style-type: none"> メニュー。メニューが、サインオン後に 1 次メニューまたは初期メニューで開始されたダイアログの一部である (たとえば、1 次メニューまたは初期メニューが現在のメニュー・スタックの最初のメニューである) 場合、この 1 次メニューまたは初期メニューに戻ります。1 次メニューから Exit キーを押すと、現在のダイアログは終了し、ダイアログが入力された画面に戻ります。 リスト画面。メニュー、またはリスト画面が要求された他の画面に戻ります。ユーザー入力は廃棄されます。 入力画面。メニュー、または入力画面 (または入力画面のセット) が要求された他の画面に戻ります。ユーザー入力は廃棄されます。ただし、終了画面を指定して、項目または変更を保管するオプションをユーザーに提供することができます。 情報画面。メニュー、または情報画面 (または情報画面のセット) が要求された他の画面に戻ります。 すべてのヘルプ画面。ヘルプが要求された画面に戻ります。

表 45. 機能キーの割り当て (続き)

名前	キー	CUA	使用法/機能
プロンプト	F4	はい	<p>有効な項目が認識されるものの画面に表示されない入力フィールドが画面に含まれる場合に、常にサポートされるべきです。F4 は、カーソルが現在配置されているフィールドに関連した選択可能な値のリスト (通常は名前のリスト) を提供します。ユーザーがリストから 1 つ以上の値を選択した場合、それらの値はフィールドの入力値として使用されます。</p> <p>F4 が押されても、カーソルを含むフィールドでその機能がサポートされていないと、ユーザーはエラー・メッセージを受け取ります。</p> <p>この機能がカーソルを含むフィールドでサポートされており、デフォルト値が存在するかまたはユーザーがフィールドで入力を行った場合、フィールド内の値は保持され、リストが表示されます。ユーザーがリストから値を選択した場合は、この値はフィールド内の前の値を置き換えます。ユーザーが値を選択しない場合は、前の値が保持されます。</p> <p>ユーザーがフィールドをブランクにしたまま F4 を押した場合、値を選択するリストも表示されます。</p> <p>コマンドをシステム・コマンド行で入力して F4 を押すと、コマンドのプロンプト (コマンド・パラメーターの入力フィールドを使用した入力画面) が表示されます。</p>
再表示	F5	はい	<p>再表示機能が適切な場合に使用します。F5 は、入力画面でデフォルトのリセットを開始したり、表示中のシステム出力を更新 (再表示) することができます。使用中のプログラムは、画面タイプに応じて適切なものを決定します。</p> <ul style="list-style-type: none"> • メニューおよびヘルプ画面。 通常は使用されません。 • 入力画面。 ユーザー入力をクリアし、画面が最初に表示された際に指定された元の値で画面を再び表示します。カーソルは、画面の最初の論理入力フィールドに位置変更されます。入力フィールドがページング可能な画面上にある場合、再表示によって入力フィールドの先頭にカーソルが位置指定されます。 • リスト画面。 ユーザー入力をクリアし、画面を更新して現在の状況を表示します。再表示では、リスト内の同じ位置を保持すべきです。リストは再び表示されますが、F5 が押された際にリスト・ページの先頭にあった項目はリスト・ページの先頭に依然として残っており、カーソル位置は先頭のリスト項目に位置指定されたままです。 • 出力専用フィールドを使用した画面 (データ出力項目)。 情報変数フィールドを更新して現行値を表示します。情報変数フィールドがページング可能な画面上にある場合、再表示ではフィールドのリスト内の同じ位置を保持すべきです。

表 45. 機能キーの割り当て (続き)

名前	キー	CUA	使用法/機能
			<p>再表示機能を使用したリストの位置指定規則の要約</p> <ol style="list-style-type: none"> 1. 画面は再び表示されますが、画面の最上部にあった項目は依然として画面の最上部に残っています。 2. その項目がもはや存在しない場合、それに続く項目は画面の最上部に表示されます。 3. リスト内の最初の項目が最上部にあった場合、それは再び最上部に表示されます (新規の最初の項目である場合でも)。 4. 画面の最上部にあった項目がもはや存在せず、それに続く項目も存在しない場合、画面は値のセット全体の最初の項目を画面の最上部に残したまま表示されます。 5. 画面の最上部にある項目の位置を決定する属性が変更されると、リストは古い項目があった位置で開始される項目とともに表示されます。
作成	F6	いいえ	<p>オプションです。F6 は、I=Create を使用した組み合わせリストに変換されていないリスト画面での作成機能に使用されます。ただし、組み合わせリストの使用は i5/OS システムで優先されます。F6 の使用法の要約は次のとおりです。</p> <ul style="list-style-type: none"> • カーソルがどの位置にあっても要求することができます。 • 追加オブジェクトを定義できる個別の画面に進みます。 • 完了時に、作成要求が行われたリスト画面を再び表示します。 • リストにソートされる作成済みオブジェクトを表示します。 • F6 が押された際に保留中だった要求は処理すべきではなく、保留中のままにします。 <p>注: 新規の項目を大量に追加できる (配布リストに項目を追加するなど) 場合、他の保留中の要求が完了するまで追加または作成機能を使用できないようにするのが適切なアプリケーション決定でしょう。この場合、メッセージを発行できるのは、F6 が要求されて他の要求が処理中である場合です。</p>
検索	F9	はい	<p>コマンド行を使用するすべてのパネルまたはポップアップ・ウィンドウで要求されます。</p>
コマンド	F9	はい	<p>コマンド行が含まれるポップアップ・ウィンドウを表示するのに使用します。</p>

表 45. 機能キーの割り当て (続き)

名前	キー	CUA	使用法/機能
取り消し	F12	はい	<p>アクティブでなければならず、すべての画面で表示される必要があります。F12 は、前の論理画面に戻ります。これは、次の画面で説明されているとおり、インターフェースで 1 つ前のステップに戻る基本的なナビゲーション機能を提供します。</p> <ul style="list-style-type: none"> • メニュー。前のメニュー、またはメニューが直接開始された他の画面に戻ります。ユーザー入力は廃棄されます。 • 項目およびリスト画面。前の画面に戻ります。ユーザー入力は廃棄されます。(重大なデータが失われることになる場合は、終了画面を表示すべきです。) • 情報画面。前に表示された画面に戻ります。 注: 情報のセット (入力または出力フィールドを含む) がページング様式で論理的に表示可能な場合、Enter キーおよび取り消しキーではなくページング可能な画面を使用して、情報全体の前方および後方のナビゲーションを行います。 <p>この機能は、論理取り消し機能が他の要因によって妨げられる場合を除き、すべての画面で使用可能でなければなりません。一例として、終了機能のみが適切である場合があります。これは、データを保管するかどうかをユーザーがまず検証 (終了画面を使用して) しなければならないためです。</p>
XXXX メインメニュー	F16	いいえ	<p>アプリケーションにとってオプションです。F16 はアクティブでなければならず、システム・メニューで表示される必要があります。これにより、システムまたはアプリケーションの 1 次メニューへの直接パスが提供されます。システムのメインメニュー以外のターゲット・メニューは明示的に述べられている必要があります。このキーは通常、アプリケーションのメインメニューへの直接パスを提供するアプリケーション内で使用できます。このキーが押されると、ユーザー入力はすべて廃棄されます。ターゲット・メニューで F3 または F12 が押されると、ユーザーは F16 が押された画面に戻ります。</p>
リストの最上部	F17	いいえ	<p>ページング可能な画面でオプションです。F17 は、最上部または先頭にページングします (位置指定フィールドの *TOP と同じ)。このキーは、位置指定フィールドが表示されていない場合にサポートすべきです。</p>
リストの最下部	F18	いいえ	<p>ページング可能な画面でオプションです。F18 は、最下部または末尾にページングします (位置指定フィールドの *BOT と同じ)。このキーは、位置指定フィールドが表示されていない場合にサポートすべきです。</p>
左	F19	はい	<p>ページング可能な横方向の領域を持つすべての画面で操作可能にすべきです。連続するビューの間に情報の共通列はありません。</p>
右	F20	はい	<p>ページング可能な横方向の領域を持つすべての画面で操作可能にすべきです。連続するビューの間に情報の共通列はありません。</p>
オプションの続き	F23	いいえ	<p>オプションです。リストのみが表示されます。F23 を使用して、リスト画面の指示域にある次のオプションのセットを表示します。最大 3 行で表示される最初のオプションのセットが完全でない場合に使用します。F23 を押すまで表示できない他のオプションが存在する場合には、オプション域の最後のオプションの後に省略符号を表示することをお勧めします。最も重要な機能が最初に表示され、あまり頻繁に使用されないオプションは F23 が押された後に表示されます。</p>

表 45. 機能キーの割り当て (続き)

名前	キー	CUA	使用法/機能
キーの続き	F24	いいえ	オプションです。F24 を使用して、より多くのアクティブな機能キーを画面の機能キー域に表示します。すべてのキー説明が 2 行の機能キー域に収まらない場合に使用します。

すべての画面のヘルプ情報の定義

i5/OS システム画面のヘルプは、カーソルが位置指定されるフィールドおよびカーソルが任意の場所に位置指定される時点に関する特定の情報、また画面の目的やユーザーがそれと対話する方法に関する一般情報で構成されます。

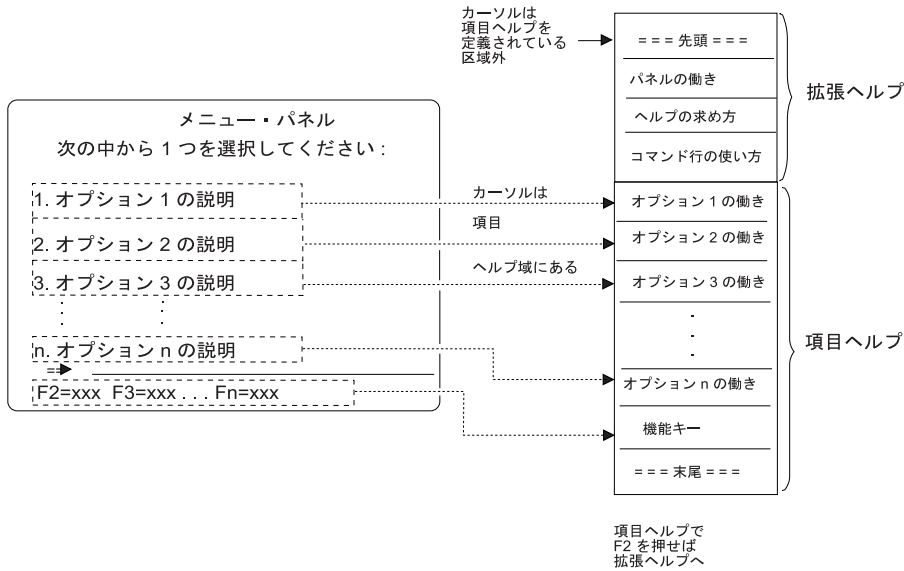
システム上のヘルプにはシステム・ヘルプ索引も含まれます。このセクションでは、画面タイプごとに、システムによって提供されるフィールド用のヘルプ・サポートのタイプについて説明します。

メニュー画面のヘルプ

以下の表および図は、メニュー・ヘルプ域ごとのヘルプ情報について説明しています。

表 46. ヘルプ域ごとのヘルプのタイプ - メニュー画面

カーソル位置	ヘルプ情報	UIM タグ
オプション番号および説明が含まれる行の上	各オプションが実行する機能	MENUI タグの HELP エlement
機能キー域の中	画面でオンにされた各機能キーによって実行される機能	KEYL および KEYI タグの HELP エlement
その他の表示位置	画面の一般ヘルプの開始 (ユーザーは画面のすべてのヘルプをページングできます。これには行全体の使用方法に関する説明が含まれます。)	PANEL タグの HELP エlement
入力のないコマンド行の上	パネルの一般ヘルプの先頭にある項目 (ユーザーは画面のすべてのヘルプをページングできます。)	PANEL タグの HELP エlement
オプション番号が入力されたコマンド行の上	各オプションが実行する機能	MENUI タグの HELP エlement



RV2W027-4

以下の表は、メニュー画面のヘルプに関する DDS の考慮事項について説明しています。

表 47. DDS の考慮事項 - メニュー画面のヘルプ

項目ヘルプ域	カバーされる領域
各メニュー・オプション	オプション番号およびオプション・テキストが入る行の 1 桁目から最右端の表示桁まで
機能キー域	機能キーを説明する最初の行の 1 桁から、機能キー域を説明する最後の行の最右端の表示桁まで

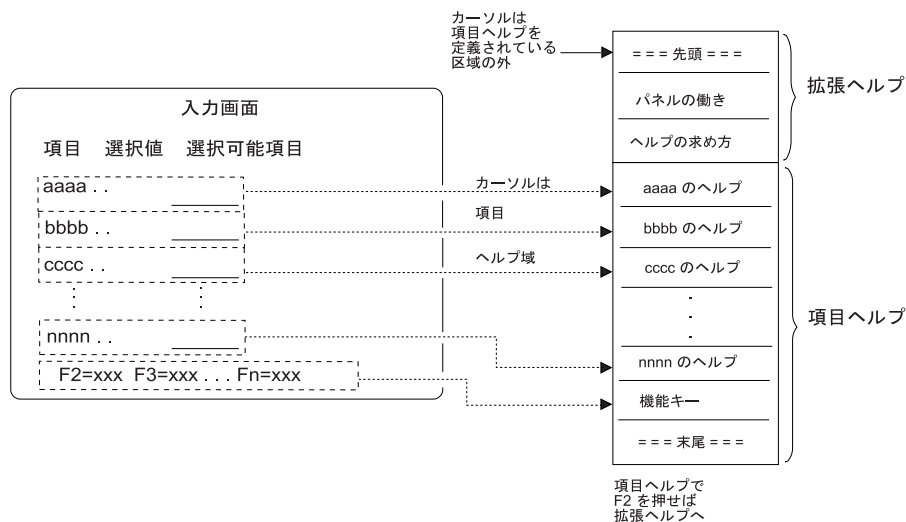
画面の残りの部分は一般ヘルプ域として定義されます。

入力画面のヘルプ

以下の表および図は、入力画面のヘルプ域ごとのヘルプ情報について説明しています。

表 48. ヘルプ域ごとのヘルプのタイプ - 入力画面

カーソル位置	ヘルプ情報	UIM タグ
入力フィールドまたはそれに関連した説明行の上 (データ入力項目)	入力フィールドのそれぞれの選択可能な項目の使用法	DATAI タグの HELP エlement
機能キー域の中	画面でオンにされた各機能キーによって実行される機能	KEYL タグの HELP エlement
その他の表示位置	画面の一般ヘルプの開始 (ユーザーは画面のすべてのヘルプをページングできます。)	PANEL タグの HELP エlement



RV2W026-6

図 139. 入力画面のヘルプ域

以下の表は、メニュー画面のヘルプに関する DDS の考慮事項について説明しています。

表 49. DDS の考慮事項 - 入力画面のヘルプ

項目ヘルプ域	カバーされる領域
各項目	項目テキストの最初の行の 1 桁から、(1) 入力フィールドまたは (2) 選択可能な項目のテキストのいずれか (どちらか大半の行を占有しているほう) の最後の行の最右端の表示桁まで (図 139 を参照)。
機能キー域	機能キーを説明する最初の行の 1 桁から、機能キー域を説明する最後の行の最右端の表示桁まで

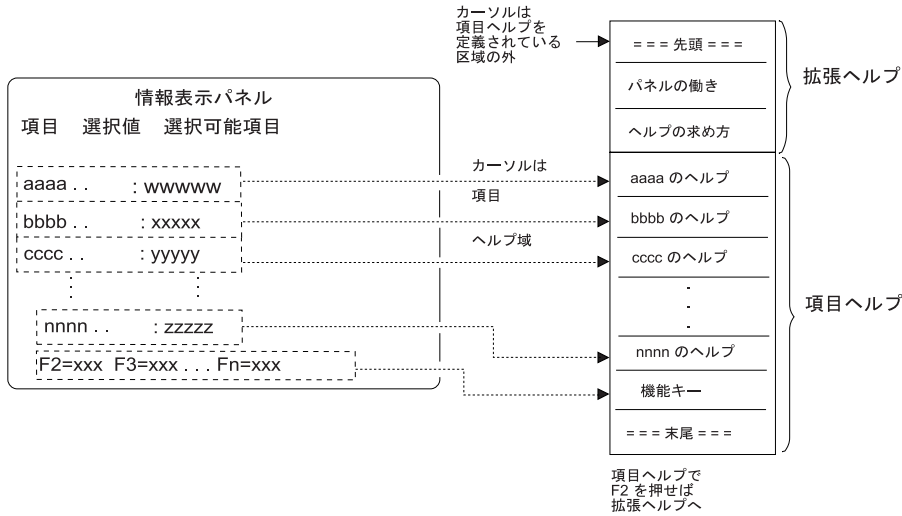
画面の残りの部分は一般ヘルプ域として定義されます。

情報画面のヘルプ

以下の表および図は、情報画面のヘルプ域ごとのヘルプ情報について説明しています。

表 50. ヘルプ域ごとのヘルプのタイプ - 情報画面

カーソル位置	ヘルプ情報
出力フィールドまたはそれに関連した説明行の中。	出力フィールドの意味。(データ出力フィールドは、説明テキストと、変数データが表示時に書き込まれる関連出力フィールドで構成される画面上の領域です。)
機能キー域の中	画面でオンにされた各機能キーによって実行される機能
その他の表示位置	画面の一般ヘルプの開始 (ユーザーは画面のすべてのヘルプをページングできません。)



RV2W043-5

図 140. 情報画面のヘルプ域

以下の表は、情報画面のヘルプに関する DDS の考慮事項について説明しています。

表 51. DDS の考慮事項 - 情報画面のヘルプ

項目ヘルプ域	カバーされる領域
データ出力項目	行の 1 桁目から最右端の表示桁まで (図 140 を参照)
機能キー域	機能キーを説明する最初の行の 1 桁から、機能キー域を説明する最後の行の最右端の表示桁まで

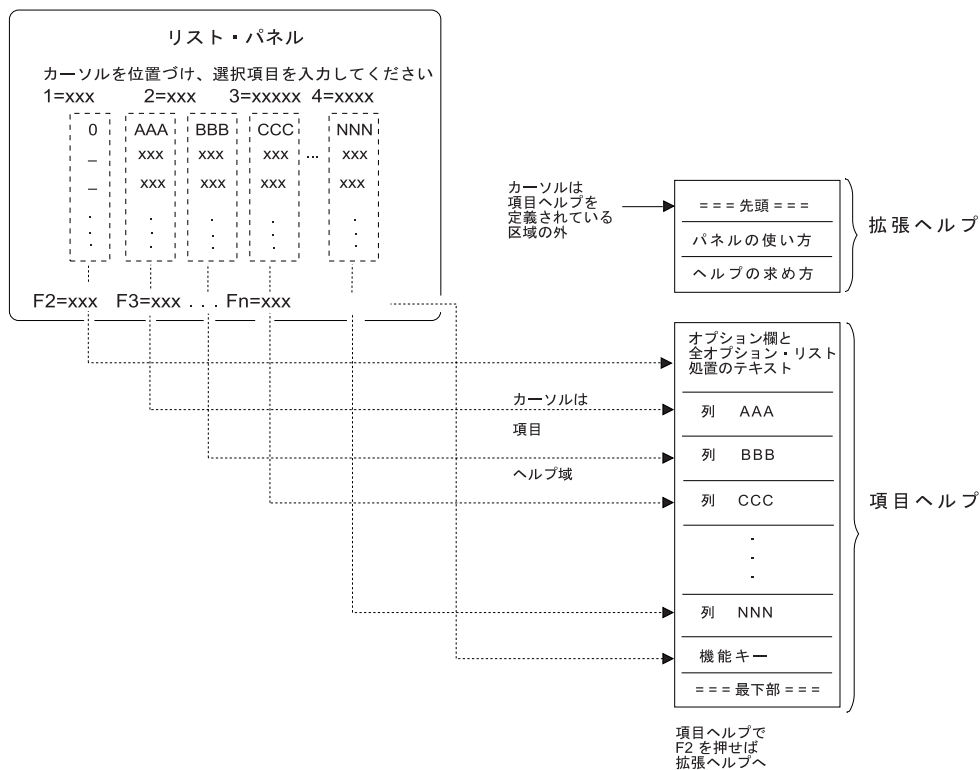
画面の残りの部分は一般ヘルプ域として定義されます。

リスト画面のヘルプ

以下の表および図は、リスト画面のヘルプ域ごとのヘルプ情報について説明しています。

表 52. ヘルプ域ごとのヘルプのタイプ - リスト画面

カーソル位置	ヘルプ情報	UIM タグ
入力フィールドまたはそれに関連した説明行の中	入力フィールドのそれぞれの選択可能な項目の使用法	LISTCOL タグの HELP エlement
特定の列	列の意味および使用法。これには、拡張処置項目が入る列の入力フィールドの使用法 (たとえば、リスト項目を作成する、またはページングせずにリスト項目に対する処置を実行するなど) が含まれます。	LISTCOL タグの HELP エlement
機能キー域の中	画面でオンにされた各機能キーによって実行される機能	KEYL タグの HELP エlement
その他の表示位置	画面の一般ヘルプの開始 (ユーザーは画面のすべてのヘルプをページングできます。)	PANEL タグの HELP エlement



RV2W025-3

図 141. リスト画面のヘルプ域

以下の表は、リスト画面のヘルプに関する DDS の考慮事項について説明しています。

表 53. DDS の考慮事項 - リスト画面のヘルプ

項目ヘルプ域	カバーされる領域
各列	列見出しの最初の行の最左端の桁から、列データの最後の行の最右端の桁まで。最左端の桁は、最も長い見出し行またはデータ列のいずれかの最左端の文字位置として定義されます。最右端の桁は、最も長い見出し行またはデータ列のいずれかの最右端の文字位置として定義されます。(図 141 を参照)。
機能キー域	機能キーを説明する最初の行の 1 桁から、機能キー域を説明する最後の行の最右端の表示桁まで

画面の残りの部分は一般ヘルプ域として定義されます。

メッセージの定義と表示

システムとの互換性を保つために、ユーザーが許可されていない選択または実際の値を入力したときはいくつでも、アプリケーションはエラー・メッセージを表示する必要があります。ユーザーが処理を継続できるように、メッセージはそれ自体可能な限り多くの情報を提供する必要があります。システム上で、メッセージは画面の下部行に現れます。

メッセージは、エラーの拡張説明 (必要な場合) および問題の救済策を提供するヘルプ情報によってサポートすべきです。一般に、ヘルプ情報は何が起こったかについて説明し、可能であれば処置が必要な場合にどうすべきかを説明します。最高の優先順位は、処置が必要なステートメントに与えてください。

ユーザーは、カーソルをメッセージ行に移動してヘルプ・キーを押すことによって、表示されているメッセージのヘルプを要求することができます。メッセージ・ヘルプは別個の画面に表示され、必要に応じて複数の画面に拡張できます。

DDS キーワード `ERRMSG` は簡単に使用できますが、これによりキーボードはロックされ、ユーザーは一度に 1 つのメッセージしか参照できません。このオプションは、メッセージが少ししかない場合に許容できます。ただし、ユーザーとメッセージとの対話を容易にするには、以下のことが目標になります。

- メッセージに応答できるようにユーザーにキーボードをアンロックさせない
- 追加メッセージ情報のためにページング可能な定様式表示画面を提供する

これを達成するには、`ERRMSG`、`ERRMSGID`、`SFLMSGID`、および `SFLMSG` キーワードを使用するのではなく、プログラム・メッセージ・キュー (`SFLMSGKEY`、`SFLMSGRCDD`、および `SFLPGMQ` キーワード) からのメッセージが含まれるサブファイルを使用する必要があります。

項目が無効であるというだけでなく、何が正しいかをユーザーに通知するメッセージを提供するには、DDS の `CHECK`、`RANGE`、`VALUES`、および `COMP` キーワードを使用してプログラムの妥当性検査を行うことをお勧めします。これらのキーワードが使用されると、メッセージは、入力されたものが無効であるものの、有効なものについての説明または指示が何も指定されていないことをユーザーに通知します。

システムでのメッセージ表示は可能な限り、以下の規則に従います。

- 検出される入力エラーによってエラー・メッセージが出され、その機能は実行されません。
- エラー・メッセージは、無効な値またはオプションが入力された画面に表示されます。エラー・メッセージは、入力されたデータおよび入力処置によって開始された別個の画面には表示されません。
- 画面で入力されたデータでエラーが検出されると、画面は以下のものを使用して再び表示されます。
 - 表示可能なエラーの最初の値。ページングできない画面の場合、値が入力されたフィールドに、エラーのある値を残した状態で画面が再び表示されます。ユーザーが値の表示ページの 1 つで値のみを入力した、ページング可能な画面の場合、画面は `Enter` キーが押された場合と同じように再び表示されます。ユーザーがページングして複数のページで値を入力した、ページング可能な画面の場合、画面はエラーが含まれる値の最初のページとともに再び表示されます。
 - エラーのある値は反転イメージで表示されます (ディスプレイ装置がこれをサポートしている場合)。

注: 強調表示された入力フィールドで反転イメージが使用されている場合、強調表示属性をオフにして (通常輝度に変更)、5250 ディスプレイ装置を使用した場合に何も表示されないという状態を避ける必要があります。

- カーソルは、エラーのある値とともに最初のフィールドに配置されます。
- キーボードはロックされません。
- 画面で入力されたデータで複数のエラーが検出された場合は以下のようになります。
 - 画面のすべてのエラーのエラー・メッセージは、ロール/ページ・キーを使用してメッセージ行で表示できます (一度に 1 つずつ)。
 - 複数のメッセージが待機中の場合、メッセージ行の最後の 3 桁には追加メッセージが存在することを示す高輝度の `"b+b"` (b はブランクを示す) が含まれます。ロール/ページ・キーによって、他のメッセージの表示が可能になります。メッセージ全体をページングするために、ユーザーはカーソルをメッセージ行に移動し、該当するロール/ページ・キーを押します。
 - 複数のメッセージが存在する場合、それらはエラーのある値が画面上に表示されるのと同じ順序で表示されます。

- ユーザーが適切な修正処置 (例、キーを押すなど) を取った場合、エラー・メッセージが除去されま
す。修正されなかったエラーのメッセージのみがメッセージ行に存在するか、または表示されるのを
待機します。
- ユーザーがメッセージ以外の項目のヘルプを要求した (ユーザーがヘルプ・キーを押した時にカーソルが
メッセージ行にない) 場合、ユーザーがヘルプ画面から戻ってもメッセージは画面上に残ったままです。

共通ユーザー・アクセス (CUA) 基本レベル・モデルの設計

CUA では、既存の単純なトランザクション型のアプリケーションについての基本モデルを定義し、そのア
プリケーションのユーザー・インターフェースを、CUA で定義されている標準インターフェース・コンポ
ネントの一部を使用するものに設計し直そうとするプログラマーに役立つようにしてあります。基本モデ
ルと他の CUA モデルの主要な違いは、基本モデルがアクション指向であり、アクションからオブジェク
トへの処理シーケンスに従うという点です。他の CUA モデルは、オブジェクトからアクションへの指向
性を持っています。

CUA 基本モデルは、既存の i5/OS インターフェースに密接に対応しています。ユーザーは、普通、メニュ
ーまたはリストから処置または作業を選択し、次にその処置の対象となるオブジェクトを選択するよう要求
されます。

基本ダイアログ処置

特定の CUA 基本ダイアログ処置は、表示域で使用可能な場合には機能キー域で表示されなければなりま
せん。表 54 は、CUA 基本ダイアログ処置を要約したもので、機能キーの割り当てと、それぞれのキーが
いつ使用されるかを示しています。その機能が使用できない場合、キーを別の機能に使用しないでくださ
い。

表 54. CUA 基本ダイアログ処置

機能	キー	使用要件
入力	Enter	すべての画面でアクティブですが、機能キー域に表示されません。
ヘルプ	F1、Help	ヘルプはすべての画面でアクティブでなければなりません。画面に 表示されません。F1 もヘルプ機能にマップされている必要があります。 F1 は機能キー域には表示されません。 CUA 画面ではヘルプ用に F1 がオンになっているべきですが、DDS では Help キーもこの機能に割り当てることができます。
終了	F3	次の例外を除き、アクティブでありすべての画面で表示される必要が あります。 <ul style="list-style-type: none"> • 確認画面。 • ユーザーが F4 を押した結果として表示されるリスト画面。 • 保管するかどうか決定する場面でユーザーが F3 を押した結果とし て表示される終了画面。 • ポップアップ・ウィンドウ
プロンプト	F4	有効な入力が認識されるものの選択可能な項目の説明域に表示できな い入力フィールドが画面に含まれる場合には常にサポートされる必要 があります。コマンド行のある画面では必須です。
再表示	F5	処置リスト画面では必須 (選択リストを除く) であり、他のリスト画 面でも推奨されます。再表示機能が適切な他の画面で使用する必要が あります。

表 54. CUA 基本ダイアログ処置 (続き)

機能	キー	使用要件
後方	前ページ、F7	ページング可能な領域を持つすべての画面で操作可能でなければなりません。機能キー域では表示されません。前ページ・キーによって、データの前のページが表示されます。 CUA 画面ではページング用に前ページ・キーがオンになっているべきですが、DDS では F7 キーもこの機能に割り当てることができます。
前方	次ページ、F8	ページング可能な領域を持つすべての画面で操作可能でなければなりません。機能キー域では表示されません。次ページ・キーによって、データの次のページが表示されます。 CUA 画面ではページング用に次ページ・キーがオンになっているべきですが、DDS では F8 キーもこの機能に割り当てることができます。
検索	F9	コマンド行のあるすべての画面で必須です。
コマンド	F9	ポップアップ・コマンド行を表示するのに使用されます。
取り消し	F12	次の例外を除き、アクティブでありすべての画面で表示される必要があります。 <ul style="list-style-type: none"> • 前の画面がすでに処理された機能のプロンプトであった場合。 • 終了機能のみが適切である画面。これは、データを保管するかどうかをユーザーがまず検証 (終了画面を使用して) しなければ終了を許可してはならないためです。
左	F19	ページング可能な横方向の領域を持つすべての画面で操作可能でなければなりません。これら連続するビューの間に情報の共通列はありません。
右	F20	ページング可能な横方向の領域を持つすべての画面で操作可能でなければなりません。これら連続するビューの間に情報の共通列はありません。

機能キー域とメッセージ行の関係

基本モデルでは、メッセージ行が他の表示域と機能キー域、およびコマンド域 (使用する場合) の間に区切り文字として置かれます。コマンド行は、機能キー域の上、メッセージ行の下にあります。

配置以外は、機能キー域のガイドラインには変更ありません。単一行の機能キーは 24 行目です。2 行の機能キーは 23 行と 24 行を占有します。メッセージ域は、機能キーのすぐ上の行です。

単一項目選択 (メニュー)

CUA 単一項目選択フィールドは、ユーザーが 1 つの項目を選択できる選択項目のリストを表示します。この画面には常に、タイトル、選択項目のリスト、および選択される項目の数を入力するためのラベル付きフィールドが含まれます。503 ページの図 142 は、画面上の CUA 単一項目選択フィールドの例を示しています。

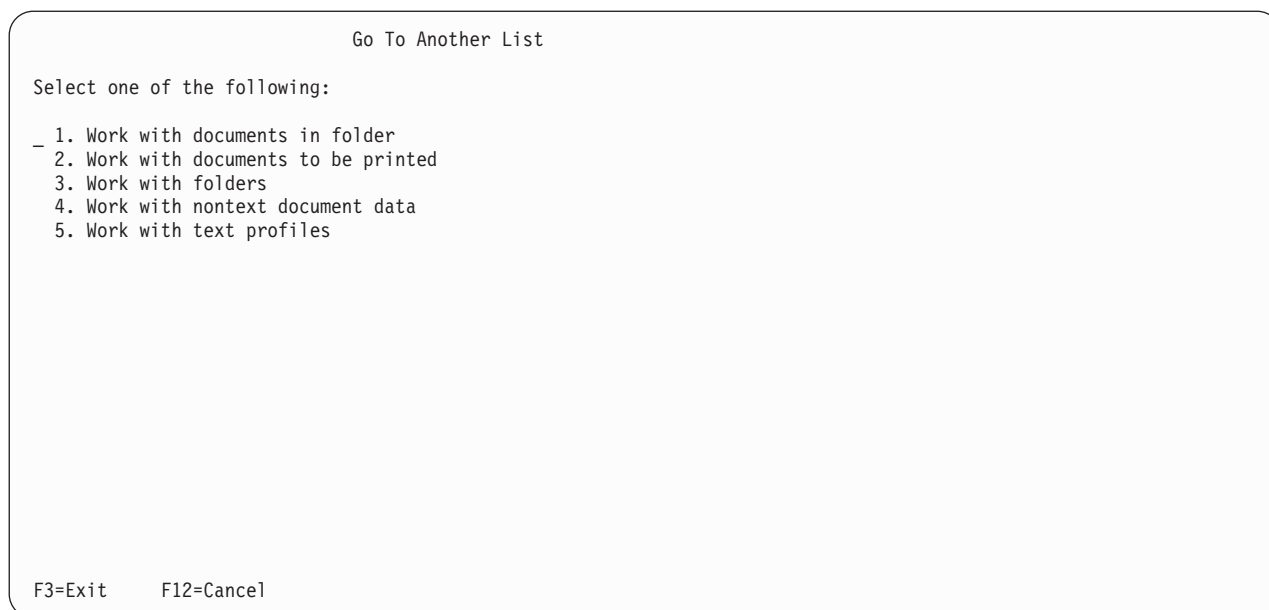


図 142. アプリケーション・メニューの例

選択項目および選択項目入力フィールド

選択項目および選択項目入力フィールドのガイドラインは以下のとおりです。

- 選択項目は指示行の 1 ブランク行後ろで開始します。
- 選択項目入力フィールドは最初の選択項目の前の 5 行目にあります。選択項目入力フィールドの長さは、選択項目数が 10 より少ない場合は 1 桁、選択項目数が 10 以上 (最大 99 まで) の場合は 2 桁になります。
- 選択項目入力フィールドは 2 桁目で始まります。
- 選択項目をリストします (1 行に 1 つ)。オプション数が 9 以下の場合には 4 桁目で始まります。番号の後にピリオドが続きますが、強調表示されません。
- 選択項目数が 9 を超える場合、選択項目入力フィールドには 2 文字必要になります (2 桁目と 3 桁目)。オプション番号 1 から 9 は 6 桁目で始まり、番号 10 以上は 5 桁目で始まります。

単一選択フィールド操作のガイドライン

単一選択フィールドは 2 つの対話方法を許可します。ユーザーは、選択項目入力フィールドに選択項目数を入力して Enter キーを押すか、希望する選択項目に直接カーソルを移動して Enter キーを押すことができます。

以下の規則を使用します。

- カーソルが選択項目入力フィールドにあるか、または 1 桁右にある場合 (3 桁目あるいは 4 桁目)、選択項目入力フィールドの選択項目を処理します。
- カーソルが選択項目入力フィールドの中またはすぐ右になく、および選択項目入力フィールドに空白が含まれる場合、カーソルと同じ行にある選択項目を処理します。

- カーソルが選択項目入力フィールドの中またはすぐ右になく、選択項目入力フィールドがブランクでない場合 (値が入力されている)、およびフィールド内の値が選択された項目と同じでない場合、カーソルが指している項目と選択項目入力フィールドの値によって示される項目のいずれかを選択するようユーザーに求めるメッセージが表示されます。

入力画面

入力画面を使用して、ユーザーは入力フィールドに入力することができます。これは通常、アクション要求に関連したシステム・パラメーターおよびオプションを示すのに使用されます。図 143 は入力画面の例を示しています。

```

Merge Options

Type choices, press Enter.

Job queue . . . . . - 1. Placed on job queue
                   - 2. Not on job queue
Completion message . . . . . - 1. Message sent
                              - 2. Message not sent
Job description . . . . . _____ + Name
  Library . . . . . _____ Name, *LIBL
Adjust/paginate option . . . . . - 1. Do not adjust
                                   2. Line ending only
                                   3. Line and page ending
Report line format . . . . . - 1. Multiple lines
                              - 2. No multiple lines
Footnotes . . . . . - 1. Collected
                   - 2. Not collected

F3=Exit  F4=Prompt  F12=Cancel

```

図 143. 入力画面

入力フィールド

CUA 入力フィールドは 2 つの形式を取ります。最初の形式は、名前、記述テキスト、またはアドレスのようにユーザー提供の値を要求する入力フィールドです。しばしばこれらのフィールドは、変数の長さおよび内容の値のリストから値を受け入れることができます。こうした有効な選択項目のリストが存在する場合、F4 をサポートして、ユーザーがそのリストを要求してそれから単に選択できるようにします。

記述テキスト (リストは F4) を使用することは、F4 をサポートするフィールドを識別するための望ましい方法です。その代わりに CUA では、フィールドが F4 をサポートする場合、正符号 (+) を指定して入力フィールドを続けます (属性バイトには 1 つの中間ブランクを使用)。

```

ファイル . . . . . _____ + 名前

```

入力フィールドの 2 番目のタイプは、固定された選択項目のセットからの選択をサポートします。CUA では、これを選択フィールドと呼びます。選択項目を 1 つしか選択できない場合、各選択項目に番号を付けます。以下に例を示します。

```

書体 . . . . . - 1. エリート (12 ピッチ)
                - 2. クーリエ (10 ピッチ)
                - 3. エッセイ・スタンダード

```


リスト画面

情報画面と同じように、CUA バージョンのサンプル・リスト画面は、ページング情報の表示が主に異なっています。図 145 は、リスト画面の例を示しています。

CUA は More という語の後にコロンのおよびページング記号を付けて、追加情報が可視域の外側に存在すること、およびその情報を参照するためにページングする方向を示します。

ユーザーが 4 方向すべて（後方、前方、左、および右）でページング可能な場合、4 つのページング記号すべてのためにスペースを予約してください。ユーザーが 2 方向にしかページングできない場合には、2 つのためにスペースを予約します。ユーザーがその記号の表す方向にページングできない場合には、記号は表示されず、代わりにブランクが表示されます。

4 つのページング記号は次のとおりです。

- < 左に情報があることを示します。
- 後方に情報があることを示します。
- + 前方に情報があることを示します。
- > 右に情報があることを示します。

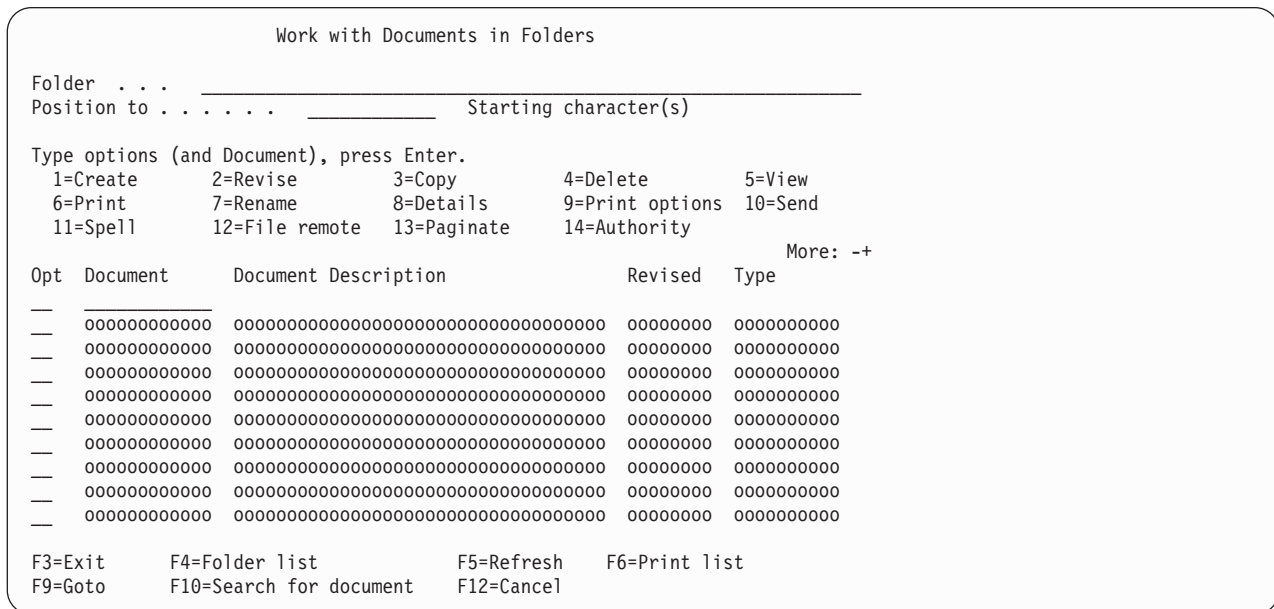


図 145. リスト画面の例

CUA に従って、代わりにテキスト・ページング標識として More: - + ではなく以下のページング標識を使用できます。

- 先頭 ユーザーが情報の始まりを表示する際にページングされる領域の上
- 続く ユーザーが後方にページング可能な場合にページングされる領域の上
- 終わり ユーザーが情報の終わりを表示する際にページングされる領域の下
- 続く ユーザーが前方にページング可能な場合にページングされる領域の下

注: この技法は、ページングされる領域の上下に余分の行を必要とします。

サブファイルの使用時に、CUA 定義のページング情報を使用することはできません。アプリケーション・プログラム自体がページングを処理する必要があります。詳しくは、478 ページの『リスト・フィールド』を参照してください。

一定量で情報をページングする場合、使用されるページング方法はカーソル独立型でなければなりません。

ヘルプ情報

CUA ヘルプ情報は、機能キーを使用して多くの定義された機能をすべてのヘルプ画面で使用可能にすることによって、i5/OS ヘルプ情報に対応しています。機能キーとそれぞれの機能は次のとおりです。

F1=ヘルプのヘルプ

F2=全般ヘルプ

F3=終了

F5=チュートリアル

F9=キー・ヘルプ

F11=ヘルプ索引

F14=チュートリアル

これらの機能は、ヘルプ画面から機能キーを使用する DDS では提供できません。ヘルプ情報に関する DDS 機能については、495 ページの『すべての画面のヘルプ情報の定義』を参照してください。

第 6 部 付録

付録 A. UIM パネル・グループ定義言語

この付録では、UIM パネル・グループ定義言語について説明します。パネル・グループ定義の言語構文および構造の性質に関するいくつかの導入部に続いて、言語の実際のステートメントである**タグ**が定義されています。この章のタグ定義はアルファベット順に編成されています。

構文図のサンプルを以下に示します。



TAGNAME はタグの名前を指定します。このサンプル・タグには、2 つの属性 VATTR (値属性) および KATTR (キーワード属性) があります。この例の 2 番目の属性は、オプション属性があることを示す代替分岐で表示されています。この属性は、3 つの値 DEFAULT、VALUE1、および VALUE2 のいずれかを持つことができます。

DEFAULT は、KATTR 属性の値を指定しない場合に使用される値です。デフォルト値がある場合、DEFAULT が KATTR 属性名の上の行に表示されるのと同じように、それは属性名の上のグループ選択項目に表示されます。DEFAULT 値を使用する代わりに、VALUE1 または VALUE2 のいずれかを選択することもできます。これらの値は、構文図で示されているように入力されます。

value は、その属性に指定されたとおりにユーザーが入力する値です。属性 value は、それに A から Z または 0 から 9 以外の文字が含まれる場合、アポストロフィ (') で囲まなければならない、1 つのソース行に含める必要があります。アポストロフィで囲まれた値にアポストロフィを入力するには、アポストロフィを 2 つ入力します。

tag-content は、ユーザーに対して表示または印刷可能な、そのタグに関連したテキストです。これは、各国語の翻訳を使用できるように設計されています。

タグの終わりおよびタグの内容の前にあるピリオドは、**マークアップ/内容区切り文字**として知られています。すべてのタグ・マークアップには、この区切り文字が必要です。

パネル・グループのソースは、小文字、大文字、または大文字小文字の混合で入力することができます。UIM はコンパイル処理時に、タグ名、属性名、および属性値を適宜、大文字に変換します。たとえば、サンプル・タグ TAGNAME の以下のような使用は同等の仕様であると見なされます。

```
:TAGNAME VATTR=VALUE.
:tagname vattr=value.
:Tagname Vattr=Value.
```

構文図で指定された順序で、タグ属性を指定する必要はありません。たとえば、サンプル・タグ TAGNAME の以下のような使用は同等の仕様であると見なされます。

```
:tagname vattr=value kattr=value1.
:tagname kattr=value1 vattr=value.
```

ときおり、タグの属性を別のソース行に継続する必要が生じます。ただし、すべての属性を継続できるわけではありません。継続できるのは、以下の表の属性のみです。

タグ言語

表 55. 継続可能なタグ属性

属性	タグ
ACTION	KEYI
ACTION	MENUI
ACTION	PDFLDC
COLHEAD	LISTCOL
COLS	LISTVIEW
CONDS	TT
EMPHASIS	LISTDEF
ENTER	LISTACT
ENTER	PANEL
EXPR	COND
EXTENTER	LISTACT
EXTPROMPT	LISTACT
LINKWHEN	LINK
NOGET	VARRCD
NOPUT	VARRCD
PERFORM	LINK
PROMPT	LISTACT
PROTECT	LISTDEF
RANGE	CHECK
REL	CHECK
ROOTS	ISCH
TOPICS	ISCHSUBT
UNLESS _n	LINK
VALUES	CHECK
VALUES	TTROW
VAR	LISTDEF
VAR	VARRCD

属性を継続するには、以下の例にあるように次のソース行で属性を繰り返します。

```
:TAGNAME
    VATTR='value1 value2 value3'
    VATTR='value4 value5 value6'
    VATTR='value7 value8 value9'.
tag-content
```

任意のタグで最大 50 の属性名 (反復したそれぞれの属性名を含む) を指定できます。

段落としてフォーマットされるタグの内容

ピリオド (.), 疑問符 (?), 感嘆符 (!) は、文末文字です。各国語の要件によっては、パネル・グループまたはメニュー・オブジェクトの作成時に、UIM ソース行で終わるそれぞれの文末文字の後に 1 つ以上のブランクが置かれます。ブランクの数は 1 から 6 であり、メッセージ・ファイル QCPFMMSG にある CPI6AB9 メッセージのメッセージ・テキストで指定されます。

以下の例では、CPI6AB9 メッセージが機能する方法について説明します。

```
:p.This is the first sentence.
This is the second!   This is the third.
```

この例では、:p. はタグ名です。:p. タグは属性を必要としません。各国語の要件に従って、2 番目の文と 3 番目の文の間にはスペースが 3 つあります。CPI6AB9 メッセージが xx に設定されると、この段落は、コンパイル後に以下のように UIM に表示されます。

This is the first sentence. This is the second! This is the third.

最初の文と 2 番目の文の間にはスペースが 2 つあります。2 番目の文と 3 番目の文の間にはスペースが 3 つあります。

CPI6AB9 メッセージは、ソース行の後に続くスペースの量に影響するため、CPI6AB9 メッセージが xxx に変更されると、前の例では各文の間のスペースは 3 つになります。

注: CPI6AB9 メッセージは、ユーザーの国または地域に合わせてすでに適切な数の x に翻訳されているはずですが。

パネル域

パネル域の 5 つのタイプは、メニュー、情報、データ、リスト、およびテキスト域です。混合パネルは、同じパネル内で複数の領域を使用して構成されます。

混合パネルに関連した制約事項は、表 56 にリストされています。

表 56. 混合パネルに関連した制約事項

	メニュー	情報	データ	非 処置 リスト	処置 リスト	テキスト
メニュー	いいえ	はい	はい	はい	いいえ	いいえ
情報	はい	はい	はい	はい	はい	いいえ
データ	はい	はい	はい	はい	はい	いいえ
非 処置 リスト	はい	はい	はい	はい	はい	いいえ
処置 リスト	いいえ	はい	はい	はい	いいえ	いいえ
テキスト	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ

これらの領域タイプのすべてはスクロールできます。スクロール、さらに適切なスクロール・ロケーション情報の表示は、メニュー、情報、データ、およびリスト域で UIM によって処理されます。テキスト域のスクロールは、テキスト域のユーザー出口プログラムによって処理されます。

パネル

パネルは、1 つ以上のパネル域で構成され、それぞれには固有のレイアウトと機能があります。固有のパネルは通常、そのアプリケーション域の性質 (情報パネル、メニュー・パネル、リスト・パネル、およびテキスト・パネルなど) に従って分類されます。

パネル・グループ・オブジェクト

パネル・グループは、ソフトウェア製品、コマンド、または関連したコマンドの集合やその他のサービスに対応可能な、パネルの論理グループです。これには、任意のタイプのパネルを含めることができます。パネル・グループは、パネル、変数、リスト定義、およびヘルプ・モジュールの名前のコンテキストを提供します。

パネルに関するヘルプ

パネルに関するヘルプは、パネル定義中で参照されるヘルプ・モジュールで構成されます。パネルの全般ヘルプは、パネル、メニュー・バー、すべての領域の項目 (定義された順序通りに)、および機能キー (定義された順序通りに) のヘルプを連結したものです。

パネル・グループの編成

以下のリストでは、パネル・グループおよびメニューの UIM ソース内でのタグの必要な順序を示します。

パネル・グループのアウトライン: パネル・グループ (PNLGRP)

- プロローグ・セクション
 - 著作権 (COPYR)
 - インポート (IMPORT)
 - クラス定義 (CLASS)
 - 変数定義 (VAR)
 - 変数レコード定義 (VARRCD)
 - リスト定義 (LISTDEF)
 - 条件定義 (COND)
 - 真理値表定義 (TT)
 - メニュー・バー定義 (MBAR)
 - キー・リスト定義 (KEYL)
- 本体セクション
 - 表示パネル定義 (PANEL)³
 - 領域定義 (MENU、INFO、DATA、LIST、TEXT)
 - コマンド行およびオプション行定義 (CMDLINE、OPTLINE)
 - 印刷ヘッド・パネル定義 (PRTHEAD)³
 - 領域定義 (INFO、DATA)
 - トレーラー・メッセージ印刷 (PRTRAIL)
 - 印刷パネル定義 (PRTPNL)³
 - 領域定義 (INFO、DATA、LIST)
 - ヘルプ・モジュール定義 (HELP)³

メニュー作成 (CRTMNU) コマンドを使用して作成された UIM メニューは、パネル・グループの特殊な書式です。これらのメニューは、以下のパネル・グループ・タグに制限されます。

UIM メニュー・アウトライン: パネル・グループ (PNLGRP)

- プロローグ・セクション
 - 著作権 (COPYR)
 - インポート (IMPORT)
 - 変数定義 (VAR)

3. PANEL、PRTHEAD、PRTPNL、および HELP タグは任意の順序で使用できます。

4. PANEL タグおよび HELP タグは任意の順序で使用できます。↓

- 条件定義 (COND)
- 真理値表定義 (TT)
- メニュー・バー (MBAR)
- キー・リスト定義 (KEYL)
- 本体セクション
 - 表示パネル定義 (PANEL)⁴
 - 領域定義 (MENU、INFO、DATA)
 - コマンド行およびオプション行定義 (CMDLINE、OPTLINE)
 - ヘルプ・モジュール定義 (HELP)⁴

名前構文

名前を付けることのできるパネル・グループ内には、タグを使用して識別されるいくつかの要素があり、そのパネル・グループ内の他の要素 (クラス、ダイアログ変数、リスト、および条件など) から参照することができます。アプリケーション・プログラムもまた、アプリケーション・プログラミング・インターフェース (API) の呼び出し時に、名前をパラメータとして UIM に渡すことによって、名前が付けられた要素を参照できます。これらの要素は、それらを定義するタグ上に NAME 属性を持っています。これらの名前は、最大 10 文字までの長さになります。A ~ Z および 0 ~ 9 の文字だけを含めることができます。名前の最初の文字は A ~ Z の文字でなければなりません。

HELP タグによって定義されたヘルプ・モジュールの名前は、最大 32 文字までの長さになり、他の要素と同じネーム・スペースで定義されます。この結果、ヘルプ・モジュールと別の要素はパネル・グループ内で同じ名前を持つことはできなくなります。名前には、文字 A ~ Z、0 ~ 9、スラッシュ (/)、および下線 (_) を含めることができます。名前にスラッシュまたは下線が含まれる場合は、それをアポストロフィで囲まなければなりません。ヘルプ名の最初の文字は、A ~ Z、スラッシュ、または下線ではなければなりません。

名前はすべて大文字の形式でパネル・グループ・オブジェクトに保管されるため、英字の小文字 (a から z) は大文字 (A から Z) に変換されます。要素の名前は、アプリケーション・プログラム・インターフェースを介して UIM に渡される時、大文字で渡されなければなりません。

これらの名前は、パネル・グループ内にとどまり同じネーム・スペースに常駐するため、ダイアログ変数と UIM リストは同じ名前を持つことはできません。

パネル・グループ・ソースにあるオブジェクト名を参照しなければならない場合があります。これらの名前は i5/OS オブジェクト名の規則に従う必要があります。名前に A ~ Z および 0 ~ 9 以外の文字が含まれる場合、アポストロフィで囲む必要があります。

シンボル

シンボルは、パネル・グループのコンパイル時に他の何かに置き換えることのできる名前です。すべてのシンボルはアンパーサンド (&) を使用し、その後にシンボル名、それに続けてピリオドを付けます。シンボルは、タグのピリオドに続くテキストになければなりません。

シンボルは、大文字または小文字のいずれかで入力できます。

以下のシンボルは、UIM によって定義されています。

タグ言語

& シンボルのトリガー処理をせずにアンパーサンド (&) を作成します。このシンボルは、タグのテキストでアンパーサンドが必要な場合に使用されます。このシンボルを使用しないと、UIM はシンボル名をその実際の値で置き換えます。

▶▶&.

&colon.

タグの処理をトリガーせずにコロン (:) を作成します。このシンボルは、タグのテキストでコロンが必要な場合に使用されます。

▶▶&colon.

&cont. 次のソース行と現在のソース行を強制的に連結します。連結が行われるのは、このシンボルがソース行の最後の文字である場合のみです。&cont. シンボルのための可視記号は作成されません。

&cont. シンボルを使用して 2 バイト文字セット (DBCS) 文字の 2 つの部分で継続する場合、最初の行の終わりのシフトイン文字と 2 行目の開始シフトアウト文字は両方とも廃棄されます。

&cont. シンボルは、図形、行、または例に有用です。このシンボルは、CIT、HP*n*、または LINK タグにより結果のソース行がソース・レコードの長さよりも長くなった場合に使用されます。

&cont. シンボルの使用法の例は次の通りです。

```
:LINES.  
Text to be continued &cont.  
:LINK perform='dsp help hyper'  
linkwhen='chkusrcls("PGMR")'.&cont.  
reference phrase&cont.  
:ELINK.&cont.  
more text on line.  
:ELINES.  
▶▶&cont.<img alt="arrow" data-bbox="850 500 883 513"/>
```

| &msg(msgid,msgf,lib,NOSUB).

| コンパイラーは、ライブラリー LIB のメッセージ・ファイル MSGF からメッセージ MSGID の
| 第 1 レベル・メッセージ・テキストを抜き出し、このシンボルの代わりにそのテキストで置き換
| えます。値 *LIBL および *CURLIB は、単一引用符を使用せずにライブラリーに指定できます。
| メッセージ・テキストは、置換変数を使用せずに、パネル・グループ・ソース・ファイルの
| CCSID で取得されます。

| メッセージ・ファイルは、デフォルトとして PNLGRP タグの SUBMSGF 属性で指定されたメッセ
| ージ・ファイルを使用します。SUBMSGF が指定されていない場合、メッセージ・ファイルはシンボ
| ルで指定される必要があります。

| NOSUB オプションを使用すると、コンパイラーは抜き出されたメッセージにある置換変数につ
| いて、ブランクの置き換えをしなくなります。

| ▶▶&msg(msgid
| |,msgf
| |,lib
| |,NOSUB
|).

| &period.

テキストでピリオド (.) を作成します。このシンボルは、制御ワードのトリガー処理を行わないテキストの 1 桁目にピリオドが必要な場合に使用されます。

▶▶&period.

&slr. テキストで右スラッシュ (/) を作成します。このシンボルは、タグ・ソースの 1 桁目に右スラッ

シユが必要な場合に使用されます。このシンボルを使用しないと、ソース・ファイルがバッチ・ジョブの入カストリームの一部である場合、ソース・ファイルの 1 桁目の右スラッシュによって問題が起こる可能性があります。

▶▶—&slr.—▶▶

注釈

タグによって実行されるマークアップに加えて、注釈をパネル・グループのソースに挿入することもできます。一般的に、注釈は、ファイルの内容を説明したり、構造、形式、日付、作成者などに関する情報を提供するのに使用されます。

注釈は、ピリオドで開始し、それに続けてアスタリスク (*) を使用します。これは、ソース・レコードの 1 桁目と 2 桁目で使用しなければなりません。注釈は、コンパイラーによって無視されますが、ソース・リストにはリストされます。これは、パネル・グループ・オブジェクトに保存されません。注釈の使用例は以下のとおりです。

```
.*****
.* panel group      xxx.xxx
.* author          X. X. XXXXXXX
.* change history  9/18/92 (xxx) Created
.*****
:pnlgrp ...
:
:epnlgrp.
```

埋め込み

他のファイルからのソースを、パネル・グループのコンパイル時にパネル・グループに組み込むことができます。これにより、複数のソース・ファイル・メンバーからパネル・グループまたはメニューを作成できるようになります。埋め込みは、.IM 制御ワードを使用して行われます。埋め込みの使用例は以下のとおりです。

```
.im member1
```

パネル・グループの作成 (CRTPNLGRP) またはメニュー作成 (CRTMNU) コマンド上の組み込みファイル (INCFIL) パラメーターで指定されたファイルのメンバー (member1) がオープンされ、元のソース・ファイルにあるかのように処理されます。埋め込みは、最大 10 の深さまでネストすることができます。10 より深くネストしようとするとうエラーが発生し、パネル・グループまたはメニュー・オブジェクトは作成されません。

DBCS 図形リテラル

DBCS 図形リテラルは UIM によってサポートされ、UIM は DBCS 図形データのリテラル文字のストリングの処理をサポートします。リテラルの構文は以下のとおりです。

▶▶—"—*shift-out*—⁽¹⁾—*D*—⁽²⁾—*shift-in*—"—*g*—▶▶

注:

- 1 *Shift-out* は X'0E' である
- 2 *D* は 1 つの DBCS 文字である
- 3 *Shift-in* は X'0F' である

タグ言語

値は、特殊文字を文字を含んでいるため、アポストロフィで囲まれなければなりません。

DBCS 図形フィールドと比較される UIM タグでコーディングされたリテラル文字のすべてのストリングは、DBCS 図形リテラルの形式でなければなりません。シフトアウト文字およびシフトイン文字は、比較が実行される前に、グラフィック・ストリングから除去されます。

リテラルを入力可能な図形フィールドと比較する場合には注意が必要です。非グラフィック DBCS 装置では、入力では使用可能なバイト数が 2 バイト少ないことを理解しておく必要があります。結果として、DBCS 図形変数の入力フィールドの最大長よりも 2 バイト小さいリテラルを常に使用しなければなりません。

DBCS 図形文字をタグのピリオドの後に使用する場合 (変換リスト項目 (TI) タグなど)、図形リテラルを使用しないでください。使用できるのは、上の図において引用符と引用符の間にあるもの (引用符を除く) すべてです。

16 進リテラル

16 進データは、文字ストリング・リテラルが許可されている場合にはいつでも属性に入力できます。16 進リテラルの構文は以下のとおりです。



注:

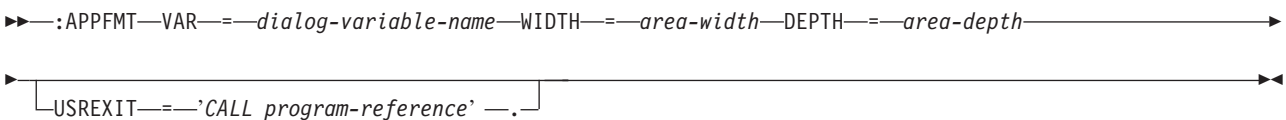
1 xx は、16 進数字 (0 から 9、a から f、または A から F) の対です。

次の例は、変換項目の 16 進値をコーディングする方法を示しています。

```
:TI value="'01"x'.*YES  
:TI value="'02"x'.*NO
```

値は、特殊文字を文字を含んでいるため、アポストロフィで囲まれなければなりません。

APPFMT (アプリケーション・フォーマット済み領域)



アプリケーション・フォーマット済み領域 (APPFMT) タグは、オプションのタグであり、アプリケーションによってフォーマットされるメニュー域の右上隅の領域を定義します。各メニュー域で指定できるのは 1 つの APPFMT タグのみです。これは、メニュー域 (MENU) および最上部指示 (TOPINST) タグの直後に置かなければなりません。

アプリケーション・フォーマット済み領域は、メニュー域の一部であり、メニュー域の右端からパネルの右端の 1 文字手前まで表示されます。幅は、このタグの WIDTH 属性によって決定されます。アプリケーション・フォーマット済み領域は、メニュー域の最上部指示行に続く空白行から、このタグの DEPTH 属性で定義された行数にまで及びます。メニュー項目の説明とアプリケーション・フォーマット済み領域の間で、2 バイト区切り記号が保持されます。

アプリケーション・フォーマット済み領域の縦の長さがメニュー域の縦の長さより小さい場合、アプリケーション・フォーマット済み領域の下の領域からメニュー域の右側までがブランクになります。アプリケーション・フォーマット済み領域の縦の長さは、メニュー域の縦の長さより大きくすることはできません。

メニュー域がスクロール可能な場合、スクロール標識は、メニュー域とアプリケーション・フォーマット済み領域の間にある区切り記号の左側に表示されます。アプリケーション・フォーマット済み領域は、メニュー域をスクロールできる場合でも、スクロールすることはできません。

アプリケーション・フォーマット済みデータは、『アプリケーション・フォーマット済みデータ』で説明されています。

パネルの全般ヘルプは、カーソルがアプリケーション・フォーマット済み領域内にある間にヘルプ・キーが押されると、表示されます。

必須属性

VAR=*dialog-variable-name*

アプリケーション・フォーマット済み領域が含まれるダイアログ変数の名前。クラス定義 (CLASS) タグを CHAR *x* の BASETYPE とともに使用して、ダイアログ変数を定義します。ここで、*x* は、領域の幅と縦の長さとの積に等しくなります。

このタグの USREXIT 属性で指定された、アプリケーション・フォーマット済みデータ用の出口プログラムを使用して、パネルを表示するたびにこのダイアログ変数を更新することができます。UIM は、『アプリケーション・フォーマット済みデータ』で説明されているように、ダイアログ変数の値をアプリケーション・フォーマット済みデータとして使用します。

WIDTH=*area-width*

メニュー域に関連したアプリケーション・フォーマット済み領域用の幅 (文字数)。指定した幅は、1 から 40 の範囲の整数であるか、パネルの幅から 17 を引いた値 (と 40 の小さいほう) 以下の整数でなければなりません。

DEPTH=*area-depth*

アプリケーション・フォーマット済み領域の縦の長さ (行数)。この縦の長さは、メニューの本体で使用可能な行数を超えることができません。メニューの本体は、最上部指示行と最下部指示行または最下部区切り記号で囲まれた領域です (境界を含まない)。

オプションの属性

USREXIT='*CALL program-reference*'

アプリケーション・フォーマット済みデータ用の出口プログラムであり、パネルを表示するたびに、アプリケーション・フォーマット済みデータを含むダイアログ変数の値を更新するために呼び出されます。

CALL ダイアログ・コマンドの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

UIM とアプリケーション・フォーマット済みデータの出口プログラムとの間のインターフェースについて詳しくは、iSeries Information Center のAPI のトピックを参照してください。

アプリケーション・フォーマット済みデータ

アプリケーション・フォーマット済みデータが含まれるダイアログ変数は、このタグの VAR 属性で指定されます。この変数の内容は、UIM によって文字の *d* × *w* 配列と見なされます。ここで、*d* は、アプリケ

APPFMT タグ

ーション・フォーマット済み領域の縦の長さ、*w* は幅を表します。配列の各行は、アプリケーション・フォーマット済み領域の 1 つの行として表示されます。

データ内の文字はどれでも、強調表示クラスのための選択文字にすることができます。アプリケーション・フォーマット済み領域で使用できるのは、出力およびテキスト・クラスのみです。入力フィールドは使用できません。クラス選択は、後続の選択文字から別のクラス選択文字または行の終わり（どちらか先に起こるほう）まで適用されます。UIM は、領域の各行の始めで、強調表示クラスを通常のテキストに設定します。

強調表示用の選択文字の範囲は、X'01' から X'06' までです。以下の強調表示用の選択文字が UIM によって認識されます。

X'01'	通常の変数出力
X'02'	強調されない変数出力
X'03'	強調された変数出力
X'04'	通常テキスト
X'05'	強調されないテキスト
X'06'	強調されたテキスト

UIM は、それぞれのクラス選択文字をクラスの適切な表示属性に置き換えます。X'01' から X'06' の範囲にある他の文字値、X'10' から X'3F'、さらに X'FF' は、X'1F' に変換され、画面上で反転表示ボックスで表示されます。文字 X'00' (ヌル)、X'0E' (2 バイトのシフトアウト)、X'0F' (2 バイトのシフトイン)、および X'40' から X'FE' (通常表示可能文字) は、未変更のまま画面に渡されます。

UIM は、通常テキスト属性をアプリケーション・フォーマット済み領域の各行の前に、フィールド終了属性をアプリケーション・フォーマット済み領域の各行の後に挿入します。これにより、強調表示属性に関して、アプリケーション・フォーマット済み領域がパネルの残りの部分から効率的に分離されます。

アプリケーション・フォーマット済みデータでは、文字セットおよびコード・ページ変換は行われません。変換が必要な場合、それはアプリケーションによって行われる必要があります。

UIM が、メニューのアプリケーション領域をフォーマットするために出口プログラムを呼び出すと、パネル・グループ (PNLGRP) タグの BIDI 属性、および表示装置のコード・ページ番号を識別する値が渡されます。

アプリケーション・フォーマット済み領域に使用される CLASS タグの NBRSHAPE および SYMSWAP 属性は、アプリケーション・フォーマット済みデータが含まれるダイアログ変数のクラスによって決定されません。

例: アプリケーション・フォーマット済み領域

オフィスビジョン / 400

システム : SYSNAMXX

次の1つを選択して下さい。

<ol style="list-style-type: none"> 1. 予定表 2. メール 3. メッセージの送信 4. ノートの送信 5. 文書およびフォルダー 6. ワード・プロセッシング 7. 登録簿/配布リスト 8. 判断サポート 9. 管理 90. サイン・オフ 	<p>時刻 : 11:42</p> <p>9月 1992</p> <table style="font-family: monospace; border-collapse: collapse;"> <tr> <td>S</td><td>M</td><td>T</td><td>W</td><td>T</td><td>F</td><td>S</td> </tr> <tr> <td></td><td></td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td> </tr> <tr> <td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td> </tr> <tr> <td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td> </tr> <tr> <td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td> </tr> <tr> <td>27</td><td>28</td><td>29</td><td>30</td><td></td><td></td><td></td> </tr> </table>	S	M	T	W	T	F	S			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30			
S	M	T	W	T	F	S																																					
		1	2	3	4	5																																					
6	7	8	9	10	11	12																																					
13	14	15	16	17	18	19																																					
20	21	22	23	24	25	26																																					
27	28	29	30																																								

終了

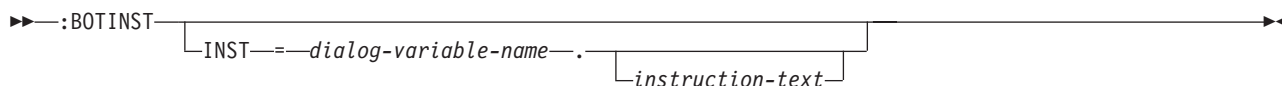
選択したオプションを延期するためには ATTN キーを押して下さい。

選択項目

—

F3= 終了 F12= 取消し F19= メッセージの表示
 (C) COPYRIGHT IBM CORP. 1985, 1992.

BOTINST (最下部指示)



最下部指示 (BOTINST) タグは、パネルの領域に最下部指示行を指定します。このタグは、領域の終了タグの直前に出現します。タグでピリオドの後に説明テキストが含まれる場合、複数の最下部指示タグを使用して、複数の指示行を表すことができます。複数のタグをコーディングした場合、異なるタグのテキストの間に空白行は表示されません。最下部指示タグで INST 属性が使用されている場合、領域ごとに使用可能な最下部指示タグは 1 つだけです。

領域に対して BOTINST タグを指定していない場合、領域に割り振られる指示行はありません。

メニューおよび情報域の場合、空白行は常に、領域の本体と指示行の間に残されます。領域がスクロール可能な場合、領域の本体と最下部指示の間に、空白行がスクロール情報用に確保されます。データの本体とリスト域に関連して、指示行がフォーマットされる方法については、544 ページの『DATA (データ表示域)』 および 603 ページの『LIST (リスト域)』の BODYSEP 属性を参照してください。

オプションの属性

INST=dialog-variable-name

表示する最下部指示テキストを含むダイアログ変数の名前。ダイアログ変数は、PANEL タグで指定された幅から 2 を引いた値以下の幅で定義する必要があります。INST 属性を使用する場合、このタグに対して instruction-text を指定することはできません。

ダイアログ変数は、クラス定義 (CLASS) タグで BASETYPE に CHAR、IGC、または BIN を指定して定義する必要があります。

ダイアログ変数のエラー状態は、テキストの強調表示の判別には使用されません。

BOTINST タグ

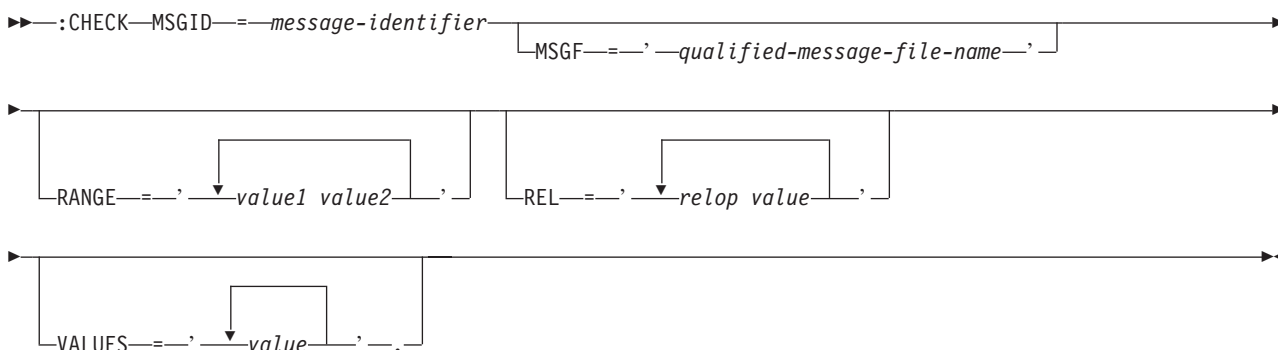
IGC 用の特殊フォーマット。(省略形 IGC は、コマンドおよびキーワードで使用され、2 バイト文字セット機能を表します。) BASETYPE が IGC であるダイアログ変数を CLASS タグ上で指定すると、UIM は特殊フォーマットを行います。ダイアログ変数値がシフトアウト文字 (X'0E') で始まる場合は、UIM は値を左に 1 バイト移動して、他の行と垂直位置合わせを行います。

オプションのテキスト

instruction-text

領域の最下部指示として表示されるテキスト。このテキストは暗黙の段落です。画面のフォーマット時に、1 表示行に収まらないテキストは、必要に応じて複数行でフォーマットされ、2 表示桁字下げされます。テキストは最大 255 文字で、テキスト反転 (RT) タグだけを含めることができます。このタグに INST 属性を指定しない限り、*instruction text* は必須です。

CHECK (妥当性検査)



妥当性検査 (CHECK) タグは、クラス定義内でそのクラスの変数に関連した妥当性検査を指定するのに使用されます。

妥当性検査は、変数値の入力時および変換リストの処理後に行われます。変換リストがこのクラスに指定されており、入力値に適用可能な場合、変換された入力値では妥当性検査は実行されません。変換リストが適用外で、エラーを示さない場合、このタグによって定義された妥当性検査は入力値に適用されます。変換リストを使用した CHECK タグの使用法については、685 ページの『TI (変換リスト項目)』および 686 ページの『TL (変換リスト)』を参照してください。

このタグの少なくとも 1 つの RANGE、REL、または VALUE 属性がコーディングされていなければなりません。複数の妥当性検査がタグでコーディングされている場合、その値は、この検査で受け入れられる検査属性の 1 つのみを通過する必要があります。複数の CHECK タグを使用する場合、値は、受け入れられるすべての検査タグを通過する必要があります。複数の属性からの操作は、論理 OR で結合されます。

必須属性

MSGID=message-identifier

妥当性検査が失敗した場合に、ユーザーに表示されるエラー・メッセージのメッセージ ID。UIM はメッセージの送信時に置換テキストを提供しないため、メッセージに置換変数を定義しないでください。

オプションの属性

MSGF='qualified-message-file-name'.

このタグの MSGID 属性で指定されたメッセージが含まれるメッセージ・ファイル。DFTMSGF 属性をパネル・グループ (PNLGRP) タグに指定しない場合は、この属性を指定する必要があります。

RANGE='value1 value2'

範囲検査は、文字タイプと数値タイプの両方に有効です。ユーザーによって入力された値が *value1* より大か等しく、*value2* より小か等しい場合、検査は通過します。この属性は複数回指定できます。

それぞれの値は、クラス定義 (CLASS) タグで指定された変数の BASETYPE と一致する数値または文字でなければならず、ダイアログ変数にすることはできません。すべての文字値は、引用符で囲まれている必要があります。このクラスの BASETYPE が TIME である場合、時間帯の値を指定しないでください。

REL='relop value'

比較検査は、数値タイプと文字タイプの両方に有効です。この属性は複数回指定できます。

それぞれの値は、CLASS タグで指定された変数の BASETYPE と一致する数値または文字でなければならず、ダイアログ変数にすることはできません。すべての文字値は、引用符で囲まれている必要があります。このクラスの BASETYPE が TIME である場合、時間帯の値を指定しないでください。

関係演算子は以下にリストされています。すべての演算子は、文字タイプと数値タイプの両方に有効です。検査は、CLASS タグの CASE および BLANKS 属性の影響を受けません。

= 等しい。ユーザーによって入力された値がこのリテラル値に等しい場合、検査は通過します。

= 等しくない。ユーザーによって入力された値がこのリテラル値に等しくない場合、検査は通過します。

> より大。ユーザーによって入力された値がこのリテラル値より大きい場合、検査は通過します。

< より小。ユーザーによって入力された値がこのリテラル値より小さい場合、検査は通過します。

>= 大きいか等しい。ユーザーによって入力された値がこのリテラル値より大きいか等しい場合、検査は通過します。

<= 小さいか等しい。ユーザーによって入力された値がこのリテラル値より小さいか等しい場合、検査は通過します。

関係演算子は、ソースのコード・ページが NOT 文字 (D) を X'5F' にマップする場合、=、≠、>、<、>=、または <= で指定することができます。そうでない場合、特殊値 *EQ、*NE、*GT、*LT、*GE、または *LE を、それぞれ関係演算子に使用することをお勧めします。

VALUES='value1 value2 ... value-n'

値検査は、文字タイプと数値タイプの両方に有効です。ユーザーによって入力された値がリスト内の値のいずれかに等しい場合、検査は通過します。文字値の検査は、CLASS タグの CASE および BLANKS 属性の影響を受けません。VALUES 属性は複数回指定でき、最大 50 個の値をこのタグに指定することができます。

それぞれの値は、CLASS タグで指定された変数の BASETYPE と一致する数値または文字でなければならず、ダイアログ変数にすることはできません。リスト内の値は空白で区切られます。すべての文字値は、引用符で囲まれている必要があります。このクラスの BASETYPE が TIME である場合、時間帯の値を指定しないでください。

CHECK タグ

例: 妥当性検査

UIM ソース

```
:CHECK RANGE='1 10' REL='= 99' MSGID=USR0234.  
:CHECK REL='< 5' REL='> 10' MSGID=USR1234.  
:CHECK VALUES='*YES' '*NO'  
MSGID=MMM0001 MSGF=USRMSGF.
```

範囲検査では、値 1 から 10、または 99 を、適用されるクラスの任意の変数に使用できます。比較検査では、5 より小さい値または 10 より大きい値を使用できます。値検査は、2 つの特殊値を受け入れます。

CIT (タイトルの引用)

```
▶▶:CIT.—┌──────────┐:ECIT.—▶▶  
          └title-text┘
```

タイトルの引用 (CIT) タグは、資料のタイトルを識別し、対応する終了タグを必要とします。このタグは情報域とヘルプ域でのみ許可されます。オンラインおよび印刷テキストの場合、タイトルの引用には下線が付いており、テキストの任意の場所に入れることができます。

CIT および ECIT タグはワード境界で指定する必要があります。ECIT タグの直後に続く 2 文字が句読記号とブランクである場合、UIM は強調属性を拡張して句読記号を含めます。これにより、句読記号とタイトルを、同じ強調を使用して表示できます。

オプションのテキスト

title-text

引用されるタイトルは必要ではありませんが、タイトルが指定されていなければ、タグは意味を持ちません。

例: タイトルの引用

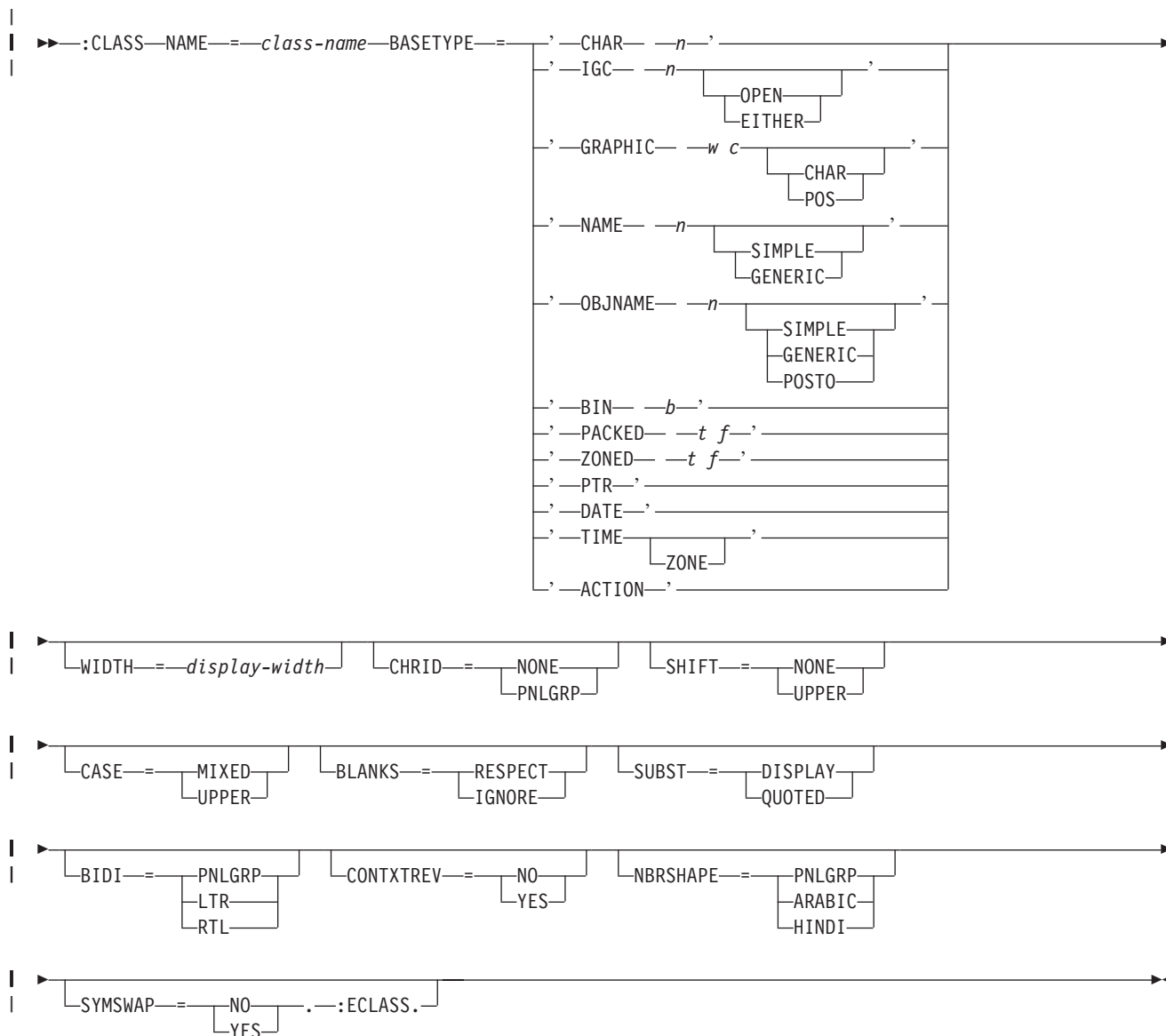
UIM ソース

```
:P.For more information about the UIM,  
see the :cit.Application Display Programming:ecit. book.
```

結果

```
For more information about the UIM,  
see the AS/400 Advanced Series Application Display Programming book.
```

CLASS (クラス定義)



クラス定義 (CLASS) タグは、変数プール内およびパネル上で変数タイプを識別します。 CLASS タグの属性は、ダイアログ変数のクラスのデータ・タイプ、およびそのクラスのすべての変数に必要な妥当性検査を定義します。また、このタグの属性は、表示画面形式との間で特定のクラスの内部ダイアログ変数をマッピングするための方式も決定します。

他のタグを CLASS タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

CLASS タグ

表 57. CLASS タグと ECLASS タグの間で許可されるタグ

タグ名	順序	ページ
TL (変換リスト)	1	686
CHECK (妥当性検査)	2	522

必須属性

NAME=*class-name*

クラスに割り当てられる名前。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

BASETYPE

クラス内の変数の基本データ・タイプ。クラスの BASETYPE は、内部ストレージの表記、およびクラスの変数の基本入出力編集特性を決定します。

CHAR *n*

このクラスの変数は固定長の文字ストリングです。バイト数 *n* は、1 から 32 767 までです。

IGC *n* [OPEN | EITHER]

このクラスの変数は、シフトアウトおよびシフトイン文字で囲まれた 2 バイト・データを含められる固定長文字ストリングです。バイト数 *n* は、4 から 32 767 までです。

OPEN は、このクラスで定義された変数に 1 バイト・データと 2 バイト・データの両方を含めることができることを示します。

EITHER は、このクラスで定義された変数に 1 バイト・データまたは 2 バイト・データのいずれかを含めることができますが、両方を含めることはできないことを示します。EITHER が指定されている場合、バイト数 *n* は偶数でなければなりません。

このタグの CASE 属性は、IGC クラスでは許可されません。IGC クラスの場合、このタグでの CHRID=PNLGRP 処理は、フィールド値に 2 バイト文字セット (DBCS) データが含まれるかどうか依存しています。

IGC の BASETYPE を使用したダイアログ変数は、変数が宣言されているパネル・グループ・オブジェクトによってパネル・グループ (PNLGRP) タグに TEXTMODE=DBCS が指定されていない限り、DBCS 使用可能ではない装置で表示することができます。これにより、同じパネル・グループ・オブジェクトおよびアプリケーションは、パネル・グループ内の事前定義テキストが SBCS のみである限り、1 バイト文字セット (SBCS) および DBCS フィールド値の両方をサポートすることができます。

GRAPHIC *w c* [CHAR | POS]

このクラスの変数に含めることができるのは、シフトアウトおよびシフトイン文字で囲まれていない DBCS 文字または UCS-2 文字のみです。DBCS 文字または UCS-2 文字の数 *w* は、1 から 16 383 (1 DBCS 文字は 2 バイトに等しい) までです。CCSID *c* には、UCS-2 エンコード・スキームを使用した CCSID のみを含めなければなりません。

CHAR は、オプションの幅の値に指定された値が、UCS2 文字の数で指定されていることを示します。オプションの幅の値が指定されていない場合、デフォルトは、*w* の 2 倍に設定されます。このパラメーターは、UCS2 CCSID が指定されている場合にのみ許可されます。

POS は、オプションの幅の値に指定された値が、表示位置の数値に指定されていることを示します。幅の属性が指定されていない場合、デフォルトの幅は、*w* に指定された値に設定されます。このパラメーターは、UCS2 CCSID が指定されている場合にのみ許可されます。

GRAPHIC ダイアログ変数の内容は、DBCS データまたは UCS-2 データのみでなければなりません。このタグで GRAPHIC が指定されており、CCSID パラメーターが指定されていない場合、フィールドには図形 DBCS データが入ります。このタグで GRAPHIC が指定されており、CCSID パラメーターが UCS-2 エンコード・スキームを使用した CCSID で指定されている場合、フィールドには UCS-2 データが入ります。どちらの場合も、アプリケーションまたは UIM のいずれかによって、シフトアウトおよびシフトイン文字が変数プールに置かれることはありません。UCS-2 データは、出力時に、指定された UCS-2 CCSID から装置の CCSID に変換されます。入力時には、装置 CCSID から指定された UCS-2 CCSID に変換されます。

CCSID パラメーターが指定されていると、いくつかの制約事項が適用されます。最初に、CCSID パラメーターを使用したすべての図形フィールドは、印刷パネルの複数の行にまたがることはできません。COND タグは、同じ UCS-2 レベル 1 エンコード・スキームの他の変数との比較だけに制限されます。UCS-2 ダイアログ変数は、CMD ダイアログ・コマンドで使用することができません。また、以下のタグも使用することはできません。

- CHECK タグ
- TI タグ

現行の図形 DBCS サポートと同じように、それぞれの UCS2 文字の長さは 2 バイトです。UCS2 CCSID から装置 CCSID への変換後に装置で表示されるデータの長さは、装置で使用可能な CCSID によって異なるため、WIDTH 属性を使用してより正確なフィールド幅を指定することができます。これにより、デフォルトのフィールド幅を迂回して、フィールド・データが切り捨てられないようにするか、フィールド長を減らして画面上の表示スペースを増やすことができます。

オプションの WIDTH 属性が CCSID パラメーターを使用した GRAPHIC basetype と一緒に指定されておらず、POS 特殊値が指定されていない場合、画面上の文字数は UCS2 文字の 2 倍になります。WIDTH 属性が CCSID パラメーターを使用した GRAPHIC basetype と一緒に指定されている場合、この値は、デフォルトのフィールド幅ではなくフィールド長に使用されます。

POS が CCSID パラメーターと一緒に指定されている場合、WIDTH 属性に入力された値は、値を表示するのに必要な表示桁の数を表します。

CHAR が CCSID パラメーターと一緒に指定されている場合、WIDTH 属性に入力された値は、画面上に表示される UCS2 文字の数を表します。CHAR や POS が指定されなかった場合にも同じことが当てはまります。

たとえば、パネル・グループに以下の行が含まれるとします。

```
:class name=example
  basetype='graphic 10 X' width=Y.
```

- X は、データが保管される UCS-2 CCSID です。Y は、このフィールドの幅です。Y が指定されなかった場合、画面上のフィールドの長さは 20 (UCS-2 文字数の 2 倍) になります。
- UCS-2 データが 1 バイト・データで構成されている場合、フィールド幅 Y を 5 個の UCS-2 文字で指定して、画面上でフィールド幅が 1 バイト文字 10 個になるようにすることができます。

CLASS タグ

- UCS-2 データが 2 バイト・データで構成されている場合、フィールド幅 Y を 11 個の UCS-2 文字 で指定して、画面上でフィールド長が 1 バイト文字 22 個になるようにすることができます。これにより、シフトアウトおよびシフトイン文字用のスペースを使用することができます。

以下の同等の定義について検討してください。

```
:class name=example1 basetype='graphic 10 13488 pos' width=20.  
:class name=example2 basetype='graphic 10 13488 char' width=10.  
:class name=example3 basetype='graphic 10 13488'.
```

この定義により、10 個の UCS2 文字が含まれるクラスが作成されます。これは、パネル上では 20 桁で表示されます。

以下に示す、同じ出力の別の定義セットを検討してください。

```
:class name=example1 basetype='graphic 10 13488 pos' width=10.  
:class name=example2 basetype='graphic 10 13488 char' width=5.  
:class name=example3 basetype='graphic 10 13488' width=5.
```

この例の定義では、10 個の UCS2 文字が含まれるクラスが作成されます。これは、パネル上では 10 表示桁で表示されます。

出力時に、指定したフィールド長よりも大きいフィールド・データは切り捨てられます。入力時に、UCS-2 フィールドに入力される文字が多すぎると、フィールドは反転表示され、入力された文字が多すぎることを示すエラーがエラー行に表示されます。入力する文字の最大数は、エラー・メッセージに表示されます。

NAME *n* [SIMPLE | GENERIC]

System/38™ 名の NAME の BASETYPE および System/38 オブジェクト名と同じ規則に従った名前 (レコード様式の名前など) を使用します。

小文字 (a から z) は、コマンドが従う同じ規則に従って、大文字 (A から Z) に変換されます。小文字から大文字への変換について詳しくは、iSeries Information Center の「CL」のトピックを参照してください。

バイト数 *n* は、1 から 255 までです。

SIMPLE は、入力された名前が単純 i5/OS 名として使用できなければならないことを示します。

GENERIC は、名前が構文的に正しい i5/OS 総称名でなければならないことを示します。

ブランクまたはその他の無効な名前値がこの BASETYPE のダイアログ変数の入力値として許可されている場合、変換リストがクラス定義内で指定されていなければなりません。

OBJNAME *n* [SIMPLE | GENERIC | POSTO]

i5/OS オブジェクト名に OBJNAME の BASETYPE を使用します。SIMPLE は、入力された名前が単一 i5/OS オブジェクト名でなければならないことを示します。

GENERIC は、名前が構文的に正しい i5/OS 総称オブジェクト名でなければならないことを示します。

POSTO は、名前が位置指定フィールドに使用されること、および構文的に正しい i5/OS 名でなくてもよいことを示します。小文字 (a から z) は、名前が二重引用符で囲まれていなければ、大文字 (A から Z) に変換されます。バイト数 *n* は、1 から 255 までです。

POSTO が指定されている場合、名前の最初と最後の文字が二重引用符であれば、最後の二重引用符は名前から除去されます。最初の文字が引用符でなければ、小文字 (a から z) はすべて大文字 (A から Z) に変換されます。POSTO が指定されている場合には、名前は有効なオブジェクト名である必要はありません。

ブランクまたはその他の無効な名前値がこの BASETYPE のダイアログ変数の入力値として許可されている場合、変換リストがクラス定義内で指定されていなければなりません。たとえば、*LIBL がライブラリー名を含む変数に許可されている値でなければならない場合、変換リストが必要です。変換リストの例については、686 ページの『TL (変換リスト)』を参照してください。

このタグの CASE 属性は、オブジェクト名クラスでは指定できません。UIM は、区切り文字 (引用符) を自動的に追加または除去し、小文字の入力値を必要に応じて大文字に変換します。

BIN *b*

この BASETYPE の変数は、符号付きおよび符号なしの 2 進数です。ビット数 *b* は、15、16、31、32、または 64 のいずれかです。

'BIN 15' および 'BIN 31' は符号付きの 2 進数です。BIN 16' および 'BIN 32' は符号なしの 2 進数です。

iSeries Information Center の「API」のトピックで説明されているように、'BIN 15' は BINARY(2) と同じものであり、'BIN 31' は BINARY(4) と同じです。

'BIN 64' は符号なしの 2 進数です。

PACKED *t f*

このクラスの変数はパック 10 進数です。桁数 *t* は、1 から 63 までです。小数点以下の桁数 *f* は、0 から *t* までです。

ZONED *t f*

このクラスの変数はゾーン 10 進数です。桁数 *t* は、1 から 63 までです。小数点以下の桁数 *f* は、0 から *t* までです。

PTR

このクラスの変数はポインターです。この BASETYPE の変数は表示したり印刷することはできません。PTR 変数が指すデータは、テキスト域で表示することができます。テキスト域について詳しくは、680 ページの『TEXT (テキスト域)』を参照してください。

DATE *n*

変数は、*cyymmdd* または *yyyymmdd* の形式で日付を表す、7 または 8 バイトのストリングです。表記は、数値 *n* によって決定されます。*n* の値 2 は、*cyymmdd* 形式 (デフォルト値) を表します。*n* の値 4 は、*yyyymmdd* 形式を表します。日付の省略形は以下のように表されます。

2 桁の年の場合:

c 世紀。ゼロは 19xx 年、および 1 は 20xx 年を表します。

yy 年

mm 月

dd 日

4 桁の年の場合:

yyyy 年

mm 月

CLASS タグ

| **dd** 日

TIME [ZONE]

| 基本 TIME クラスの値は、*hhmmss* の形式で時刻を表す 6 バイト・ストリングです。時刻の省略形は以下のように表されます。

| **hh** 時

| **mm** 分

| **ss** 秒

| オプションの ZONE 属性は、時刻値が追加の 10 バイトの時間帯値をサポートすることを示します。

| **ZZZZZZZZZZ**

| 時間帯

| オプションの ZONE 属性を使用すると、値は、*hhmmsszzzzzzzzzz* の形式で時刻を表す 16 バイト・ストリングになります。

| 時間帯は妥当性検査されず、どんな種類の時間帯変換もサポートされません。それで、時間帯はすべてブランクにしてもかまいません。CHECK または COND 式で指定された、または TI タグの VALUES 属性で指定されたリテラル TIME 値には、時間帯値を含めることはできません。時間帯は、アプリケーションによって、あるいはユーザー入力からのみ指定されます。

| さらに、以下の 2 つの特殊値がサポートされます。

| ***LCL** ローカル時間帯が使用されます。

| ***SYS** システムの時間帯が使用されます。

| たとえば、変数には、ローカル時間帯で 13 時間、59 分、および 0 秒を表す、値 135900*LCL を含めることができます。

| これらの特殊値は、値が変数プールで更新される前に解決されます。

ACTION

| この変数は、BIN 15 と同じ内部形式を使用する 2 バイトの 2 進数であり、処置リストのオプション欄、または選択リストの選択欄として使用されます。

| この BASETYPE を使用した変数が、処置リストのリスト欄 (LISTCOL) タグの VAR 属性に指定されている場合、UIM は、領域内のリスト・アクション (LISTACT) タグに指定されている OPTION 属性の最大値に基づいて、入力フィールドの幅を決定します。

| 表 58. BASETYPE ごとの属性要約

BASETYPE	デフォルトの WIDTH	デフォルトの属性	最小 WIDTH 属性	最大 WIDTH 属性	デフォルトの SHIFT 属性	デフォルトの CASE 属性	デフォルトの BLANKS 属性
BIN 15	6		1	255	UPPER	UPPER	IGNORE
BIN 16	5		1	255	UPPER	UPPER	IGNORE
BIN 31	11		1	255	UPPER	UPPER	IGNORE
BIN 32	10		1	255	UPPER	UPPER	IGNORE
BIN 64	20		1	255	UPPER	UPPER	IGNORE
CHAR <i>n</i>	<i>n</i>		<i>n</i>	32767	NONE	MIXED	RESPECT
IGC <i>n</i>	<i>n</i>		<i>n</i>	32767	NONE	MIXED	RESPECT
GRAPHIC <i>w</i>	<i>w</i>		<i>w</i>	16383	NONE	MIXED	RESPECT

表 58. BASETYPE ごとの属性要約 (続き)

BASETYPE	デフォルトの WIDTH	デフォルトの 属性	最小 WIDTH 属性	最大 WIDTH 属性	デフォルトの SHIFT 属性	デフォルトの CASE 属性	デフォルトの BLANKS 属性
GRAPHIC <i>w c</i>	<i>w</i>		<i>w/2</i>	16383	NONE	MIXED	RESPECT
NAME <i>n</i>	<i>n</i>		<i>n</i>	255	UPPER	UPPER	IGNORE
OBJNAME <i>n</i>	<i>n</i>		<i>n</i>	255	NONE	テキストを参照	IGNORE
PACKED <i>t f</i> where <i>t=f</i>	<i>t+3</i>		<i>f+3</i>	255	UPPER	<i>n/a</i>	<i>n/a</i>
PACKED <i>t f</i> where <i>t[=f</i> and <i>f>0</i>	<i>t+2</i>		<i>f+2</i>	255	UPPER	<i>n/a</i>	<i>n/a</i>
PACKED <i>t f</i> where <i>t[=f</i> and <i>f=0</i>	<i>t+1</i>		1	255	UPPER	<i>n/a</i>	<i>n/a</i>
ZONED <i>t f</i> where <i>t=f</i>	<i>t+3</i>		<i>f+3</i>	255	UPPER	<i>n/a</i>	<i>n/a</i>
ZONED <i>t f</i> where <i>t[=f</i> and <i>f>0</i>	<i>t+2</i>		<i>f+2</i>	255	UPPER	<i>n/a</i>	<i>n/a</i>
ZONED <i>t f</i> where <i>t[=f</i> and <i>f=0</i>	<i>t+1</i>		1	255	UPPER	<i>n/a</i>	<i>n/a</i>
DATE 2	8		8	255	UPPER	<i>n/a</i>	<i>n/a</i>
DATE 4	10		10	255	UPPER	<i>n/a</i>	<i>n/a</i>
TIME	8		8	255	UPPER	<i>n/a</i>	<i>n/a</i>
ACTION	6 (テキスト を参照)		1	255	UPPER	UPPER	IGNORE

オプションの属性

WIDTH=display-width.

このクラスの値が表示される表示フィールドの幅。GRAPHIC の BASETYPE では、この値は、DBCS 文字または UCS-2 文字の数になります。表示幅のデフォルト値および許容値は、530 ページの表 58 で明記されているように、BASETYPE 属性によって異なります。

このクラスを使用する変数の場合、BASETYPE のデフォルトより大きい値は、変換リストにある値の外部表示画面形式をサポートしていなければならない場合があります。変換リストについて詳しくは、686 ページの『TL (変換リスト)』を参照してください。

CHRID=NONE | PNLGRP

CRTPNLGRP コマンドの CHRID パラメーターが *JOBCCSID である場合、コンパイラーにより警告メッセージが出され、CLASS タグの CHRID 属性は無視されます。*JOBCCSID が指定されている場合、すべての変数はジョブ CCSID から装置 CHRID に変換されます。

NONE が指定されていれば、表示装置または印刷装置用の正しい文字セットおよびコード・ページでデータが入っていると見なされます。

PNLGRP が指定されていれば、このクラスに関連したデータは、CRTPNLGRP 制御言語コマンドの CHRID パラメーターで指定された文字セットおよびコード・ページを使用して解釈されます。このコード・ページをダウンロードしてこうしたデータの表示を処理できない装置の場合、このクラスの変数の出力時と入力時の両方で文字変換が行われます。このような場合の変換テーブルは、以下の規則によって決定されます。

	文字セット	コード・ページ
パネル・グループ	xxx	yyy
装置	aaa	bbb
出力変換テーブル	table = QUSRSYS/Qyyyyaabbb	
入力変換テーブル	table = QUSRSYS/Qbbbxxxyyy	

CLASS タグ

変換テーブルが見つからない場合、変換は行われません。この変換は、変換リストの妥当性検査またはインバウンド処理の前に実行されます。

GRAPHIC の BASETYPE の場合、CHRID=PNLGRP をこのタグで使用することはできません。GRAPHIC フィールドの 1 バイト部分がないため、UIM は、DBCS データが含まれる GRAPHIC フィールドでは文字セットおよびコード・ページの変換を実行しません。UCS-2 データが含まれる GRAPHIC フィールドの場合、指定された UCS-2 CCSID と装置 CCSID の間のみで変換が行われます。

CHRID=PNLGRP が IGC の BASETYPE を使用したクラスに指定されている場合、UIM は、フィールドの 1 バイト部分のみを変換します。フィールドの DBCS 部分では変換は行われません。

SHIFT=NONE | UPPER

NONE がコーディングされている場合、このクラスのフィールドに使用される特殊なキーボード・シフトはありません。

UPPER がコーディングされている場合、キーボードは、このクラスの変数に関連した画面入力フィールドのデータ入力項目用に大文字にシフトされます。デフォルトのキーボード・シフトは BASETYPE によって決定されます。BASETYPE および対応するデフォルトの要約については、530 ページの表 58 を参照してください。

CASE=MIXED | UPPER

MIXED は、このクラスの任意のフィールドの文字が、変数プールに割り当てられる際に、保存されることを示します。

UPPER は、小文字 (a から z) がプールへの割り当て前に大文字になることを示します。この変換は、このタグの CHRID 属性の処理後、および変換リストの妥当性検査またはインバウンド処理の前に実行されます。GRAPHIC の BASETYPE の場合、CASE=UPPER を使用することはできません。デフォルトの大/小文字は BASETYPE によって決定されます。BASETYPE および対応するデフォルトの要約については、530 ページの表 58 を参照してください。

BLANKS=RESPECT | IGNORE

RESPECT は、値が変数プールへ割り当てられる際に入力フィールドの空白が保存されることを示します。

IGNORE は、変数プールへ割り当てられる際に値が左寄せされることを示します。デフォルトは BASETYPE によって決定されます。先行空白を除去する処理は、このタグの CHRID および CASE 属性の処理後、および変換リストの妥当性検査またはインバウンド処理の前に実行されます。BASETYPE および対応するデフォルトの要約については、530 ページの表 58 を参照してください。

IGC の BASETYPE に BLANKS=IGNORE が指定されている場合、先行 SBCS 空白のみがフィールド値から除去されます。先行空白の除去処理は、フィールド内の最初のシフトアウト文字で停止します。

SUBST=DISPLAY | QUOTED

DISPLAY が指定されている場合、変数のこのクラスの CMD ダイアログ・コマンドで使用される置換値は、表示されている値と同じものです。

QUOTED が指定されていれば、変数のこのクラスの CMD ダイアログ・コマンドで使用される置換値は、アポストロフィで囲まれます。値の中のアポストロフィは二重になります。変換リスト (TL) タグを使用して変換された値は引用符で囲まれません。

BIDI=PNLGRP | LTR | RTL

このクラスの変数の左から右または右から左の方向付けを設定します。この属性は、BIDI=NONE が PNLGRP タグで指定されると、無視されます。

PNLGRP は、PNLGRP タグの BIDI 属性からのパラメーターを、このクラスの変数の方向付けを設定するために使用すべきことを示します。

LTR は、このクラスの変数には左から右の方向付けが使用されるべきことを示します。

RTL は、このクラスの変数には右から左の方向付けが使用されるべきことを示します。

CHAR および IGC の BASETYPE を使用したクラスの場合、方向付けはデフォルトで PNLGRP を使用します。NAME、OBJNAME、BIN、PACKED、ZONED、DATE、TIME、および ACTION の BASETYPE を使用したクラスの場合、方向付けのデフォルト値として LTR が使用されます。

方向付けがパネルの方向付けと逆である入力フィールドの場合、カーソルの方向は逆になります。このタグの CONXTREV 属性が処理されると、カーソルの方向は後で変更することができます。

CONXTREV=NO | YES

コンテキスト依存の取り消しがこのクラスの両方向変数に対して実行されることを指定します。この属性は、BIDI=NONE が PNLGRP タグで指定されると、無視されます。

CONXTREV=NO の場合、変数の方向付けがパネル・グループの方向付けと一致しなければ、変数の表示行ごとの値は、表示される前に逆になります。入力時に、方向付けがパネルと反対であるフィールドの各表示行の内容も、表示フィールドから変数プールへ移動される前に逆になります。NO がデフォルトです。

CONXTREV=YES の場合、変数の最終的な表示方向を決定するためにダイアログ変数の内容が検査されます。ダイアログ変数値は、表示される前に検査されます。ダイアログ変数が入力フィールドで使用される場合、入力値が変数プールにコピーされる前に、再び値が検査されます。検査に関する説明は以下のとおりです。

- PNLGRP タグで BIDI=LTR が指定されている場合、
 - ダイアログ変数が反転するのは、ダイアログ変数値に少なくとも 1 つのアラビア語またはヘブライ語のアルファベット文字が含まれている場合、またはダイアログ変数値にラテン語文字が含まれておらず CLASS タグで BIDI=RTL が指定されている場合だけです。
- PNLGRP タグで BIDI=RTL が指定されている場合、
 - ダイアログ変数が反転するのは、ダイアログ変数値にアラビア語またはヘブライ語のアルファベット文字が含まれておらず、ダイアログ変数値にラテン語文字が含まれているか、あるいは CLASS タグで BIDI=LTR が指定されている場合だけです。

変数の表示方位が反転する必要があるかどうかを判別するのに使用される、アラビア語およびヘブライ語のコード・ページは、次のように説明されます。

- コード・ページ 420 の場合
 - 以下の 16 進数値はアラビア語文字を表します。42 から 49、51、52、55 から 59、62 から 69、70 から 79、80、8A から 8F、90、9A から 9F、A0、AA から AF、B0 から B5、B8 から BF、CB、CD、CF、D0、および DA から DE。
 - 以下の 16 進数値はラテン語文字を表します。6F、81 から 89、91 から 99、A2 から A9、C1 から C9、D1 から D9、DF、E2 から EB、ED、EE、EF、F0 から F9、および FB から FE。
 - 残りの文字は、アラビア語およびラテン語であるとは見なされません。
- コード・ページ 424 の場合
 - 以下の 16 進数値はヘブライ語文字を表します。41 から 49、51 から 59、62 から 69、および 71。
 - 以下の 16 進数値はラテン語文字を表します。81 から 89、91 から 99、A2 から A9、C1 から C9、D1 から D9、E2 から E9、および F0 から F9。

CLASS タグ

- 残りの文字は、ヘブライ語およびラテン語であるとは見なされません。

CONTXTREV=YES 入力フィールドの場合、初期のカーソルの方向は、CONTXTREV=NO 入力フィールドの場合と同じ方法で決定されます。続いて、以下の内容否定の結果に基づいて、さらに処理および再決定が行われます。

- PNLGRP タグで `BIDI=LTR` が指定されている:

- CLASS タグで `BIDI=LTR` が指定されている場合:

- カーソルの方向が逆になるのは、ダイアログ変数の内容が逆転している場合のみです。

- CLASS タグで `BIDI=RTL` が指定されている場合:

- カーソルの方向が逆になるのは、ダイアログ変数の内容が逆転していない場合で、なおかつダイアログ変数の内容に少なくとも 1 つのラテン語アルファベット文字が含まれる場合だけです。(これにより、カーソルの方向が空フィールドで否定されないようにすることができます。)

- PNLGRP タグで `BIDI=RTL` が指定されている:

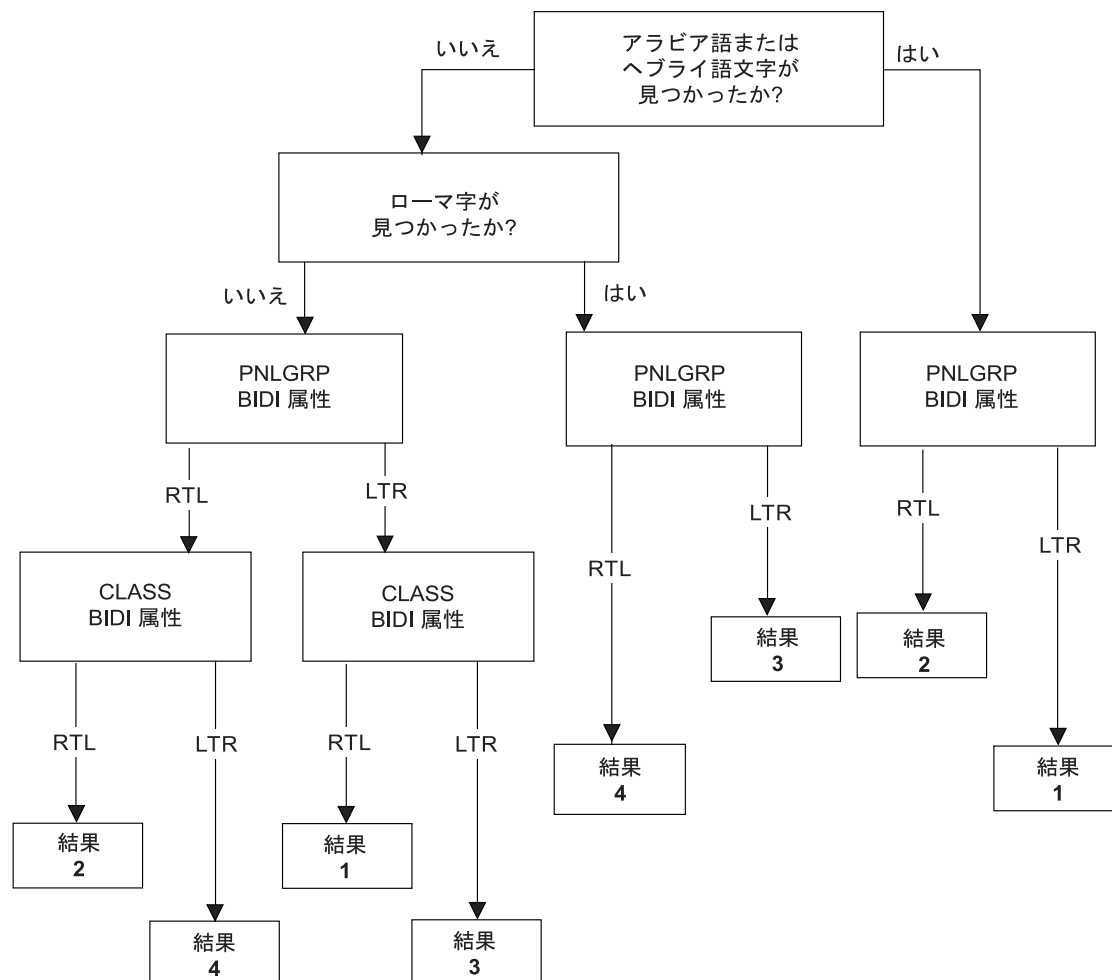
- CLASS タグで `BIDI=LTR` が指定されている場合:

- カーソルの方向が逆になるのは、ダイアログ変数の内容が逆転していない場合のみです。

- CLASS タグで `BIDI=RTL` が指定されている場合:

- カーソルの方向が逆になるのは、内容が逆転していて、なおかつ変数に少なくとも 1 つのラテン語アルファベット文字が含まれる場合だけです。(これにより、カーソルの方向が空フィールドで否定されないようにすることができます。)

CONTXTREV=YES がさまざまなフィールドおよびパネルで機能する方法に関する詳細な要約は、以下の決定木を参照してください。



結果 1:
右から左への
カーソル動作。
ダイアログ変数の
値を反転させる。

結果 2:
右から左への
カーソル動作。

結果 3:
左から右への
カーソル動作。

結果 4:
左から右への
カーソル動作。
ダイアログ変数の
値を反転させる。

RV2W060-0

NBRSHAPE=PNLGRP | ARABIC | HINDI

このクラスの両方向変数のアプリケーション生成数値の表示形状を制御します。この属性は、BIDI=NONE が PNLGRP タグで指定されるか、または 420 (アラビア語) 以外のコード・ページが表示装置によって使用されると、無視されます。

PNLGRP が指定されている場合、PNLGRP タグの NBRSHAPE 属性からのパラメーターをこのクラスの変数に使用すべきです。PNLGRP がデフォルトです。

ARABIC が指定されている場合、通常の数値 X'F0' から X'F9' が数値に使用されます。

HINDI が指定されている場合、数値の変換は、各国語ごとに定義される 2 つの変換テーブルのセットを使用して実行されます。最初の表は出力操作に使用され、2 番目の表は入力操作に使用されます。

NBRSHAPE 属性を使用すると、アプリケーション・プログラムが数値の表示に使用される実際のコード・ポイントを認識できないようにすることができます。

CLASS タグ

| SYMSWAP=NO | YES

| このクラスの両方向変数用の対称文字の交換を制御します。この属性は、BIDI=NONE が PNLGRP タグで指定されると、無視されます。対称文字の 2 つの対の例を次に示します。

| ' (と) '

| ' < と > '

| YES が指定されている場合、対称文字の交換は、右から左の方向付けを使用して表示される任意のフィールド上の入出力操作で実行されます。これが起こる状況には次のものがあります。

- | • このタグで CONXTREV=NO が、右から左の方向付けを持つ変数に指定されている。
- | • このタグで CONXTREV=YES が、右から左の方向付けを持つ変数に指定されており、フィールドの内容によって結果のフィールドの方向付けが右から左のままになる。
- | • このタグで CONXTREV=YES が、左から右の方向付けを持つ変数に指定されており、フィールドの内容によって結果のフィールドの方向付けが右から左に変更される。

| SYMSWAP=NO が指定されている場合、対称文字の交換は実行されません。NO がデフォルトです。

| ヘブライ語 (コード・ページ 424) の対称文字の完全リストは次のとおりです。

| (と) 右括弧および左括弧 (4D と 5D)

| { と } 右中括弧および左中括弧 (C0 と D0)

| [と] 右大括弧および左大括弧 (BA と BB)

| < と > より小記号とより大記号 (4C と 6E)

| 《 と 》 二重より小記号および二重より大記号 (8A と 8B)

| アラビア語 (コード・ページ 420) の対称文字の完全リストは次のとおりです。

| (と) 右括弧および左括弧 (4D と 5D)

| < と > より小記号とより大記号 (4C と 6E)

| UIM によって生成されるすべての対称文字は、SYMSWAP=YES が指定されたものとして処理されます。現時点で、対称文字には以下のものが含まれます。

| > より大記号。メニュー・オプションにマークを付けるのに使用されます (コマンド行プロンプトで使用)。

| () 右括弧および左括弧記号。空リスト・メッセージの周りに使用されます。

| 例: クラス定義

| UIM ソース

```
| .* Class of options whose only legal values
| .* are Y, N, and blank.
| .* These values are translated to an
| .* internal form.
| :class name=option
|     basetype='bin 15'
|     case=upper width=1.
| :tl msgid=MMM1234.
| :ti value=1.Y
| :ti value=2.N
| :ti value=3.
| :etl.
| :eclass.
| .*
```



```

| .* class of object names or the special
| .* value *all
| :class name=object
|     basetype='objname 10 generic'.
| :tl.
| :ti value='"*ALL"*.*ALL
| :etl.
| :eclass.
| .*
| .* Class of options from 1-5 or 7-9,
| .* in a field of width 1.
| :class name=listoption
|     basetype='bin 15' width=1.
| :check range='1 5' range='7 9'.
| :eclass.

```

数値の表示画面形式

数値は通常、先行ゼロなしで表示されます。数値の絶対値が 1 より小さい場合、ジョブの 10 進形式区切り属性の値に基づいて、小数点の前に先行ゼロが付けられるか、省略されます。数値は、負である場合にのみ符号が付けられます。小数点が表示されるのは、小数桁がある場合のみです。使用される小数点文字は、ジョブの 10 進形式区切り属性によって定義されたものです。数値は、すべての小数部の桁を保持したまま表示されます。

出力時に、値が大きすぎて表示フィールドに収まらない場合、フィールドはプラス符号で埋められます。フィールド全体が入力時にプラス符号で構成されている場合、妥当性検査はそのフィールドで実行されず、変数プール内の変数の値は変更されないままです。

数値リテラルは次の形式でなければなりません。



または



コンマは、入力時に小数点のピリオドの代わりに使用できます。内部形式への変換により有効数字が失われると、フィールドはエラーであると見なされ、UIM は、変数プールの内容を変更せずにエラー・メッセージをユーザーに表示します。小数点の後のすべてのゼロ以外の数字は有意であると見なされ、切り捨てようとするエラーになります。小数点の前の先行ゼロおよび小数点の後の後続ゼロは、有意とは見なされません。符号が指定されていない場合、値は正であると見なされます。

2 進数値は、出力時に文字ストリングに変換することができ、入力時に文字ストリングから変換することもできます。この場合、このタグの WIDTH 属性に指定される値は、変換リスト項目 (TI) タグで指定された変換済みの文字ストリングの幅を許容するものでなければなりません。変換リストについて詳しくは、686 ページの『TL (変換リスト)』および 685 ページの『TI (変換リスト項目)』を参照してください。

CLASS タグ

ACTION の BASETYPE を持つ変数は、ゼロ値がブランクとして表示され、ゼロ入力値が無効である場合を除いて、数値に関する編集規則を使用して処理されます。プラス符号は入力時に使用することはできません。変換リストおよび妥当性検査処理は、BASETYPE が ACTION である変数には許可されません。

数値の表示のいくつかの例が以下の表に示されています。

BASETYPE	WIDTH	内部値	表示される値	注
BIN 15	1	1	1	
BIN 15	1	-1	+ (値が大きすぎて表示できない)	
BIN 15	5	1	1	
BIN 15	5	-1	-1	
BIN 15	5	-3456	-3456	
BIN 15	5	-32768	++++ (値が大きすぎて表示できない)	
BIN 16	5	65535	65535	
BIN 31	10	2147483645	2147483645	
BIN 31	9	2147483645	+++++++ (値が大きすぎて表示できない)	
BIN 32	10	4294967295	4294967295	
BIN 32	9	4294967295	+++++++ (値が大きすぎて表示できない)	
BIN 64	20	18446744073709551615	18446744073709551615	
BIN 64	20	18446744073709551615	+++++++ (値が大きすぎて表示できない)	
PACKED 2 2		.45	.45	計算されたフィールド幅は 5
PACKED 2 2		-.45	-.45	計算されたフィールド幅は 5
PACKED 5 2		000.45	0.45	計算されたフィールド幅は 7
PACKED 5 2		-000.45	-0.45	計算されたフィールド幅は 7
PACKED 5 2	8	000.45	0.45	
PACKED 5 2	8	-000.45	-0.45	
PACKED 5 2	8	123.45	123.45	
PACKED 5 2	8	-123.45	-123.45	
PACKED 5 2	4	-123.45	+++ (値が大きすぎて表示できない)	
PACKED 5 2	4	003.45	3.45	
PACKED 5 0		00001	1	計算されたフィールド幅は 6
PACKED 5 0		00000	0	計算されたフィールド幅は 6
PACKED 5 0		-00000	0	計算されたフィールド幅は 6
PACKED 5 0		-00001	-1	計算されたフィールド幅は 6

文字、日付、および時刻値の表示画面形式

文字値は、変数プールにあるとおりに表示され、必要に応じて右側にヌルが埋め込まれます。入力フィールドに入力された値が変換リストを使用して変換されていない場合、エラーがユーザーに報告され、末尾ブラ

- | ンクの除去後に値が変数の BASETYPE より大きい場合、ダイアログ変数は更新されません。末尾の SBCS
- | ブランクのみが除去されます。DBCS ブランクは有意であると見なされます。

- | 日付は常に、現行ジョブの日付形式および日付区切り記号によって指定された形式で表示されます。ユーザ
- | ーによって日付が入力される場合、現行ジョブの日付形式によって指定された形式でなければなりません
- | が、現行ジョブの日付形式によって指定された区切り文字はオプションになります。日付形式では世紀桁の
- | 項目は使用できないため、年の数値が 40 より大きい場合には、それは 0 であると見なされます。年の数
- | 値が 40 より小か等しい場合、世紀桁は 1 であると見なされます。

- | 'DATE 4' として定義されるフィールドでは、ユーザーは完全な 4 桁の年で日付を入力する必要があります。ありま
- | す。ユーザーが 2 桁の年のみで日付を入力した場合、日付は無効であるとしてフラグが立てられます。

- | 時刻は、現行ジョブの時刻区切り文字によって指定された時刻区切り文字を使用して、常に 24 時間の時刻
- | 形式で表示されます。現行ジョブの時刻区切り文字によって指定された区切り文字はオプションです。時刻
- | 区切り文字を使用せずに時刻を入力する場合、4 桁または 6 桁のいずれかが提供されなければなりません
- | 。

- | 時刻値に時間帯が含まれる場合、その時間帯値は、値の残りの部分から単一の空白文字で区切られる必
- | 要があります。入力時に、時間帯値は時刻値の残りの部分からシングル・スペースで区切られる必要があり
- | ます。そうしないと、エラーが通知されます。

CMDLINE (コマンド行)

▶▶:CMDLINE=SIZE=SHORT|LONG|NAME=command-line-name|instruction-text▶▶

コマンド行 (CMDLINE) タグは、パネルにコマンド行があることを指定し、追加のコマンド行情報を提供します。このタグは、表示パネルでのみ許可されます。コマンド行は、任意のパネルで表示でき、パネル上のすべての領域から独立させることができます。または、パネル上の処置リストあるいはメニュー域に関連付けることもできます。

コマンド行で入力できるのは、パネルにメニュー域または処置リスト域が含まれる場合を除き、システム・コマンドのみです。パネルにメニュー域が含まれる場合、ユーザーは、メニュー・オプションを入力するか、またはコマンド行でコマンドを入力します。処置リストの場合、ユーザーは、コマンドとして実行されるか、処置リスト処理用のパラメーター情報として動作するストリングを入力します。

このタグは、パネルの最終タグでなければならず、表示パネルの終わりの直前に置かれます。CMDLINE タグとオプション行 (OPTLINE) タグは相互に排他的です。

必須属性

SIZE=SHORT | LONG

SHORT コマンド行は、画面の 1 つの行のみを取り、その行を充てんします。

LONG コマンド行は画面上の 2 つの行を取ります。

表示パネル (PANEL) タグで、80 バイト (または 132 バイト) 以外のパネル幅が指定されている場合、SIZE=SHORT を指定しなければなりません。SIZE=LONG は、27 行× 132 バイトのパネル、または 24 行× 80 バイトのパネルの場合にのみ有効です。

CMDLINE タグ

パネル・グループ (PNLGRP) タグで BIDI=RTL が指定されている場合、すべての SIZE=LONG コマンド行はコンパイラーによって 1 行のコマンド行に自動的にフォーマットされ、コマンド行と機能キー域の間にはブランク行区切り文字が入れられます。

オプションの属性

NAME=*command-line-name*

コマンド行に関連した名前。この名前は、Add Pop-Up Window (QUIADDPW) API とともに後で使用し、コマンド行の近くにウィンドウを配置することができます。

命名規則については、515 ページの『名前構文』を参照してください。

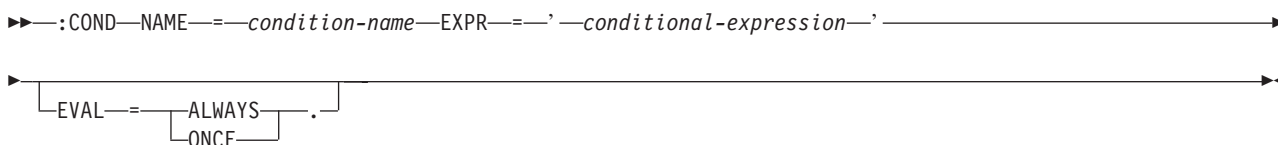
オプションのテキスト

instruction-text

コマンド行の指示として表示されるテキスト。このテキストは暗黙の段落です。

画面のフォーマット時に、1 表示行に収まらないテキストは、必要に応じて複数行にフォーマットされ、2 表示桁下げされます。テキストは最大 255 文字で、テキスト反転 (RT) タグだけを含めることができます。テキストが提供されない場合は、指示行はコマンド行の上に表示されません。

COND (条件定義)



条件定義 (COND) タグは、特定の処理が行われる場合に真でなければならない条件を定義します。他のタグがこの条件を参照することもあります。パネルが表示される前に評価されます。条件が真であれば、参照中のタグによって指定されたアクションまたはフォーマットが実行されます。

必須属性

NAME=*condition-name*

式で指定された条件の名前。この名前は、パネル・グループ内で固有でなければなりません。

命名規則については、515 ページの『名前構文』を参照してください。

EXPR='conditional-expression'.

条件式は、以下の形式で真または偽の式になります。



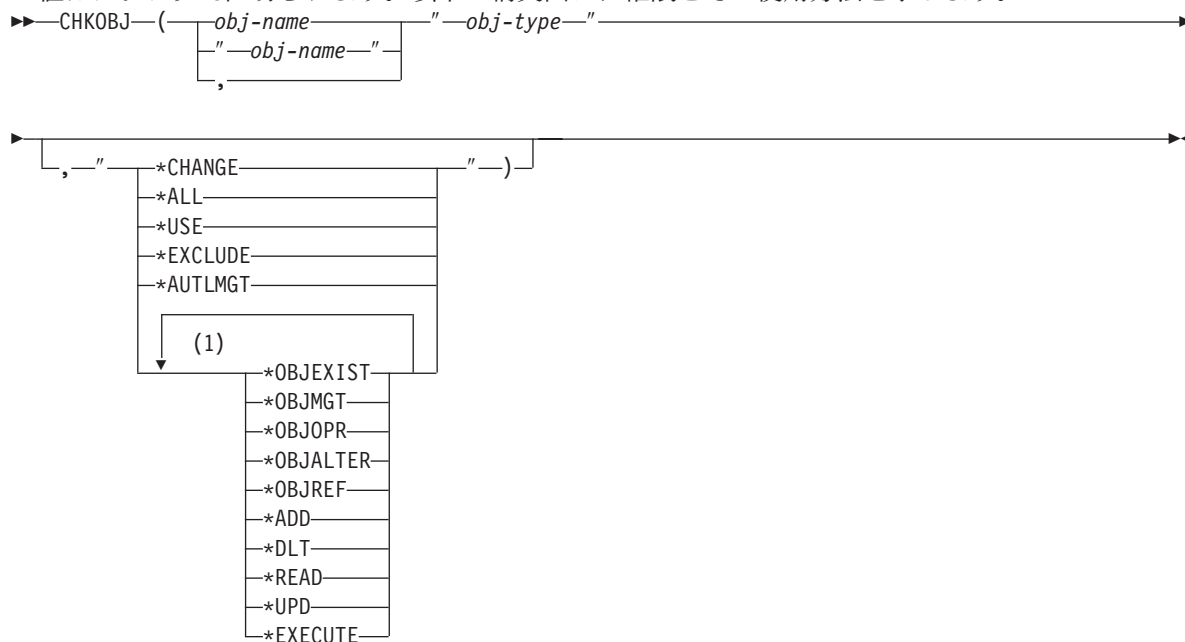
ここで、A は以下のいずれかオペランドになります。

- 条件式
- ダイアログ変数とリテラル値の比較

COND タグ

CHKOBJ

オブジェクトがシステム上にあり、現行ジョブが、少なくとも権限によって指定された許可レベルをオブジェクトに対して所有している場合に、真と評価されます。引数は、二重引用符 (") で囲まれた文字ストリング・リテラルでなければなりません。オブジェクト名は、i5/OS オブジェクトの命名規則に従います。オブジェクト・タイプは DSPOBJD コマンドの許容オブジェクト・タイプのいずれかであり、権限は検査対象の単一値または権限リストである必要があります。これらの権限の値は空白で区切られます。以下の構文図に、権限とその使用方法を示します。



注:

1 それぞれの値は 1 回だけ使用でき、最大値は 7 です。

権限が指定されていない場合、許可検査は実行されず、この関数は存在検査になります。

CHKPGM

CHKPGM 関数は、UIM から出口プログラムとして呼び出される、修飾プログラム名を受け入れます。プログラムを呼び出すことができない場合、関数は偽と評価されます。出口プログラムは、関数が真または偽のいずれに設定されるかを判別し、標識を UIM に戻します。プログラム名は、二重引用符 (") で囲まれていなければなりません。*LIBL 特殊値はライブラリー名の代わりに使用できます。ライブラリー名が入力されていない場合、*LIBL がデフォルトになります。

以下の構文図に、CHKPGM 関数の有効な引数値を示します。



CHKUSRCLS

関数に指定されたユーザー・クラス引数は、現行ジョブのユーザー・プロファイルからのユーザー・クラス・パラメーターと比較されます。ユーザー・プロファイルの値が関数の引数と同じかそれ以上の場合、関数は真と評価されます。関数の引数は二重引用符 (") で囲まなければなりません。

以下の構文図に、ユーザー・クラスの有効な引数値を示します。



```

CHKOBJ("OBJECT", "*FILE", "*USE")

CHKOBJ("PANELGRP", "*PNLGRP")
  *AND CHKUSRCLS("PGMR")

CHKOBJ("DOCUMENT", "*DOC", "*READ *UPD")
  *OR CHKUSRCLS("SYSOPR")

*NOT(CHKOBJ("PROGRAM", "*PGM"))
CHKPGM("*LIBL/PROGRAM")
CHKPGM("PROGRAM")

```

オプションの属性

EVAL=ALWAYS | ONCE

条件が参照されるたびにこのタグの `EXPR` 属性で指定された式を評価する必要があるのか、それとも 1 回だけ評価する必要があるのかを示します。

`EVAL=ALWAYS` が使用されている場合、条件が参照されるたびに新規の真の値を決定するために式が評価されます。パネル・グループまたはメニューにあるすべての `EVAL=ALWAYS` 条件は、パネルまたはメニューが表示されるたびに評価されます。 `ALWAYS` がデフォルトです。

`EVAL=ONCE` が指定されている場合、パネル・グループを使用するオープン・アプリケーションごとに 1 回だけ、式が評価されます。 `UIM` は、条件が参照される初回のみ、このタイプの条件について式を評価します。最初の参照後、すべての参照は最初の参照によって決定されたのと同じ真の値を使用します。 `ONCE` は、コストがかかる可能性のある関数 (`CHKOBJ` など) の評価を式が必要としている場合は特に、アプリケーションのオープン中に変化しない条件のパフォーマンスを改善するために指定できます。

例: オプションの条件付け

以下の例では、`COND` タグを使用してメニュー項目を条件付ける方法を示します。最初の条件は、2 つのソース行に分割された複雑な式を使用します。2 番目の条件では、システム上でメニューを使用するためのアプリケーション・ユーザーの権限を検査し、3 番目の条件は、アプリケーション・ユーザーのユーザー・クラスを検査します。

UIM ソース

```

:
:cond name=a
  expr='(var1 = 100 *AND var2 > 0)'
  expr='*OR *NOT(var3 = 0)'.
:cond name=b
  expr='chkobj("MENU2", "*MENU", "*READ, *UPD")'.
:cond name=c
  expr='chkusrcls("PGMR")'.
:
:menui
  cond=a
  help='option1/help'
  action='CMD CALL PGM01'

```

COND タグ

```
option=1.Run report
:menui
cond=b
help='option2/help'
action='MENU MENU2'
option=2.Administration
:menui
cond=c
help='option30/help'
action='MENU UTIL'
option=30.Utilities
:
```

COPYR (著作権)

▶▶:COPYR—.—*notice-text*—▶▶

著作権 (COPYR) タグは、著作権表示を提示する方法を提供します。1 つの COPYR タグだけが許可されます。著作権表示は、パネル・グループがオープンされた後に表示される最初のパネルのメッセージ領域に表示されます。このタグは、パネル・グループ (PNLGRP) タグの直後にコーディングされなければなりません。

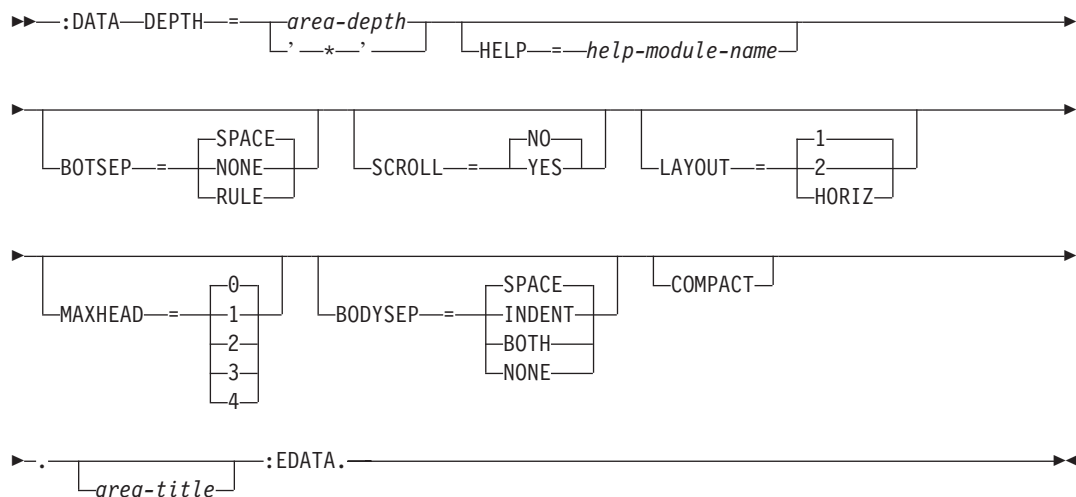
必須テキスト

notice-text

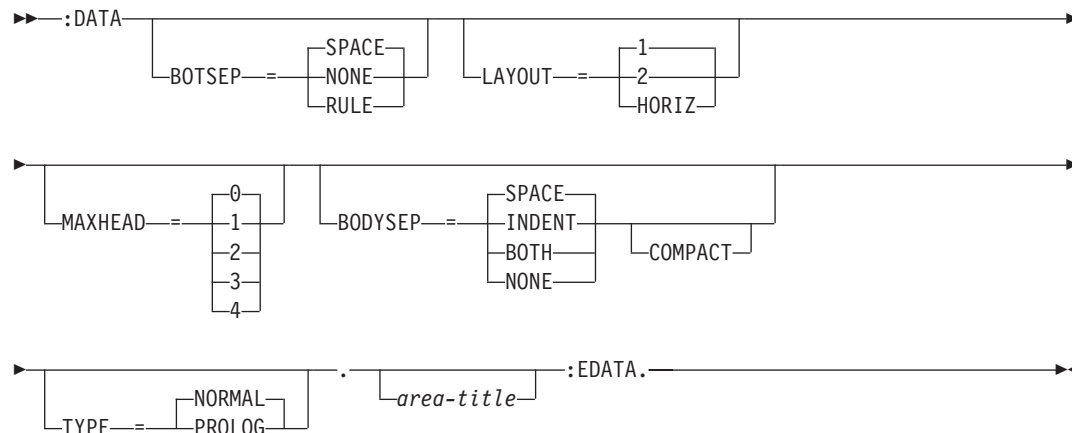
表示される著作権表示のテキスト。テキストは 1 から 76 文字でなければなりません。

DATA (データ表示域)

表示パネルの構文:



印刷パネルの構文:



データ表示域 (DATA) タグは、パネル内でのデータ表示域について記述します。このタグは、表示パネルおよび印刷パネルで許可されます。この領域は、データ入力項目または出力データ項目、あるいはこの 2 つの組み合わせに使用できます。

他のタグを DATA タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 59. DATA タグと EDATA タグの間で許可されるタグ

タグ名	順序	使用法	ページ
TOPINST (最上部指示行)	1	D	687
DATA COL (データ列)	2	B	555
DATAGRP (データ項目グループ)	3	B	556
DATA I (データ項目)	3	B	559
DATASLT (データ選択フィールド)	3	D	568
BOTINST (最下部指示行)	4	D	521

必須属性

DEPTH=area-depth | '*'

領域の深さを行で示したものの。区切り記号が指定されていれば、それも含まれます。この属性は表示パネルには必要ですが、印刷パネルでは許可されません。 '*' が指定されている場合、他のすべてが割り振られた後で画面上に残っているスペースはこの領域に割り当てられます。パネル内の 1 つの領域のみ、 '*' でコード化できます。

オプションの属性

HELP=help-module-name

領域内のすべての項目について説明するオンライン情報を示します。この属性は表示パネルにのみ許可されます。ヘルプ・モジュールの名前は別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説された名前の規則に従っていなければなりません。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

領域内のすべての項目には、それぞれ単一のヘルプ・モジュール名を関連付ける必要があります。

DATA タグに HELP 属性が指定されている場合、ヘルプは領域内のすべてのグループおよび項目に適用されます。また、HELP 属性は、領域内のデータ・グループ (DATAGRP)、データ項目 (DATAI)、データ選択フィールド (DATASLT)、およびデータ選択フィールド選択項目 (DATASLTC) タグでは許可されません。

DATA タグに HELP 属性が指定されていない場合は、DATAGRP タグ、または DATAI タグおよび DATASLT タグで HELP 属性を指定して、ヘルプ・モジュール名を領域内の個々の項目に関連付けなければなりません。

ユーザーが、このタグに LAYOUT=HORIZ 属性を指定したデータ域に関するオンライン情報にアクセスしやすくするには、個々の DATAI タグに HELP 属性を指定する代わりに、DATA タグに HELP 属性を指定します。

BOTSEP=SPACE | NONE | RULE

データ表示域の下部区切り記号を定義します。SPACE が指定されている場合、スペースから成る 1 行が使用されます。SPACE がデフォルトです。

NONE は、区切り行が存在しないことを示します。

RULE は、下線付きのスペースの行が使用されることを示します。

SCROLL=NO | YES

領域がスクロール可能かどうかを指定します。この属性は表示パネルにのみ許可されます。NO は、領域がスクロール可能でないことを示します。NO がデフォルトです。

YES は、データ表示域がスクロール可能であることを示します。SCROLL=YES は、LAYOUT=HORIZ が指定されている場合には許可されません。スクロール情報の行を提供するために、UIM によって 1 行のスペースが使用されます。BOTSEP=SPACE であれば、この領域に最下部指示も含まれている場合を除き、スペースの 1 行のみが使用されます。

LAYOUT=1 | 2 | HORIZ

縦方向にフォーマットされた領域のレイアウト列の数、または横形式を使用すべきかどうかを示します。

LAYOUT=1 の場合、データ項目の列は 1 つだけです。パネル定義 (PANEL) タグで WIDTH=80 バイトのパネルの場合、データ項目の単一列が 2 ~ 79 桁目にあります。WIDTH=132 のパネルの場合、データ項目の単一列が 2 ~ 131 桁目にあります。

LAYOUT=2 の場合、幅が等しい 2 つのレイアウト列がデータ項目に使用されます。領域内で指定されたデータ項目は、レイアウト列の上から下へ、レイアウト列間では左から右へという順番で表示されます。WIDTH=80 のパネルの場合、2 つのレイアウト列が 2 ~ 38 および 43 ~ 79 桁にあります。WIDTH=132 のパネルの場合、データ項目の 2 つのレイアウト列が 2 ~ 64 および 69 ~ 131 桁にあります。

LAYOUT=HORIZ の場合、データ・フィールドは、項目間に 5 つのスペースを入れて、パネルの幅にまたがって横方向にリストされます。このレイアウトは、2 つまたは 3 つの短い値のみが必要な場合に役立ちます。これは、単一の表示行に制限すべきです。これは、UIM が領域内の異なる行に表示される項目を縦方向に位置合わせしないためです。

MAXHEAD=0 | 1 | 2 | 3 | 4

列見出しに使用可能な最大行数。列見出しは、データ列 (DATACOL) タグを使用して指定されます。0 から 4 行を指定できます。0 がデフォルトです。これは、列見出しが許可されないことを意味します。列見出しは横方向表示では使用できないため、横方向レイアウトを持つデータ表示域には MAXHEAD=0 のみが有効です。列見出しについて詳しくは、555 ページの『DATACOL (データ列)』を参照してください。

列見出しが使用されている場合、必要な見出し行の数よりも大きい MAXHEAD 値を指定することによって、各国語の翻訳のための拡張スペースを提供することもできます。そうでない場合、DATACOL タグによって決定される列の幅以内で拡張スペースを提供する必要があります。

BODYSEP=SPACE | INDENT | BOTH | NONE

画面上で領域の本体と他のエレメント (特に領域内の最上部指示行と最下部指示行) を区別する視覚的な区切りのタイプ。

SPACE は、最後の最上部指示行の後および最初の最下部指示行の前に空白行を置きます。領域の本体にある項目は、レイアウト列の左端から始まります。項目は、指示行については字下げされません。領域に最上部または最下部指示行が含まれない場合、領域本体の前後で空白行が確保されることはありません。

INDENT は、領域の本体にある項目を、指示行が始まるレイアウト列の左端から 2 バイト字下げするのに使用されます。領域に最上部または最下部指示行が含まれる場合、領域がスクロール可能な場合を除き、指示と領域本体の間で空白行が確保されることはありません。領域がスクロール可能であれば、スクロール情報用の行を提供するために空白行が確保されます。

BOTH は、最後の最上部指示行の後および最初の最下部指示行の前に空白行を置き、さらに、領域の本体にある項目を、指示行が始まるレイアウト列の左端から 2 バイト字下げします。領域に最上部または最下部指示行が含まれない場合、領域本体の前後で空白行が確保されることはありません。

NONE は、指示行と本体の間に空白行を置かず、領域がスクロール可能な場合を除き、レイアウト列について本体を字下げすることはありません。領域がスクロール可能であれば、スクロール情報用の行を提供するために空白行が確保されます。

COMPACT

COMPACT が指定されている場合、領域にある列見出しとすべての項目および外部グループの間に空白行は残されません。COMPACT が指定されていない場合、列見出しの後に、またすべての項目と外部グループの間に空白行が入れられます。

TYPE=NORMAL | PROLOG

このデータ域がプロログ域かどうかを示します。プロログ域は、先頭ページのタイトル行の後に 1 回限り印刷されます。この属性は、印刷ヘッド・パネル (PRTHEAD) タグにのみ許可されます。

NORMAL がデフォルト値です。

オプションのテキスト

area-title

領域のタイトル。テキストが指定されていない場合、領域にタイトル行は割り振られません。テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみ含めることができ、最大長は 55 文字です。

印刷の書式設定の考慮事項

印刷されるデータ域は、表示されるデータ域と同じようにフォーマットされますが、以下の例外があります。

- 印刷時に、データ・グループの見出しおよびデータデータ列に加えて、ページ上にはデータ入力項目の少なくとも 2 つの行があります。
- データ入力項目の 2 つの行がページに収まらない場合、ページ排出が発生し、データ入力項目は次のページに印刷されます。

LAYOUT=2 であるデータ域のフォーマットの場合、UIM は、ページにまたがってデータ入力項目のバランスを取り、保持処理は無視されます。

このフォーマットは見栄えが良くない場合がありますが、LAYOUT=2 のデータ域は、小さいデータ域でのみ、またユーザーが保持処理に関心がない場合にのみ使用すべきです。

データ列見出しがある場合、データ域が別のページに続くようであれば、それらの見出しは各ページで繰り返されます。

例 1: データ入力パネル

以下の例では、データ入力パネルのサンプルを示します。パネルには 5 つのプロンプトがあります。4 つのプロンプトは、事前に関数のダイアログ変数を設定することによって、デフォルトを呼び出し側プログラムに提供します。

UIM ソース

```
:panel name=entry1
  help=hentry1
  topsep=space
  key1=keys
  .Sample Entry Panel
:data depth='*'
  maxhead=2.
:topinst.Type choices, press Enter:
:datacol width=20.Item
:datacol width=10.Choice
:datacol width='*'. 'Possible Choices'
:datai usage=inout
  help=hfname
  var=fname.File name
:datac.Name of document to be printed
:datai usage=inout
  help=hstyle
  var=prtstyle.Type style for printing
:datac.1=Prestige Elite (12 pitch)
:datac.2=Courier (10 pitch)
:datac.3=Essay Standard (proportional)
:datac.4=Essay Bold (proportional)
:datai usage=inout
  help=hmargin
  var=margin.Left margin
```

```

:datac.Number of spaces from the left
edge of the paper (1-20)
:datai usage=inout
      help=hcopy
      var=copies.Copies
:datac.Number of copies (1-99)
:datai usage=inout
      help=hduplx
      var=duplex.Duplex
:datac.1=Yes (Print both sides of paper)
:datac.2=No (Print one side only)
:edata.
:epanel.

```

結果

Sample Entry Panel		
Type choices, press Enter:		
Item	Choice	Possible Choices
File name	_____	Name of document to be printed
Type style for printing	<u>1</u>	1=Prestige Elite (12 pitch) 2=Courier (10 pitch) 3=Essay Standard (proportional) 4=Essay Bold (proportional)
Left margin	<u>6</u>	Number of spaces from the left edge of the paper (1-20)
Copies	<u>1</u>	Number of copies (1-99)
Duplex	<u>1</u>	1=Yes (Print both sides of paper) 2=No (Print one side only)
F3=Exit F12=Cancel		

例 2: データ入力パネルでの 2 列形式

以下の例では、出力項目を含む 2 列形式を使用したデータ域、およびプロンプトを持たないデータ入力項目フィールドの形式を示します。

UIM ソース

```

:panel name=panelx
      help=helpx
      topsep=space
      key1=keys
      .Another Sample Panel
:panel depth=14 layout=2.
:panel datacol width='24'.
:panel datacol width='*'.
:panel datagrps compact.Compact data group
:panel datai usage=out
      help=hvar1
      var=var1.First element
:panel datai usage=out
      help=hvar2
      var=var2.Second element
:panel edatagrps.
:panel datai usage=out
      help=hvar3
      var=var3.Third element

```

DATA タグ

```
:datagrpf grpsep=qindent
compact.
:datai usage=out
help=hvar4
var=var4.Fourth element
:datai usage=out
help=hvar5
var=var5.Fifth element
:datai usage=out
help=hvar6
var=var6.Sixth element
:edatagrpf.
:datagrpf compact.Another compact group
:datai usage=out
help=hvar7
var=var7.Seventh element
:datai usage=out
help=hvar8
var=var8.Eighth element
:datai usage=out
help=hvar9
var=var9.Ninth element
:edatagrpf.
:datagrpf help=hvar1012
grpsep=none compact.
:datai usage=out
var=var10.Tenth element
:datai usage=out
var=var11 cond=cond11.
:datai usage=out
var=var12 cond=cond12.
:edatagrpf.
:datagrpf grpsep=none
compact.
:datai usage=out
help=hvar13
var=var13.Thirteenth element
:datai usage=out
help=hvar14
var=var14.Fourteenth element
:datai usage=out
help=hvar15
var=var15.Fifteenth element
:edatagrpf.
:edata.
:data depth=3
bodysep=none.
:datacol width=0.
:datacol width='*'.
:datagrpf.Sixteenth element
:datai usage=inout
help=hvarx
var=varx.
:edatagrpf.
:edata.
:epanel.
```

結果

```

                Another Sample Panel

Compact data group:                Another compact group:
  First element . . . : XXXXXXXXXX  Seventh element . . : XXXXXXXX
  Second element . . . : XXXXXXXXXX  Eighth element . . . : XXXXXXXX
                                     Ninth element . . . : XXXX
Third element . . . . : XXXXXXXX
                                     Tenth element . . . . : XXXXXXXX
Fourth element . . . . : XXXXXXXX   XXXXXXXX
  Fifth element . . . : XXXXXXXX     XXXXXXXX
  Sixth element . . . : XXXXXXXX
                                     Thirteenth element . . : XXXXXXXX
                                     Fourteenth element . . : XXXXXXXX
                                     Fifteenth element . . : XXXXXXXX

Sixteenth element:
_____

F3=Exit  F12=Cancel

```

例 3: データ項目用の 2 つの表示域

以下の 2 つの例では、変数プール内のダイアログ変数から初期設定された 3 つのデータ域を示します。最初の 2 つの領域は横方向に位置合わせされ、3 番目の領域は縦方向にフォーマットされます。

UIM ソース

```

:panel name=xmp3
      key1=x1
      help=hxmp3
      .Data Item Extenders
:data depth=2
      layout=horiz
      botsep=space.
:datai usage=out
      var=date
      help=hdatetime.Date and time
:dataix usage=out
      var=time
      newline=no
      itemsep=1.
:edata.
:data depth=2
      layout=horiz
      botsep=space.
:datai usage=out
      var=jobname
      help=hdatetime.Job name
:dataix usage=out
      var=user
      newline=no
      itemsep=1.
:dataix usage=out
      var=jobnbr
      newline=no itemsep=1.
:edata.
:data depth='*'
      layout=1
      botsep=space
      compact.
:datacol width=0.

```

DATA タグ

```
:datacol width='*'.
:datagrpf grpsep=indent
      help=dependptfs
      compact.Dependent PTFs
:datayi usage=out
      var=ptf1.
:datayix usage=out
      var=ptf2.
:datayix usage=out
      var=ptf3.
:datayix usage=out
      var=ptf4
      newline=yes.
:datayix usage=out
      var=ptf5.
:edatagrpf.
:edata.
:epanel.
```

結果

```
                                Data Item Extenders

Date and time:  12/31/99 12:59:59

Job name:  DSP01      QSECOFR      012345

Dependent PTFs:
XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX

F3=Exit  F12=Cancel
```

例 4: メニュー域を持つデータ表示域

UIM ソース

```
:panel name=xxx
      key1=x1
      help=hxxx
      .Work with Files
:datay depth=2
      layout=horiz.
:datayi var=fname
      help='filename'.File
:datayi var=lname
      help='libname'.Library
:edata.
:menu depth='*'.
:topinst.Select one of the following:
:menui action='return 1'
      help='hoption1'
      option=1.Display file attributes
:menui action='return 2'
```



```

        help='hoption2'
        option=2.Display file contents
:menui action='return 3'
        help='hoption3'
        option=3.Change ownership
:menui action='return4'
        help='hoption4'
        option=4.Change authorizations
:menui action='return 5'
        help='hoption5'
        option=5.Destroy
:menui action='return 6'
        help='hoption6'
        option=6.Backup to tape
:emenu.
:optline.Selection
:epanel.

```

結果

```

                                Work with Files
                                System:  xxxxxxxx
File:  MEM047123      Library:  QGPL

Select one of the following:

    1. Display file attributes
    2. Display file contents
    3. Change ownership
    4. Change authorizations
    5. Destroy
    6. Backup to tape

Selection
-
F3=Exit  F12=Cancel

```

例 5: ネストされたデータ・グループを持つデータ入力パネル

UIM ソース

```

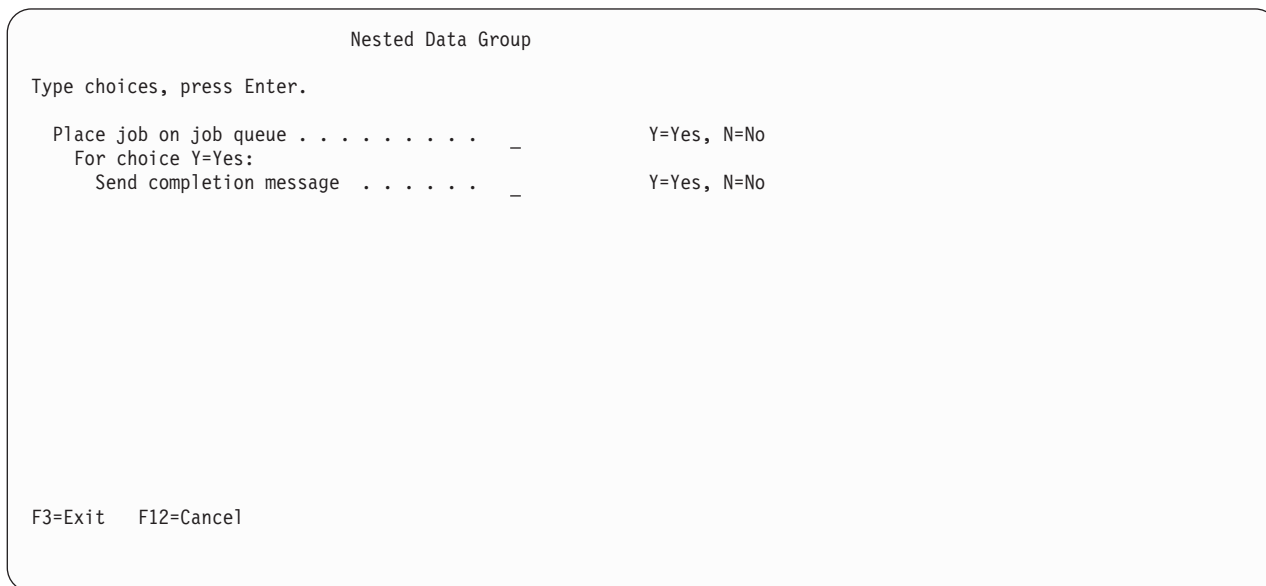
:panel name=nestg
        help=helpn
        topsep=space
        key1=keys
        .Nested Data Group
:data depth='*'
        scroll=no.
:topinst.Type choices, press Enter.
:datacol width=40.
:datacol width=10.
:datacol width='*'.
:datagr compact
        grpsep=none.
:datai usage=inout
        help=hjobq
        var=jobq.Place on job queue
:datac.Y=Yes, N=No
.* The following "datagr" tag is

```

DATA タグ

```
.*needed to cause indentation for
.* the "For choice Y=Yes" data group.
:datagrpf compact
    grpsep=indent.
:datagrpf compact
    grpsep=indent.For choice Y=Yes:
:datalf usage=inout
    help=hcmpmsg
    var=cmpmsg.Send completion message
:datacf.Y=Yes, N=No
:edatagrpf.
:edatagrpf.
:edatagrpf.
:epanel.
```

結果



DATAAC (データ項目選択)

```
▶▶:DATAAC [CHOICE=dialog-variable-name] . [text-for-choices] ▶▶
```

データ項目選択 (DATAAC) タグは、データ表示域の可能な選択列にテキストを提供します。このタグは、表示パネルおよび印刷パネルで許可されます。これは、データ列 (DATAACOL) タグが領域で 2 つだけ指定されているか、または領域のデータ表示域 (DATA) タグで LAYOUT=HORIZ が指定されている場合は無効です。

DATAAC タグは、対応するデータ項目 (DATAI) タグの後になければなりません。いくつかのテキスト行を提供するために、複数の DATAAC タグを DATAI タグの後に指定できます。DATAAC タグが入力されていなければ、データ項目の可能な選択列にテキストは存在しません。

オプションの属性

CHOICE=dialog-variable-name

表示する選択項目テキストを含むダイアログ変数の名前。ダイアログ変数は、データ域の 3 番目の

DATA COL タグで指定された幅 (可能な選択項目列) より小さいか等しい幅で定義する必要がありません。CHOICE 属性が使用されている場合、このタグのピリオドの後に *text-for-choices* を指定することはできません。

ダイアログ変数は、クラス定義 (CLASS) タグで BASETYPE に CHAR、IGC、または BIN を指定して定義する必要があります。

ダイアログ変数のエラー状態は、テキストの強調表示の判別には使用されません。

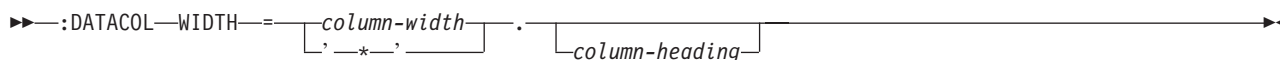
IGC 用の特殊フォーマット。 (省略形 IGC は、コマンドおよびキーワードで使用され、2 バイト文字セット機能を表します。) BASETYPE が IGC であるダイアログ変数を CLASS タグ上で指定すると、UIM は特殊フォーマットを行います。変数値がシフトアウト文字 (X'0E') で始まる場合は、UIM は値を左に 1 バイト移動して、他の行のデータ選択項目と垂直位置合わせを行います。

オプションのテキスト

text-for-choices

このテキストは暗黙の段落です。行は、必要に応じて列内の複数の行にフォーマットされ、列の先頭から 2 スペース字下げされます。テキストは最大 255 文字で、テキスト反転 (RT) タグだけを含めることができます。このタグに CHOICE 属性を指定しない限り、*text-for-choices* は必須です。

DATA COL (データ列)



データ列 (DATA COL) タグは、項目列、選択列、および可能な選択列の幅を指定し、列見出しを提供することもできます。このタグは、表示パネルおよび印刷パネルで許可されます。これは、データ表示域で LAYOUT=HORIZ が指定されている場合は無効です。これは、領域のデータ表示域 (DATA) タグの後、および最初のデータ項目 (DATA I)、データ選択フィールド (DATA SLT)、またはデータ・グループ (DATA GRP) タグの前になければなりません。

DATA タグの LAYOUT 属性が 1 または 2 を指定していれば、2 つまたは 3 つの DATA COL タグを指定する必要があります。2 つの DATA COL タグだけが使用されている場合、可能な選択列は領域に含まれず、データ項目選択 (DATA C) タグも領域で使用することはできません。3 つの DATA COL タグが指定されていれば、可能な選択列を含む 3 つの列すべてが割り振られます。

必須属性

WIDTH=column-width | '*'

列の幅 (バイト)。指定された幅は正の整数でなければならず、データ表示域の列と区切り記号の幅の合計は、DATA タグによって決定されたレイアウト列の幅を超えてはなりません。WIDTH=0 は、最初の DATA COL タグにのみ許可されます。これは、データ項目またはデータ選択フィールドのプロンプト・テキスト用の項目列を使用せずに領域をフォーマットします。

'*' がコーディングされている場合、領域の幅の残りの部分は列に使用されます。領域にある 1 つの DATA COL だけが '*' を指定できます。WIDTH=0 が最初の DATA COL タグに指定されている場合を除き、3 列区切り記号は保持されます。この場合、項目列は割り振られず、選択項目列の前に区切り記号は置かれません。

DATACOL タグ

データ項目の値は列をまたがる場合がありますが、正しい列の境界で開始される必要があります。3 つの DATACOL タグが指定されていると仮定した場合、その項目の可能な選択項目テキストは、可能な選択項目に割り当てられた場所から開始されます。この可能な選択項目テキストは、データ項目フィールドと同じ行または次の行にある場合があります。

オプションのテキスト

column-heading

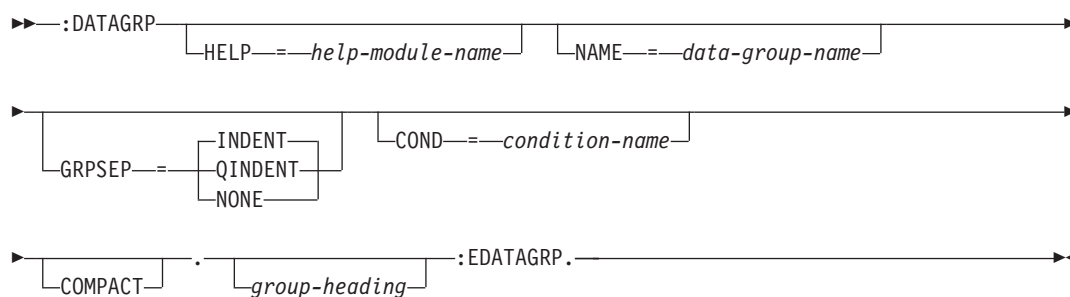
データ域にあるデータ項目の上に置かれる列見出し。列見出しが指定されていない場合は、何も表示されません。

テキストは複数の行に表示されることがあります。また、テキスト反転 (RT) タグのみを含めることができます。列見出しのそれぞれの語は新規の行に置かれます。列見出しの行に複数の語が必要な場合、それらをアポストロフィ (') で囲む必要があります。それぞれの語または引用符付きストリングは、このタグで WIDTH 属性によって定義された列の幅に収まらなければなりません。許可される語または引用符付きストリングの最大数は、DATA タグの MAXHEAD 属性によって指定されます。MAXHEAD=0 がデータ域に使用されている場合、列見出しは許可されません。

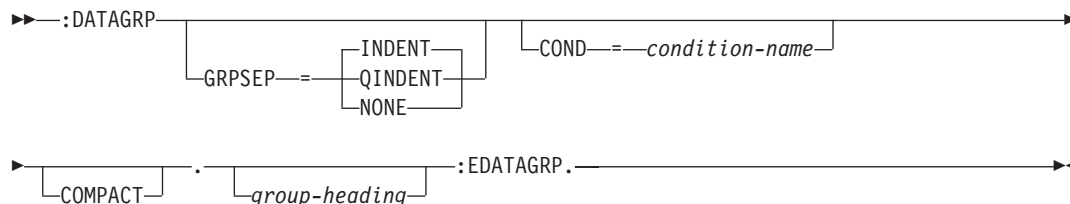
列見出しは左寄せされます。領域の DATACOL タグで列見出しテキストが指定されていない場合は、見出し情報のための行は画面上に確保されません。

DATAGRP (データ・グループ)

表示パネルの構文:



印刷パネルの構文:



データ・グループ (DATAGRP) タグは、データ表示域のデータ項目およびデータ選択フィールドをグループ化するために使用されます。このタグは、表示パネルおよび印刷パネルで許可されます。1 つのデータ・グループは、別のデータ・グループ内にネストすることができます。最外部グループを含めて最大 4 レベルのデータ・グループのネストが許可されます。このタグは、データ表示域 (DATA) タグで LAYOUT=HORIZ が指定されている場合は無効です。

UIM は、データ項目グループの保持処理を行います。データ・グループが他のデータ・グループ内にネストされている場合、すべてのグループに対して保持処理が行われます。外側のグループを分割する必要がある場合、そのグループ内にネストされているグループに対して保持処理が試行されます。グループ全体に必要な行数がレイアウト列に残っている行数を超える場合、グループ全体は強制的に次の列に送られます。単一行レイアウトの場合、グループ全体は、強制的に次のスクロール可能なページに送られます。1つのレイアウト列には大きすぎるデータ・グループは、1つの列の先頭で開始され、すべての項目に必要な数の列またはページまで継続されます。

オプションの属性

HELP=*help-module-name*

グループ内のすべての項目について説明するオンライン情報を示します。この属性は表示パネルにのみ許可されます。ヘルプ・モジュールの名前は別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説された名前の規則に従っていなければなりません。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

領域内のすべての項目には、それぞれ単一のヘルプ・モジュール名を関連付ける必要があります。DATA タグに HELP 属性が指定されている場合、ヘルプは領域内のすべてのグループおよび項目に適用されます。また、HELP 属性は、領域内のデータ・グループ (DATAGRP)、データ項目 (DATAI)、データ選択フィールド (DATASLT)、およびデータ選択フィールド選択項目 (DATASLTC) タグでは許可されません。

DATAGRP タグに HELP 属性が指定されている場合、ヘルプは、グループ内のすべての項目に適用されます。また、HELP 属性は、データ・グループ内の DATAI、DATASLT、DATASLTC、またはネストされた DATAGRP タグでは許可されません。

DATA タグ、外部 DATAGRP タグ、またはネストされた DATAGRP タグに HELP 属性を指定しない場合は、グループ中のすべての DATAI および DATASLT タグに HELP 属性が必要です。

NAME=*data-group-name*

グループに関連した名前。この名前は、Add Pop-Up Window (QUIADDPW) API とともに使用し、このデータ・グループの近くにウィンドウを配置することができます。この属性は表示パネルにのみ許可されます。

命名規則について詳しくは、515 ページの『名前構文』を参照してください。

GRPSEP=INDENT | NONE | QINDENT

グループ内で項目を区別するのに使用する視覚的な区切りのタイプを指定します。

INDENT は、それぞれのデータ項目、ネストされたデータ・グループのグループ見出し、またはグループ内のデータ選択フィールドについて、左端の列、プロンプト、または値を 2 桁字下げするのに使用されます。字下げされるのは左端の列だけです。グループ見出しは字下げされません。

QINDENT は、グループ内の最初のものを除く、それぞれのデータ項目またはネストされたデータ・グループまたはデータ選択フィールドを字下げするのに使用されます。このデータ項目グループ内の最初の項目または埋め込みグループは字下げされません。その他の項目、フィールド、および埋め込みグループはすべて字下げされます。データ項目またはデータ選択フィールドの、プロンプト列と値列の両方が 2 桁字下げされます。ネストされたデータ・グループのグループ見出しおよびそのグループ内のすべての項目は、2 桁字下げされます。

どのデータ項目、選択フィールドおよびグループを字下げするかに関する決定は、パネル・グループが作成され条件付けに影響を受けないときに行われます。このオプションは、制御言語 (CL) プロンプターが修飾名に使用するのと同じ形式で、データ項目を提示します。BIDI=RTL パネル・グループで

DATAGRP タグ

は、値を字下げする修飾名は後方にあるため、BIDI=RTL パネル・グループに QINDENT が指定されている場合、値列は字下げされません。最初の項目の後の各項目のプロンプト・テキストは、依然として字下げされます。

NONE は、データ項目、データ選択フィールド、またはグループ内でネストされたデータ項目グループをいずれも字下げしません。このオプションは、グループ見出しテキストが指定されている場合にはお勧めできません。これは、グループ見出しを使用せずに、関連データ項目、データ項目グループ、およびデータ選択フィールドを一緒にグループ化するのに使用できます。コンパクトな項目グループと、非コンパクト域にネストされたグループを定義することによって、これを行います。

COND=*condition-name*

グループは、指定された条件が真の場合にのみ表示または印刷されます。グループ内のすべてのタグが表示されるのは、このデータ・グループが表示される場合のみです。条件は、パネル・グループ・プロログ内で条件定義 (COND) タグを使用して定義する必要があります。

COMPACT

COMPACT が指定されている場合、個別の項目またはネストされたデータ・グループ見出しの間に空白行は残されません。COMPACT が指定されている別のデータ・グループの内側にこのデータ・グループがネストされている場合、指定の有無にかかわらず、COMPACT がこのデータ・グループに対して暗黙指定されます。

オプションのテキスト

group-heading

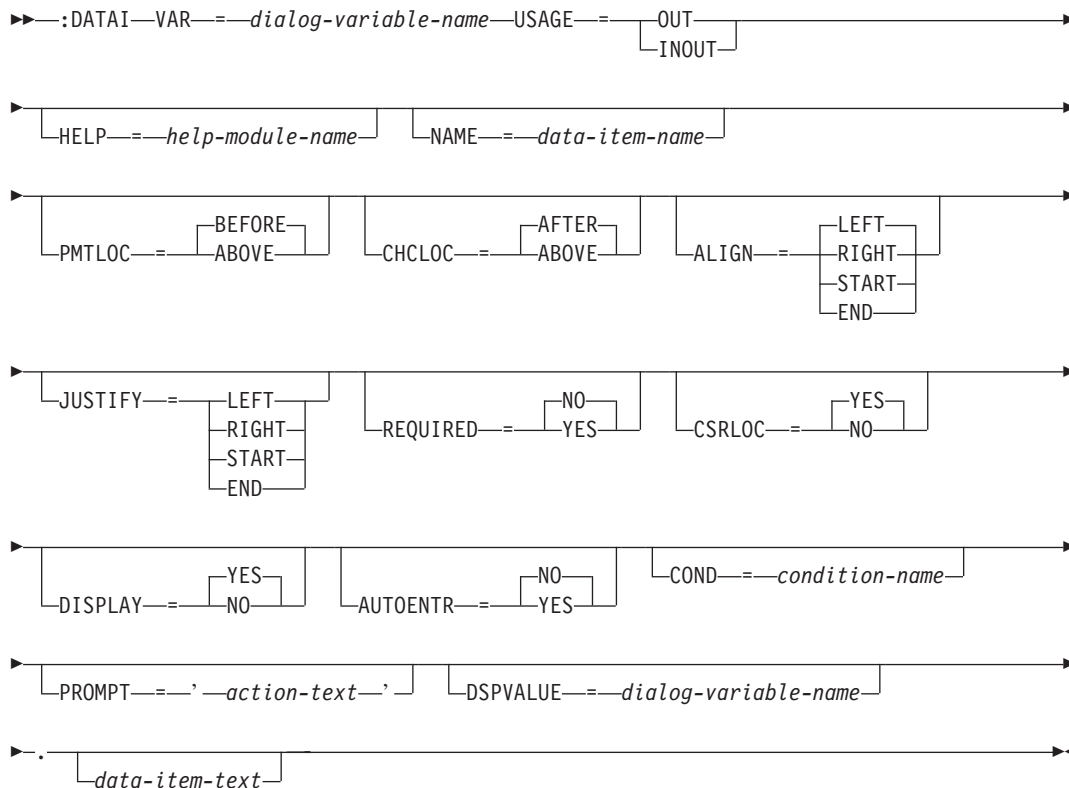
グループのデータ項目の上に置かれる見出し。テキストは、パネルの 1 行のみを占めますが、いくつかの列にまたがる場合があります。テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみを含めることができます。

テキストが省略されると、画面上にグループ見出し用の行は割り振られません。しかし、DATAGRP タグと EDATAGRP タグの間にある項目は、画面および他の UIM 操作上での保持処理に関してグループとして処理されます。

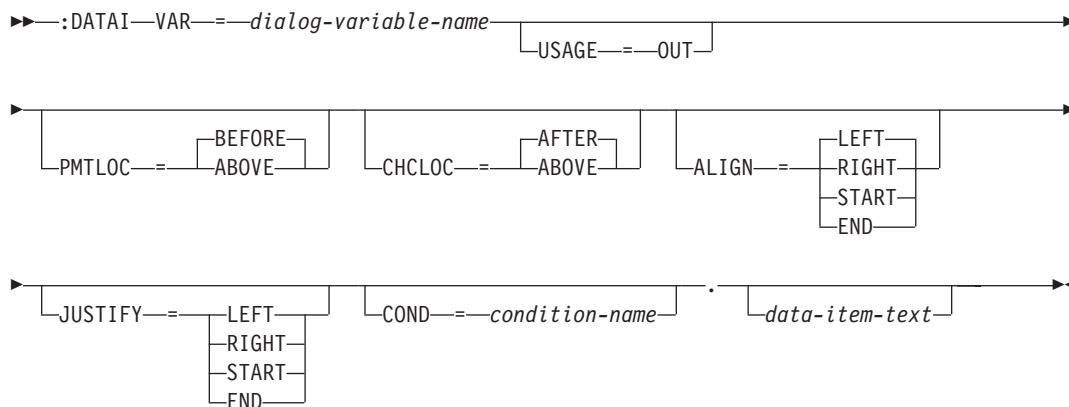
コロンは、それがまだ指定されておらず、その余地がある場合にテキストの最後に追加されます。

DATAI (データ項目)

表示パネルの構文:



印刷パネルの構文:



データ項目 (DATAI) タグは、データ表示域に配置される項目を定義します。このタグは、表示パネルおよび印刷パネルで許可されます。

データ項目エクステンダー (DATAIX) タグを使用して、データ項目の一部として表示される追加のダイアログ変数を指定できます。データ選択項目 (DATAC) タグを使用して、データ項目の一部として表示される選択項目候補のテキストを指定できます。 DATAIX および DATAC タグは、書式設定、条件付け、スクロール、およびヘルプに関連して、 DATAI タグの一部として扱われます。

DATAI タグ

プロンプト、フィールドとエクステンダー、および選択項目候補のテキストは、データ域に収まる必要があることに注意してください。 UIM では、スクロール中に個別のデータ項目を分割することはできません。

必須属性

VAR=*dialog-variable-name*

データ項目を構成するダイアログ変数の名前。ダイアログ変数の現行値が表示または印刷されます。

データ表示域 (DATA) タグで LAYOUT=1 が指定されている領域に項目が現れる場合、その値は選択項目列の幅よりも長くなることがあります。この場合、値は必要な数の表示行に折り返される単一のフィールドとして表示されます。 DATA タグに LAYOUT=2 が指定されている場合、値の幅は、2 番目のデータ列 (DATACOL) タグによって決まる開始位置とレイアウト列の終了位置との間に収まる必要があります。

LAYOUT=HORIZ の指定されたデータ表示域では、表示行にデータ項目が既に存在して、プロンプトと値の合計長が現在行に収まらない場合、5 つの列分離子がデータ項目間に許容されれば、プロンプトおよび値は次の表示行に示されます。水平レイアウトでは、プロンプトと値の合計がパネルの単一行に収まる必要があります。

USAGE=OUT | INOUT

データ項目の表示使用方法を示します。この属性は表示パネルでは必須で、印刷パネルではオプションです。

USAGE=OUT は、出力データ項目を定義します。 OUT は、表示される変数が出力専用であり、ユーザーはこれを変更できないことを示します。

USAGE=INOUT は、データ入力項目を定義します。 INOUT は、変数がデータ入力用であり、ユーザーが変更できることを示します。

オプションの属性

HELP=*help-module-name*

データ項目について説明するオンライン情報を示します。この属性は表示パネルにのみ許可されます。ヘルプ・モジュールの名前は、別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説した名前の規則に従っていなければなりません。名前の規則については詳しくは、515 ページの『名前構文』を参照してください。

領域内のすべての項目には、それぞれ単一のヘルプ・モジュール名を関連付ける必要があります。HELP 属性が DATA タグに指定されている場合、ヘルプは領域内のすべてのグループおよび項目に適用されるので、HELP 属性を領域内のいずれかのデータ・グループ (DATAGRP) および DATAI タグに指定することはできません。

HELP 属性が DATAGRP タグに指定されている場合、ヘルプはグループ内のすべてのデータ項目に適用されるので、HELP 属性をグループ内の DATAI タグに指定することはできません。

DATA タグまたはデータ項目を含む DATAGRP タグに HELP 属性を指定しない場合は、DATAI タグに HELP 属性が必要です。

ユーザーが LAYOUT=HORIZ データ域に関するオンライン情報にアクセスしやすくするには、水平域の個々の DATAI タグに HELP 属性を使用する代わりに、DATA タグに HELP 属性を使用します。

HELP 属性は、このデータ項目に関連したすべての DATAIX タグに適用されます。

NAME=*data-item-name*

項目に関連した名前。この名前を Add Pop-Up Window (QUIADDPW) API と共に使用して、ウィンドウをこのデータ項目の近くに配置することができます。この属性は表示パネルにのみ許可されます。

命名規則について詳しくは、515 ページの『名前構文』を参照してください。

PMTLOC=BEFORE | ABOVE

変数値の配置を、データ項目のプロンプト・テキストとの関係で決めます。

BEFORE は、プロンプト・テキストが変数値の前 (左側) に配置されることを示します。変数値は、プロンプト・テキストが終了した行から開始します。

ABOVE は、プロンプト・テキストがデータ項目変数の上側に配置されることを示します。変数は、プロンプト・テキストの先頭から 2 スペース字下げされます。 **ABOVE** は、**LAYOUT=HORIZ** が **DATA** タグに指定されているときは使用できません。

ABOVE の結果としてプロンプト・テキストがデータ項目変数の上側に配置される場合、このタグの **ALIGN** 属性は書式設定に使用されません。

DATAIX タグ上の **NEWLINE=CALC** または **NEWLINE=YES** の結果として新規の行に配置されるすべてのデータ項目エクステンダーは、データ項目変数と同じ列から開始します。

CHCLOC=AFTER | ABOVE

データ選択項目テキストの配置を、データ項目の変数値との関係で決めます。

AFTER は、データ選択項目テキストが変数値の後に配置されることを示します。これは変数値の終了位置と同じ行、またはその下の行のどちらかです。

ABOVE は、データ選択項目テキストがデータ項目変数の上側に配置されることを示します。 **ABOVE** を使用できるのは、**PMTLOC=ABOVE** が指定されているときだけです。この属性は、**LAYOUT=HORIZ** が **DATA** タグに指定されているときは使用できません。

この属性は、**DATAI** タグの直後にある **DATAC** タグのデータ選択項目テキストの配置だけを決めます。 **DATAIX** タグの直後にある **DATAC** タグは、この属性の影響を受けません。

ALIGN=LEFT | RIGHT | START | END

変数の表示値が、2 番目の **DATACOL** タグによって定義された選択項目列の中でどのように配置されるかを決めます。

ALIGN=LEFT は、表示値の一番左の文字を選択項目列の左端に合わせて配置します。

ALIGN=RIGHT は、表示値の一番右の文字を選択項目列の右端に合せて配置します。 **ALIGN=RIGHT** は、**LAYOUT=HORIZ** が **DATA** タグに指定されているときは使用できません。クラス定義 (**CLASS**) タグの **WIDTH** 属性で設定された表示値の幅が選択項目列の幅よりも長い場合、**ALIGN=RIGHT** の機能は **ALIGN=LEFT** と同じになります。

START は **LEFT** の同義語であり、**END** は **RIGHT** の同義語です。

JUSTIFY=LEFT | RIGHT | START | END

ダイアログ変数をどのように編集して表示値に入れるかを決めます。この属性のデフォルトは、このタグの **ALIGN** 属性に定義された値と同じです。

JUSTIFY=LEFT の場合、ダイアログ変数は表示値の中で左寄せされて、先行ブランクは保持されます。

JUSTIFY=RIGHT の場合、ダイアログ変数は表示値の中で右寄せされて、末尾ブランクは除去されません。

START は **LEFT** の同義語であり、**END** は **RIGHT** の同義語です。

REQUIRED=NO | YES

この項目が必須かどうかを示します。この属性は表示パネルにのみ許可されます。 **NO** は、この項目が必須ではないことを示します。

DATAI タグ

YES は、表示上でこの項目が必須であり、それに応じてフィールドが強調表示されることを示します。YES が有効なのは、このタグ上でデータ項目に関して USAGE=INOUT が指定されているときだけです。

REQUIRED=YES がコーディングされる時、ユーザー入力に関する明示的な検査は行われません。ただし、REQUIRED=YES のとき、ユーザーがこのフィールドに何も入力しない場合でも、UIM は入力編集および妥当性検査の処理を行います。これにより、妥当性検査 (CHECK) タグを使用して、ユーザーが必須フィールドに必ずデータを入力するようにすることができます。

CSRLOC=YES | NO

UIM によってデフォルト・カーソルの位置決めが行われるときに、カーソルを入力フィールドに配置可能にするかどうかを示します。この属性は表示パネルにのみ許可されます。

YES は、カーソルが入力フィールドに配置されることを保障するものではありません。それは、UIM によってデフォルト・カーソルの位置決めが行われるときに、カーソルを入力フィールドに配置できることを UIM に示すだけです。

NO は、カーソルをそのフィールドに配置できないことを示します。CSRLOC=NO は、リスト域の上のデータ域にある配置先フィールドなどの入力フィールドに使用するためのものです。パネルが最初に表示されるとき、または位置決め操作が完了した後は、カーソルを位置決めフィールドに配置することはできません。CSRLOC=NO がコーディングされた場合でも、デフォルト・カーソルの位置決めが行われるときに UIM はカーソルをそのフィールドに配置できます。これは以下の状況で生じます。

- パネルの現行ページにあるすべての入力フィールドで、CSRLOC=NO がマーク付けされている。
- 入力フィールドが間違っマーク付けされている。
- アプリケーション・プログラムの制御によるカーソルの位置決めで、カーソルがそのフィールドに置かれるように指定されている。

CSRLOC 属性は、このデータ項目に関連したすべての DATAIX タグに適用されます。

DISPLAY=YES | NO

パネルが表示されるときに、フィールドの内容を表示するかどうかを示します。この属性は表示パネルにのみ許可されます。

YES は、そのフィールドが表示されることを示します。

NO は、そのフィールドが表示されないことを示します。DISPLAY=NO は、表示すべきでないパスワード・フィールドなどの入力フィールドのためのものです。

AUTOENTR=NO | YES

そのフィールドが、自動入力の入力フィールドであるかどうかを示します。この属性は表示パネルにのみ許可されます。自動入力の入力フィールドは、ユーザーがフィールドの最後の位置にブランクを含む文字を入力したときに、装置からホストに戻ります。これは、ユーザーが Enter キーを押した場合と同じ結果になります。

NO は、そのフィールドが自動入力フィールドではないことを示します。

YES は、そのフィールドが自動入力フィールドであることを示します。YES を指定した場合、USAGE=INOUT もこのタグに指定する必要があります。

UIM はその使用を制限しませんが、AUTOENTR=YES は幅が 1 文字の入力フィールドのためのものです。

COND=*condition-name*

指定された条件が真である場合のみ、その項目は表示上有効になります。関連するすべての DATAC

および DATAIX タグは、このデータ項目が表示に含まれる場合にだけ表示されます。条件は、パネル・グループ内で条件定義 (COND) タグを使用して定義する必要があります。

PROMPT='action-text'

PROMPT ダイアログ・コマンドを介して F4=List が要求される時に取られるアクション。この属性は表示パネルにのみ許可されます。 **PROMPT** 属性が許可されるのは、このデータ項目に **USAGE=INOUT** が指定されているときだけです。 **action-text** の有効な形式を以下に示します。

- 'CALL program-reference'。 UIM とカーソル移動に影響されるプロンプトの出口プログラムとの間のインターフェースに関する説明は、iSeries Information Center で **API** のトピックを参照してください。
- 'RETURN positive-integer'

これらのアクションについての説明は、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

DSPVALUE=dialog-variable-name

このデータ項目に入力された現行データを含むダイアログ変数。この属性は、表示パネルに対してのみ、そして **PROMPT** 属性がこのタグにコーディングされている場合にのみ許可されます。この変数は、**VARUPD** 処理が実行されてもされなくても更新されます。変数プールは、機能キーの定義に使用される **VARUPD** 属性に基づいて更新されます。この更新は、表示値の変数の処理とは別に行われます。

指定されるダイアログ変数は、その長さがこのタグの **VAR** 属性に指定されたダイアログ変数の幅と等しい、**CHAR** または **IGC** 変数として **CLASS** タグ上で定義する必要があります。この変数に入れられる前に、変換リストの処理または値の検査が行われることはありません。このタグの **VAR** 属性に名前が指定された変数のクラスが、文字セットおよびコード・ページを変換する必要があることを指定している場合、この変数に関して文字セットおよびコード・ページが変換されます。

オプションのテキスト

data-item-text

データ項目について説明するテキスト。テキストは複数の行に表示されることがあります。また、テキスト反転 (RT) タグのみを含めることができます。

その項目がデータ項目グループにある場合、**DATAGRP** タグの **GRPSEP** 属性に指定されているように字下げされます。テキストが長すぎてプロンプト列に収まらない場合、それは項目列に収まるように必要に応じて追加の複数行にフォーマットされて、2 スペース字下げされます。

テキストが指定されていない場合、データ値がプロンプトなしで表示されます。プロンプト・テキストは、以下の特殊な場合を除いて、すべての **DATAI** タグに指定すべきです。

- データ項目が指示または項目の上に表示されたグループ見出しテキストによって完全に説明されているとき。
- データ項目に、パネル内で直前に示されたデータ項目の値から論理的に継続している値が含まれるとき。条件付けを使用して、使用可能な値の数だけの行をパネル上に表示できます。

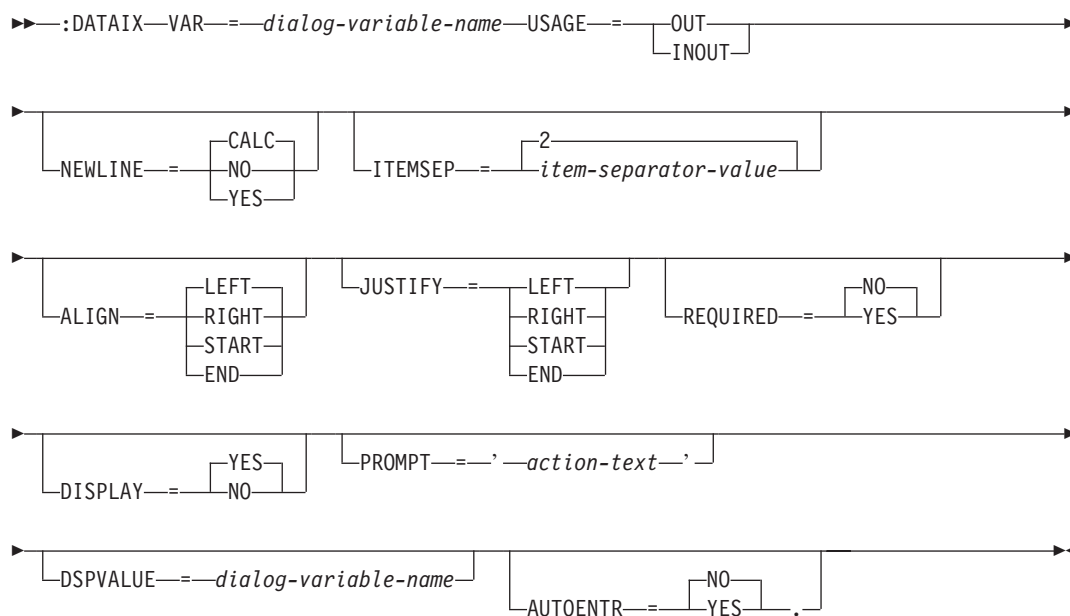
このタイプの情報を示す **DATAI** タグのセットを、常にデータ・グループ内に指定して、**UIM** が関連する値を表示上でまとめて示すこと、および適切な情報をパネルの全般ヘルプで提供することを確実にします。グループ内の最初のデータにプロンプト・テキストを指定するか、または **DATAGRP** タグに見出しテキストを指定してグループ全体について説明する必要があります。グループまたは領域全体に対して、ヘルプを指定してください。

DATAI タグ

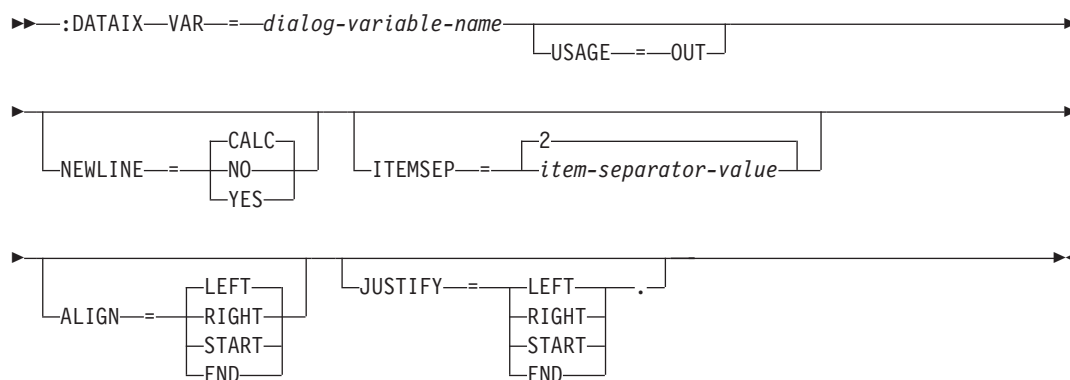
データ・グループが使用されないとき、UIM は項目をさまざまなレイアウト列で、さらにはさまざまなスクロール可能ページで表示することができます。複数のデータ項目に同じヘルプ・モジュールが指定される場合、パネルの全般ヘルプはそのパネル上の項目ごとにオンライン情報を繰り返します。

DATAIX (データ項目エクステンダー)

表示パネルの構文:



印刷パネルの構文:



データ項目エクステンダー (DATAIX) タグは、データ項目 (DATAI) タグに関連付ける追加のダイアログ変数を提供します。このタグは、表示パネルおよび印刷パネルで許可されます。

DATAIX タグは、対応する DATAI タグの後に出現する必要があります。複数の DATAIX タグを DATAI タグの後に指定して、複数のダイアログ変数を提供することもできます。DATAIX タグを入力しない場合、DATAI タグで指定された 1 つのダイアログ変数だけが、そのデータ項目に対して出現します。

DATAIX タグとデータ選択項目 (DATAIC) タグは、任意の順序で指定できます。それらを指定する順序により、それらが DATAI タグのダイアログ変数との関係で出現する位置が決まります。

必須属性

VAR=dialog-variable-name

直前の DATAI タグと共に表示された、追加のダイアログ変数の名前。ダイアログ変数の現行値が表示または印刷されます。

LAYOUT=1 データ表示域では、表示値がパネルの幅よりも長くなるのが可能です。この場合、値は必要な数の行に折り返される単一のフィールドとして表示されます。LAYOUT=2 データ表示域では、表示値が 1 行に収まる必要があります。

LAYOUT=HORIZ データ表示域では、データ項目のプロンプト、値、およびすべての DATAIX 値が、パネルの単一行に収まる必要があります。パネルの 1 つの行にすでにデータ項目がある場合、そしてプロンプト・テキストと DATAI タグのダイアログ変数にこのダイアログ変数の値を加えたものが現在行に収まらず、データ項目の間に 5 列の区切り記号が許可される場合には、項目エクステンダーを含むデータ項目全体がパネルの次の行に置かれます。

USAGE=OUT | INOUT

データ項目の表示使用方法。この属性は表示パネルでは必須ですが、印刷パネルではオプションです。

USAGE=OUT は、出力データ項目を定義します。OUT は、表示される変数が出力専用であり、ユーザーはこれを変更できないことを示します。

USAGE=INOUT は、データ入力項目を定義します。INOUT は、変数がデータ入力用であり、ユーザーが変更できることを示します。

オプションの属性

NEWLINE=CALC | NO | YES

データ項目エクステンダーの変数の配置を、データ項目の変数値、選択項目候補のテキスト、および事前定義されているデータ項目エクステンダーとの関係で決めます。

NEWLINE=CALC は、UIM が変数の配置を決めることを示します。以下のすべての条件が真である場合、変数は直前のダイアログ変数と同じ行に配置されます。

- 変数が、現在行の 2 番目のデータ列 (DATAICOL) タグで定義された値列の中に収まる。
- DATAIX タグの直後に DATAIC タグがない。
- 現在行は、直前の行から折り返されたものではない。

上記の条件のいずれかが真ではない場合、データ項目エクステンダーは次の表示行または印刷行に置かれます。それは、このタグのALIGN 属性に応じて配置されます。

NEWLINE=CALC が LAYOUT=HORIZ の指定されたデータ表示域に指定されているとき、値は直前の値と同じ行に置かれます。

NEWLINE=NO は、値が直前の値と同じ行で、その右側に置かれる必要があることを示します。直前の値は、DATAI タグまたは直前の DATAIX タグのどちらかのダイアログ変数です。DATAIX タグが DATAIC タグの直後に出現する場合、NEWLINE=NO は許可されません。

NEWLINE=YES は、値が次の表示行または印刷行に置かれる必要があることを示します。NEWLINE=YES は、LAYOUT=HORIZ が指定されたデータ表示域では許可されません。

DATAIX タグ

ITEMSEP=2 | *item-separator-value*

このデータ項目エクステンダーのダイアログ変数と直前のダイアログ変数とが、いくつかのスペースによって分離されるかを示します。

LAYOUT=1 または LAYOUT=2 が指定されたデータ表示域では、この属性の最大値は 2 番目の DATACOL タグで定義された選択項目列の幅です。LAYOUT=HORIZ が指定されたデータ表示域では、この属性の最大値は 5 です。デフォルト値は 2 で、最小値は 1 です。

ITEMSEP 属性は、NEWLINE=YES がこのタグに指定されているときには許可されず、NEWLINE=CALC が指定されてエクステンダー変数が新規の行に置かれているときには無効となります。

ALIGN=LEFT | RIGHT | START | END

表示値が、2 番目の DATACOL タグによって定義された選択項目列の中でどのように配置されるかを決めます。

ダイアログ変数の前に DATAI タグのダイアログ変数または直前の DATAIX タグがある場合、ALIGN 属性は選択項目列の中での表示値の位置に影響を与えません。

そのダイアログ変数が表示行または印刷行での最初の変数である場合、ALIGN 属性はテキストを以下のように書式設定します。

- ALIGN=LEFT は、表示値の一番左の文字を選択項目列の左端に合わせて配置します。
- ALIGN=RIGHT は、表示値の一番右の文字を選択項目列の右端に合せて配置します。表示値の幅が選択項目列の幅よりも長い場合、ALIGN=RIGHT の機能は ALIGN=LEFT と同じになります。

ALIGN=RIGHT は、LAYOUT=HORIZ が指定されたデータ表示域では許可されません。

START は LEFT の同義語であり、END は RIGHT の同義語です。

JUSTIFY=LEFT | RIGHT | START | END

ダイアログ変数をどのように編集して表示値に入れるかを決めます。この属性のデフォルトは、このタグの ALIGN 属性に指定された値、またはその属性に関してデフォルトとなる値と同じです。

JUSTIFY=LEFT の場合、ダイアログ変数は表示値の中で左寄せされます。先行ブランクは保持されません。

JUSTIFY=RIGHT の場合、ダイアログ変数は表示値の中で右寄せされて、末尾ブランクは除去されます。

START は LEFT の同義語であり、END は RIGHT の同義語です。

UCS-2 として定義された変数では、JUSTIFY 属性は無視されます。

REQUIRED=NO | YES

NO は、この項目が必須ではないことを示します。この属性は表示パネルにのみ許可されます。

YES は、表示上でこの項目が必須であり、それに応じてフィールドが強調表示されることを示します。YES が有効なのは、データ項目エクステンダーに関して USAGE=INOUT が指定されているときだけです。

REQUIRED=YES がコーディングされる時、ユーザー入力に関する明示的な検査は行われません。

REQUIRED=YES のとき、ユーザーがこのフィールドに何も入力しない場合でも、UIM は入力編集および妥当性検査の処理を実行します。これにより、妥当性検査 (CHECK) タグを使用して、ユーザーが必須フィールドに必ずデータを入力するようにすることができます。

DISPLAY=YES | NO

パネルが表示されるときに、フィールドを表示するかどうかを示します。この属性は表示パネルにのみ許可されます。YES は、そのフィールドが表示されることを示します。

NO は、そのフィールドが表示されないことを示します。DISPLAY=NO は、表示すべきでないパスワード・フィールドなどの入力フィールドのためのものです。

PROMPT='action-text'

PROMPT ダイアログ・コマンドを介して F4=List が要求される時に取られるアクション。この属性は、表示パネルに対してのみ、そしてこのデータ項目エクステンダーに関して USAGE=INOUT がこのタグにコーディングされている場合にのみ許可されます。アクション・テキストの有効な形式を以下に示します。

- 'CALL program-reference'。 UIM とカーソル移動に影響されるプロンプトの出口プログラムとの間のインターフェースに関する説明は、iSeries Information Center で API のトピックを参照してください。
- 'RETURN positive-integer'

これらのアクションについての説明は、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

DSPVALUE=dialog-variable-name

このデータ項目エクステンダーに入力される現行データを含むダイアログ変数。この属性は、表示パネルに対してのみ、そして PROMPT 属性がこのタグにコーディングされている場合にのみ許可されます。

この変数は、VARUPD 処理が実行されてもされなくても更新されます。変数プールは、機能キーの定義に使用される VARUPD 属性に基づいて更新されます。この更新は、表示値の変数の処理とは別に行われます。

指定されるダイアログ変数は、その長さがこのタグの VAR 属性に指定されたダイアログ変数の幅と等しい、CHAR または IGC 変数としてクラス定義 (CLASS) タグ上で定義する必要があります。

この変数に入れられる前に、変換リストの処理または値の検査が行われることはありません。このタグの VAR 属性に名前が指定された変数のクラスが、文字セットおよびコード・ページを変換する必要があることを指定している場合、この変数に関して文字セットおよびコード・ページが変換されます。

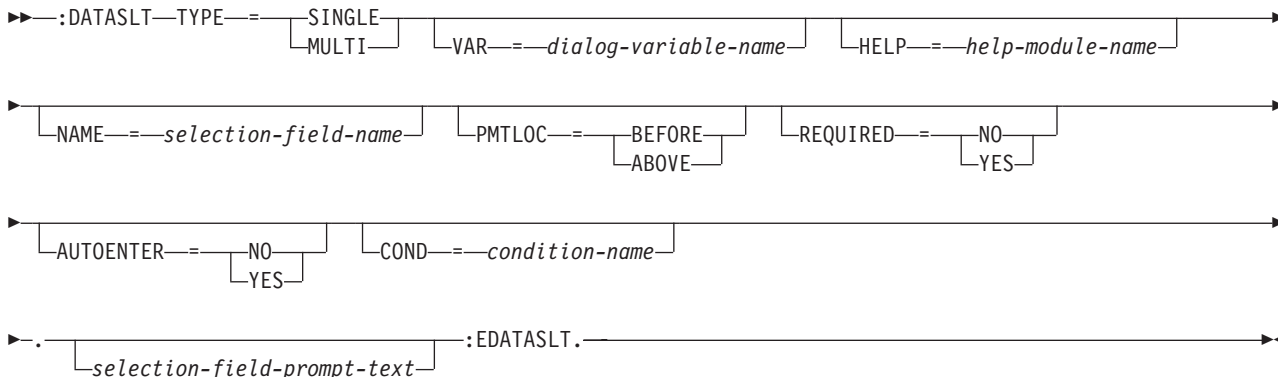
AUTOENTR=NO | YES

そのフィールドが、自動入力の入力フィールドであるかどうかを示します。この属性は表示パネルにのみ許可されます。自動入力の入力フィールドは、ユーザーがフィールドの最後の位置にブランクを含む文字を入力したときに、装置からホストに戻ります。これは、ユーザーが Enter キーを押した場合と同じ結果になります。

NO は、そのフィールドが自動入力フィールドではないことを示します。これはデフォルト値です。

YES は、そのフィールドが自動入力フィールドであることを示します。YES を指定した場合、USAGE=INOUT もこのタグに指定する必要があります。UIM はその使用を制限しませんが、AUTOENTR=YES は幅が 1 文字の入力フィールドのためのものです。

DATASLT (データ選択フィールド)



データ選択フィールド (DATASLT) タグは、データ表示領域の選択フィールドを定義します。このタグは、表示パネルでのみ許可されます。

DATASLT は単一または複数項目選択フィールドです。選択フィールドは内容と選択項目の数が固定されています。データ選択フィールドは、データ表示領域 (DATA) タグが LAYOUT=1 または 2 の場合にのみ指定できます。

データ選択フィールド選択項目 (DATASLTC) タグは、選択フィールドの選択項目を指定します。DATASLTC タグは、フォーマット、条件付け、スクロール、およびヘルプの点で DATASLT タグの一部です。

プロンプト、フィールド、および選択項目はデータ域内に収まらなければなりません。UIM では、スクロール時に選択フィールドが分割されるのを許可しません。

他のタグを DATASLT タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 60. DATASLT タグと EDATASLT タグの間で許可されるタグ

タグ名	順序	使用法	ページ
DATASLTC (データ選択フィールド選択項目)	1	D	573

必須属性

TYPE=SINGLE | MULTI

選択フィールドが単一選択項目または複数選択項目の選択フィールドかどうかを指定します。

SINGLE は、選択フィールドが単一選択項目のためのものであることを示します。

MULTI は、選択フィールドが複数選択項目のためのものであることを示します。

オプションの属性

VAR=*dialog-variable-name*

単一選択項目選択フィールドを構成するために使用されるダイアログ変数の名前。この属性は、TYPE=SINGLE の場合は必須です。TYPE=SINGLE の場合、変数は画面に表示され、その変数はクラス定義 (CLASS) タグで BASETYPE に BIN(31) を指定して宣言する必要があります。この変数には、ユーザーが選択した選択項目のオプション番号が含まれます。選択項目が選択されていない場合、変数はゼロを戻します。ダイアログ変数の現行値がディスプレイに表示されます。

TYPE =MULTI がこのタグで指定されている場合、この属性は許可されません。

HELP=*help-module-name*

データ選択フィールドの目的を説明するオンライン情報を示します。ヘルプ・モジュールの名前は、別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説した名前の規則に従っている必要があります。名前の規則について詳しくは、515 ページの『名前構文』を参照してください。

領域内のすべての項目には、それぞれ単一のヘルプ・モジュール名を関連付ける必要があります。HELP 属性が DATA タグで指定されている場合、ヘルプは領域内のすべてのグループと項目に適用され、HELP 属性は領域内のデータ・グループ (DATAGRP) および DATASLT タグでは許可されません。

HELP 属性が DATAGRP タグで指定されている場合、ヘルプはグループ内のすべてのデータ選択フィールドに適用され、HELP 属性はグループ内の DATASLT タグでは許可されません。

DATA タグまたはデータ選択フィールドを含む DATAGRP タグのいずれでも HELP 属性を指定しない場合は、DATASLT タグに HELP 属性が必要です。

NAME=*selection-field-name*

選択フィールドに関連した名前。この名前は後に、この選択フィールドに関連したウィンドウを配置するために、Add Pop-Up Window (QUIADDPW) API とともに使用することができます。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

PMTLOC=BEFORE | ABOVE

選択フィールドに関連した入力フィールドとの関連において、プロンプト・テキストの配置を制御します。

BEFORE は、プロンプト・テキストがデータ・フィールドの変数値の前 (左側) に配置されることを示します。変数値は、プロンプト・テキストが終了した行と同じ行で始まります。BEFORE はデフォルトです。

ABOVE は、プロンプト・テキストがデータ項目変数の上側に配置されることを示します。変数は、プロンプト・テキストの先頭から 2 スペース字下げされます。

REQUIRED=NO | YES

この属性は、TYPE=SINGLE の場合にのみ許可されます。NO は、フィールドが必須でないことを示します。NO がデフォルト値です。

YES は、フィールドが画面上で必須であり、適宜、強調表示されることを示します。REQUIRED=YES がコード化されると、ユーザー項目に対する明示的検査は行われません。しかし、YES を指定すると、ユーザーがフィールドに何も入力しなくても、UIM が入力編集および妥当性検査処理を実行します。これにより、妥当性検査 (CHECK) タグを使用して、ユーザーが必須フィールドに必ずデータを入力するようにすることができます。REQUIRED=YES は、TYPE=MULTI がこのタグで指定されているときには指定できません。

DATASLT タグ

AUTOENTR=NO | YES

フィールドが自動入力の入力フィールドかどうかを示します。自動入力の入力フィールドは、ユーザーが文字 (ブランクを含む) をフィールドの最後の位置に入力すると、デバイスからホストに画面を戻します。これは、ユーザーが Enter キーを押した場合と同じ結果になります。

NO は、フィールドが自動入力フィールドではないことを示します。これはデフォルト値です。

YES は、フィールドが自動入力フィールドであることを示します。UIM ではその使用を制限していませんが、AUTOENTR=YES は 1 文字幅の入力フィールドでのみ使用することが意図されています。YES は、TYPE=MULTI がこのタグで指定されているときには許可されません。

COND=condition-name

指定された条件が真の場合にのみ、選択フィールドが表示されます。この選択フィールドが画面に表示される場合に限り、関連した DATASLTC タグが表示されます。条件は、パネル・グループ内で条件定義 (COND) タグを使用して定義する必要があります。

オプションのテキスト

selection-field-prompt-text

選択フィールドを説明するテキスト。テキストは複数の行に表示されることがあります。また、テキスト反転 (RT) タグのみを含めることができます。テキストの長さは 255 バイト以下です。

選択フィールドがデータ項目グループに表示される場合、DATAGRP タグの GRPSEP 属性による指定に従って字下げされます。テキストが長すぎてプロンプト列に収まらない場合、列に収まるように、必要に応じてそれ以降の行でフォーマットされ、最初の行の先頭から 2 スペース字下げされます。

テキストが指定されていない場合、選択フィールドはプロンプトなしで表示されます。プロンプト・テキストはすべての DATASLT タグに指定する必要があります。ただし、項目の上部に表示されている指示またはグループ見出しテキストによって選択フィールドが完全に記述されている場合は除きます。

例 1: データ入力パネル

この例は、サンプル・データ入力パネルを示しています。パネルには、関数のダイアログ変数を事前に設定することにより、呼び出し側プログラムによってデフォルトが提供される 2 つの選択フィールドがあります。

UIM ソース

```
:class name=select
    basetype='bin 31'.
:var name=prtstyle
    class=select.
:var name=duplex
    class=select.
    :
:panel name=entry1
    help=hentry1
    topsep=space
    ...
    .Sample Entry Panel
:data depth='*'
    bodysep=both.
:topinst.Type choices, press Enter.
:datacol width=30.
:datacol width='*'.
:dataslt help=hstyle
    var=prtstyle
    type=single
    .Type style for printing
```

```

:datasltc help=hstylep
    option=1
    .Prestige elite (12 pitch)
:datasltc help=hstylec
    option=2
    .Courier (10 pitch)
:datasltc help=hstylees
    option=3
    .Essay standard (proportional)
:datasltc help=hstyleeb
    option=4
    .Essay bold (proportional)
:edataslt.
:dataslt type=single
    help=hduplx
    var=duplex
    .Duplex
:datasltc help=hduplxy
    option=1
    .Print both sides of paper
:datasltc help=hduplxn
    option=2
    .Print one side only
:edataslt.
:edata.
:epanel.
:invellip.

```

結果

```

Sample Entry Panel

Type choices, press Enter.

Type style for printing . . .  1 1. Prestige elite
                               2 2. Courier (10 pitch)
                               3 3. Essay standard (proportional)
                               4 4. Essay bold (proportional)

Duplex . . . . .  1 1. Print both sides of paper
                  2 2. Print one side only

```

この例では、*PRTSTYLE* および *DUPLEX* ダイアログ変数は BIN 31 でなければなりません。また、選択されたオプション値がこれらの変数で呼び出し側プログラムに戻されます。

例 2: 複数選択フィールド

この例は、複数選択フィールドを示しています。ユーザーは、スラッシュ (/) または各国指定の文字を、希望するオプションの前にある入力フィールドに入力することにより、値を選択します。複数の見出しオプションを選択できます。

UIM ソース

```

:class name=sfield
    basetype='char 1'.
:var name=hdoptsb
    class=sfield.
:var name=hdoptsu
    class=sfield.
:var name=hdoptsuc
    class=sfield.
:var name=hdoptssn
    class=sfield.
:var name=hdoptstm

```

DATASLT タグ

```
class=sfield.
:var name=hdoptsdt
class=sfield.
:panel name=xxx
key1=x1.Heading Options
:data depth='*'
bodysep=both.
:topinst.Select one or more
choices, press Enter.
:datacol width=30.
:datacol width='*'.
:dataslt type=multi
help=haward.Heading options
:datasltc var=hdoptsb
help=hbold.Bold
:datasltc var=hdoptsu
help=hundl.Underline
:datasltc var=hdoptsuc
help=hupperc.Uppercase
:datasltc var=hdoptssn
help=hsecl.Section numbers
:datasltc var=heoptstm
help=htime.Time
:datasltc var=hdoptsdt
help=hdate.Date
:edataslt.
:edata.
```

結果

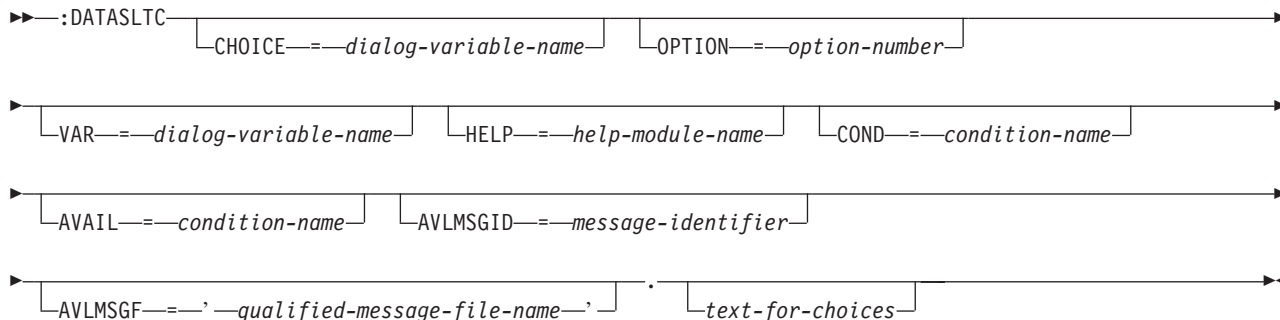
```
Heading Options                               System:  xxxxxxxx
Select one or more choices, press Enter.

Heading options . . . . .  _ Bold
                           _ Underline
                           _ Uppercase
                           _ Section numbers
                           _ Time
                           _ Date
```

この例でユーザーが *Underline*、*Section numbers*、および *Date* を選択する場合、ダイアログ変数の値は以下ようになります。

- hdoptsb=0
- hdoptsu=1
- hdoptsuc=0
- hdoptssn=1
- hdoptstm=0
- hdoptsdt=1

DATASLTC (データ選択フィールド選択項目)



データ選択フィールド選択項目 (DATASLTC) タグは、単一または複数項目選択フィールドの選択項目候補を定義します。このタグは、表示パネルでのみ許可されます。これは、領域のデータ表示域 (DATA) タグで LAYOUT=HORIZ が指定されている場合は無効です。

DATASLTC タグは、対応するデータ選択フィールド (DATASLT) タグの後ろに出現する必要があり、DATASLTタグと EDATASLTC タグの間には少なくとも 1 つの DATASLTC タグを指定する必要があります。DATASLTC タグは、選択フィールドの選択項目を定義します。

オプションの属性

CHOICE=dialog-variable-name

表示する選択項目候補のテキストを含むダイアログ変数の名前。ダイアログ変数は、テキストが一行に収まるように定義する必要があります。

ダイアログ変数は、クラス定義 (CLASS) タグで BASETYPE に CHAR、IGC、または BIN を指定して定義する必要があります。

ダイアログ変数のエラー状態は、テキストの強調表示の判別には使用されません。

CHOICE 属性が指定されている場合、text-for-choices を指定することはできません。

IGC 用の特殊フォーマット。(省略形 IGC は、コマンドおよびキーワードで使用され、2 バイト文字セット機能を表します。)BASETYPE が IGC のダイアログ変数を CLASS タグ上で指定すると、UIM は特殊フォーマット設定を行います。変数値がシフトアウト文字 (X'0E') で始まる場合は、UIM は値を左に 1 文字移動して、他の行のデータ選択項目と垂直位置合わせを行います。

OPTION=option-number

このオプションに割り当てられる番号。オプション番号は、1 から 99 の整数です。この属性は、TYPE=SINGLE が指定された DATASLT タグ内で指定する場合は必須ですが、TYPE=MULTI が指定されている場合は使用できません。

選択フィールド選択項目は、選択フィールドで定義した順序で表示されます。2 つの選択項目の番号が連続していない場合は、2 つの選択項目の間に空白行が自動的に挿入されます。2 つの選択項目のオプション番号が同じで、両方とも同時にオンに条件指定すると、最初に定義された選択項目が表示されます。

VAR=dialog-variable-name

選択フィールド選択項目が選択されているかどうかを示すためのダイアログ変数の名前。この属性は、TYPE=MULTI の場合は必須です。この変数は、CLASS タグで BASETYPE に CHAR 1 を指定して宣言する必要があります。

DATASLTC タグ

選択項目が選択されていない場合は、ダイアログ変数の値は '0' です。選択項目が選択されると、値 '1' がダイアログ変数に戻されます。画面が表示されているときに変数の現行値が '1' の場合は、スラッシュ (/) が画面に表示されます。ユーザーがスラッシュまたは国別指定文字を入力すると、ユーザーに '1' が戻されます。ユーザーが選択のためにどの文字を使用するとしても、パネルが再表示されるときにその文字が表示されます。パネルが表示されたときにダイアログ変数の現行値が '1' 以外の場合は、変数は未選択として表示されます。

VAR 属性は、DATASLT タグで TYPE=MULTI が指定されている場合は必須です。TYPE=SINGLE が指定されている場合は、VAR 属性を使用することはできません。

HELP=*help-module-name*

選択フィールド選択項目の目的を説明するオンライン情報を示します。ヘルプ・モジュールの名前は、別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説した名前の規則に従っている必要があります。名前の規則については、515 ページの『名前構文』を参照してください。

この属性を指定できるのは、この DATASLTC タグを含む DATASLT タグに HELP 属性が指定されている場合だけです。

選択フィールド中の 1 つの DATASLTC タグに HELP 属性を指定した場合は、その選択フィールド中のすべての DATASLTC タグに HELP 属性を指定しなければなりません。

DATASLT タグ上で SELECT=MULTI を指定した複数項目選択フィールドの場合、この属性で識別されるオンライン情報は、選択フィールド内の任意の位置にカーソルが置かれている場合に表示される文脈ヘルプの一部として組み込まれます。

DATASLT タグ上で SELECT=SINGLE を指定した単一項目選択フィールドの場合、この属性で識別されるオンライン情報は、このタグの選択テキスト上にカーソルがある間にヘルプが要求されたときに表示されます。このオンライン情報は、カーソルが選択フィールド内にあるが、フィールド内のいずれかの選択項目のテキスト上にはない場合に表示される文脈ヘルプの一部としても組み込まれます。これには、カーソルが選択フィールドのプロンプト・テキスト上にある場合、または選択フィールドの入力フィールドにある場合などがあります。カーソルが入力フィールドにあり、有効な選択項目が入力された場合、ヘルプが要求されたときにその選択項目のヘルプが表示されます。

COND=*condition-name*

指定された条件が真の場合にのみ、選択フィールドの選択項目が有効になります。条件は、パネル・グループ・プロローグ内で条件定義 (COND) タグを使用して定義する必要があります。選択項目が条件によってオフになると、その選択フィールド選択項目は選択フィールドに表示されず、その選択項目のヘルプはヘルプを表示するための要求に組み込まれなくなります。

AVAIL=*condition-name*

選択フィールド選択項目が使用可能かどうかを示す条件の名前。条件は、パネル・グループ・プロローグ内で COND タグを使用して定義する必要があります。

条件が真の場合、選択フィールド選択項目は使用可能です。条件が偽の場合、選択フィールド選択項目は使用できません。このタグの COND 属性で指定されている条件があれば、その条件がこの属性より優先して使用されます。

使用不可の選択項目は別の色で表示され、選択項目のオプション番号の最初の部分がアスタリスク (*) で表示されます。

選択フィールドが DATASLT タグで TYPE=MULTI として指定されている場合、AVAIL 属性を指定することはできません。

AVLMSGID=message-identifier

このタグの AVAIL 属性での指定によって、選択フィールド選択項目が使用可能でない場合に、その項目が選択されると表示されるメッセージのメッセージ ID。この属性を指定できるのは、AVAIL 属性が指定されている場合だけです。

この属性が指定されない場合、UIM は選択項目が現在使用可能でないことを告げるデフォルト・メッセージを表示します。

AVLMSGF='qualified-message-file-name'

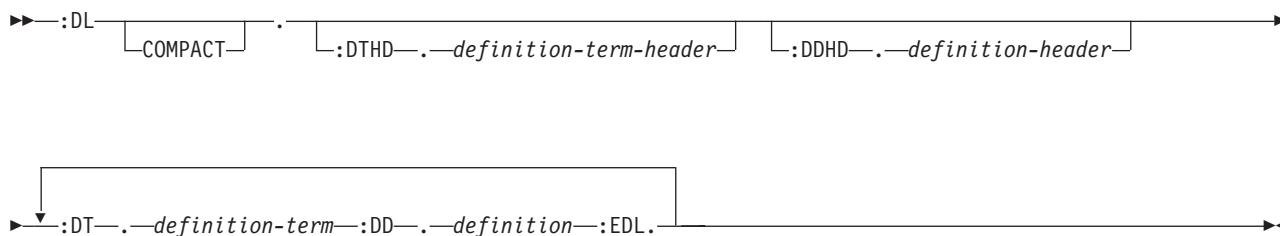
メッセージ ID が入っているメッセージ・ファイル名。この属性を指定できるのは、このタグに AVLMSGID 属性が指定されている場合だけです。パネル・グループ (PNLGRP) タグに DFTMSGF 属性が指定されていない場合で、このタグに AVLMSGID 属性が指定されている場合は、この属性を指定する必要があります。

オプションのテキスト

text-for-choices

このテキストは暗黙の段落です。表示のフォーマット時に、1 行に収まらないテキストは、2 行目以降に 2 桁字下げしてフォーマットされます。テキストは最大 255 文字で、テキスト反転 (RT) タグだけを含めることができます。このタグに CHOICE 属性が指定されていない限り、*text-for-choices* は必須です。

DL (定義リスト)



定義リスト (DL) は、語または句、およびそれに対応する定義、記述、または説明のリストです。このタグには、対応する終了タグが必要です。定義リストは、情報域およびヘルプ域でのみ許可されます。

定義される用語とその定義は、定義用語 (DT) タグおよび定義記述 (DD) タグによって識別されます。用語の列および定義の列に使用される見出しは、それぞれ定義用語見出し (DTHD) タグと定義記述見出し (DDHD) タグによって識別できます。

定義リストはテキスト内の任意の場所に出現できます。定義リストを他のリストまたは定義リスト内にネストすることも、他のリストを定義リスト内にネストすることもできます。

2 つの DT タグまたは 2 つの DD タグを連続して使用することはできません。

他のタグを DL タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

DL、DT、DD、DTHD、DDHD タグ

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 61. DL タグと EDL タグの間で許可されるタグ

タグ名	順序	使用法	ページ
FIG (図表)	1	B	577
LINES (不定形式行)	1	B	597
XMP (例)	1	B	697
NT (注記)	1	B	644
P (段落)	1	B	648
PC (段落の連結)	1	B	658
LP (リストの一部)	1	B	633
OL (番号付きリスト)	1	B	645
SL (単純リスト)	1	B	678
UL (順不同リスト)	1	B	690
PARML (パラメーター・リスト)	1	B	656
DL (定義リスト)	1	B	575

オプションの属性

COMPACT

項目の間に空白行を入れずにリストをフォーマットします。

必須タグ

DT.definition-term

定義用語のテキスト。用語は入力したとおりに、強調表示句 2 (HP2) でフォーマットされます。

用語は、1 バイト文字セット (SBCS) で 10 文字、または 2 バイト文字セット (DBCS) で 4 文字と想定され、左マージンに対する位置 2 から始まります。用語が想定されている文字数より長い場合、記述行に拡張されます。

DD.definition

定義用のテキスト。用語の記述は暗黙の段落であり、テキスト項目を含めることができます。段落 (P) タグを使用して、記述段落の後に追加の段落を挿入できます。

用語が記述領域に拡張されない限り、記述は左マージンに対する位置 14 から始まります。

オプションのタグ

DDHD.definition-header

定義ヘッダー用のテキスト。ヘッダーは入力したとおりに、強調表示句 2 (HP2) でフォーマットされます。

DTHD.definition-term-header

定義用語ヘッダー用のテキスト。ヘッダーは入力したとおりに、強調表示句 2 (HP2) でフォーマットされます。

例 1: 定義リスト

この例では、見出し付きの定義リストがフォーマットされる方法を説明します。

UIM ソース

```
:DL.
:DTHD.Term
:DDHD.Description
:DT.DL
:DD.This is a sample
definition list term
and description.
:DT.EDL
:DD.Here is another.
:EDL.
```

結果

用語	説明
DL	This is a sample definition list term and description.
EDL	Here is another.

例 2: 短縮定義リスト

これは、COMPACT 属性を使用した定義リストです。

UIM ソース

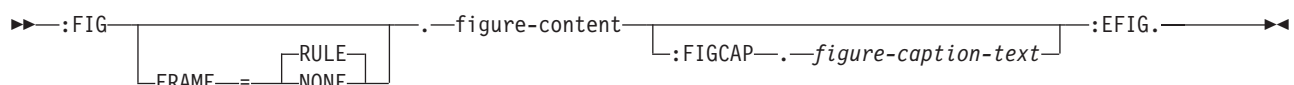
```
:DL compact.
:DT.COMPACT
:DD.This causes the list
to be compacted, so the
blank lines are removed.
:DT.ANOTHER
:DD.Here is another.
:EDL.
```

結果

COMPACT	This causes the list to be compacted, so the blank lines are removed.
ANOTHER	Here is another.

定義リスト内で不定形式行 (LINES)、図表 (FIG)、および例 (XMP) タグを使用するときには注意が必要です。なぜなら、1 行に収まらないテキストは次の行の列 1 に折り返されるからです。行および図表は現行左マージンから始まり、例は現行左マージンから 4 スペース字下げされます。ネストされたリストがフォーマットされると、現行左マージンは変わります。さまざまな位置 (リスト内を含む) に組み込まれた LINES、FIG、または XMP タグを含む情報源がある場合、それは毎回同じように見えるわけではありません。

FIG (図表)



FIG、FIGCAP タグ

図表 (FIG) タグは、ダイアグラム、グラフ、またはその他の図を識別します。このタグには、対応する終了タグが必要です。図表タグは情報域とヘルプ領域でのみ許可されます。また、図表には図表表題を含めることができます。これは、図表表題 (FIGCAP) タグで識別されます。

図表では自動編集機能がオフになります。そのため、テキストは入力されたとおりにフォーマットされません。図表は、入力された箇所で、画面またはウィンドウの幅全体にフォーマットされます。

リスト内の図表を使用するときには注意が必要です。なぜなら、1 行に収まらない図表テキストは次の行の列 1 に折り返されるからです。

図表は常に、列 2 に設定された左マージン付きでフォーマットされます。フォーマット幅は、ヘルプ・モジュールで指定された幅より 2 列小さくなります。FIG タグが情報域に表示される場合、最初のフォーマット幅は、パネル・タグで指定された幅より 2 バイト小さくなります。

図表表題は左マージンで始まります。図表タグまたは図表表題テキストのどちらかが 1 バイト文字であるには、図表タグ "Figure:" と図表表題テキストの間には 2 スペースがあります。図表タグと図表表題の両方が 2 バイト文字である場合には、両者の間には 4 スペースがあります。図表表題が 1 行に収まらない場合には、2 行目に折り返し、その行は左マージンからの位置 4 で始まります。

オプションの属性

FRAME=RULE | NONE

図表の周囲にあるフレームのタイプ。RULE がデフォルトです。図表の上下には、ハイフンの行が表示されます。

NONE はフレームを使用しないことを示します。図表の前後に空白行が置かれます。

オプションのタグ

FIGCAP.*figure caption text*

図表の表題 (必要な場合)。表題はソース内の複数行に表示されることがあります。FIGCAP タグは、EFIG タグの直前に表示される必要があります。

例: サンプル図表

この例は、FIG タグを使用して入力された仕方でテキストがフォーマットされる様子を示しています。さらに、FIG タグがフレームを指定して、テキストの他の部分から分けて設定する方法を示しています。

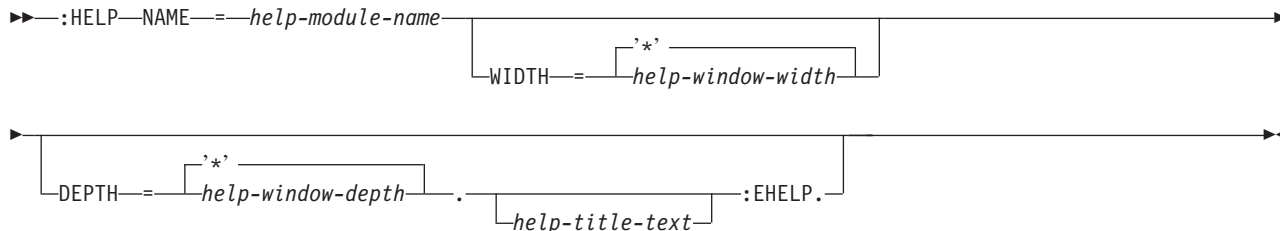
UIM ソース

```
:FIG frame=rule.  
some  
  sample  
  text  
:FIGCAP.A Sample Figure  
:EFIG.
```

結果

```
some  
  sample  
  text  
Figure: A Sample Figure
```

HELP (ヘルプ・モジュール)



ヘルプ・モジュール (HELP) タグは、ヘルプ・モジュールの先頭を示します。EHELP タグは、ヘルプ・モジュールの末尾に指定する必要があります。ヘルプ・モジュールは、ヘルプ表示および索引検索トピックに使用することができます。索引検索関数について詳しくは、588 ページの『ISCH (索引検索)』を参照してください。また、ヘルプ・モジュールを互いにリンクさせることができます。ヘルプ・モジュールを互いにリンクさせる方法について詳しくは、598 ページの『LINK (ハイパーテキスト・リンク定義)』を参照してください。

HELP 属性を持つすべてのタグは、このモジュールを参照できます。

必須属性

NAME=help-module-name

このヘルプ・モジュールの名前。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

この名前は以下から参照できます。

- UIM パネル定義タグの HELP 属性。
- インポート (IMPORT) タグの NAME 属性。
- 別のヘルプ・モジュール内の組み込みヘルプ (IMHELP) タグの NAME 属性。
- 索引検索サブトピック (ISCHSUBT) タグの TOPICS 属性。
- データ記述仕様 (DDS) 仕様の HLPPNLGRP キーワード。
- 制御言語 (CL) コマンド定義。
- Display Help (QUHDSPH) API のヘルプ ID 配列パラメーター。
- DSPHELP ダイアログ・コマンド。

NAME 属性の値の最大長は 32 文字です。名前では、区切り文字または区切り記号としてスラッシュ (/) を使用できます。コマンドのヘルプ・モジュールに名前を付ける際には、スラッシュを使用して、コマンド名とパラメーター名を区切ります。スラッシュが使用される場合、名前はアポストロフィ (') で囲む必要があります。ヘルプ・モジュールはネストできません。

オプションの属性

WIDTH=*' | help-window-depth

ヘルプ・ウィンドウが表示されるとき幅。WIDTH 属性に指定される値には、右と左のマージン用に 2 バイトのスペースが含まれている必要があります。最小幅は 32 バイトで、最大幅は 74 バイトです。

HELP タグ

フルスクリーンまたは拡張ヘルプ・ウィンドウとして表示されるヘルプ・モジュールは、4 バイト字下げされます。そのため、フォーマット幅は、幅より 6 バイト小さくなります。たとえば、最大幅の 74 バイトでフォーマットされたヘルプ・モジュールのフォーマット幅は 68 バイトになります。

'*' 以外の値が指定されている場合、UIM はウィンドウの幅の増減を行いません。HELP タグと EHELP タグの間のテキストが特定の幅のウィンドウに依存する場合にのみ、幅を指定する必要があります。

ヘルプが機能キーまたはリスト域に関するものであり、WIDTH='*' の場合、UIM は幅 38 バイトを使用します。ヘルプがメニュー域またはデータ域に関するものである場合、UIM は幅 60 バイトを使用します。UIM はこの幅を使用してウィンドウの位置を見付け、最も適切な方法でオンライン情報を表示するサイズになるまで、ウィンドウを拡張します。

使用される幅は、UIM がオンライン・ヘルプ情報を集めるときに最初に検出した HELP タグの幅です。

DEPTH='*' | help-window-depth

ヘルプ・ウィンドウが表示されるときの高さ。高さ指定される値は 9 行より小さくすることはできません。また、以下を含んでいる必要があります。

- ヘルプ・タイトル用に 1 行。
- 上部区切り行用に 1 行。
- 下部区切り記号用に 1 行。
- 機能キー用に 2 行。
- メッセージ行用の 1 行。

ポップアップ・ウィンドウに表示されるヘルプの場合、DEPTH 属性の最大値は 21 です。

'*' 以外の値が指定されている場合、UIM はウィンドウの高さの増減を行いません。HELP タグと EHELP タグの間のテキストが特定のウィンドウの幅に依存する場合に限り、固有の高さを指定する必要があります。

ヘルプがデータ域、メニュー域、または機能キーに関するものである場合、および DEPTH='*' の場合、UIM は高さ 9 を選択します。ヘルプがリスト域に関するものである場合、UIM は高さ 14 を選択します。UIM はこの高さを使用してウィンドウの位置を見付け、最も適切な方法でオンライン・ヘルプ情報を表示するように設計されたサイズになるまで、ウィンドウを拡張します。

使用される高さは、UIM がオンライン・ヘルプ情報を集めるときに最初に検出した HELP タグの高さです。

オプションのテキスト

help-title-text

ヘルプ・パネルで使用されるタイトル。テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみを含めることができます。help-title-text がポップアップ・ウィンドウの幅より長い場合、タイトル・テキストはポップアップ・ウィンドウの幅に切り捨てられます。このタグの WIDTH 属性の値が 57 より大きいか、または '*' と等しい場合、help-title-text の最大長は 55 文字です。そうでない場合、最大長は WIDTH 属性に指定された値から 2 を引いたものです。

表示されるヘルプ・パネル・タイトルは、文脈ヘルプまたは拡張ヘルプが要求されるかどうか、およびヘルプがフルスクリーンとして表示されるか、またはポップアップ・ウィンドウに表示されるかどうかによって決まります。

ポップアップ・ウィンドウに表示されるヘルプ情報には、次のようにしてタイトルが付けられます。

- 文脈ヘルプの場合、関連項目の HELP タグのタイトルが使用されます。HELP タグでタイトルが定義されていない場合、デフォルトのタイトルである "Help" が使用されます。
- 拡張ヘルプの場合、パネルの最初の HELP タグのタイトルが使用されます。HELP タグでタイトルが定義されていない場合、デフォルトのタイトルまたは QUHDSPH API の全表示タイトルが使用されます。

ヘルプ・タイトルのテキストがポップアップ・ウィンドウの幅より長い場合、タイトル・テキストはワード境界でポップアップ・ウィンドウの幅に切り捨てられます。デフォルトのタイトルまたは QUHDSPH API の全表示タイトルが使用されます。

フルスクリーン・モードで表示されるヘルプ情報には、次のようにしてタイトルが付けられます。

- 文脈ヘルプの場合、HELP タグのタイトルが使用されます。
- 拡張ヘルプの場合、最初の HELP タグのタイトルが使用されます。
- HELP タグでタイトルが定義されていない場合、デフォルトのタイトルまたは QUHDSPH API の全表示タイトルが使用されます。

ヘルプ・モジュールは、各モジュールを個別にフォーマットできるように、コード化される必要があります。ヘルプ・モジュールは以下のタグのいずれかで始まらなければなりません。

DL	定義リスト
FIG	図表
H1-H4	見出し
IMHELP	組み込みヘルプ
ISCH	索引検索
LINES	不定形式行
NT	注記
OL	番号付きリスト
P	段落
PARML	パラメーター・リスト
SL	単純リスト
UL	順不同リスト
XH1-XH4	拡張ヘルプの見出し
XMP	例

例: ヘルプ・パネル定義

この例は、ヘルプ・パネルの定義方法およびテキストの表示方法を示しています。

UIM ソース

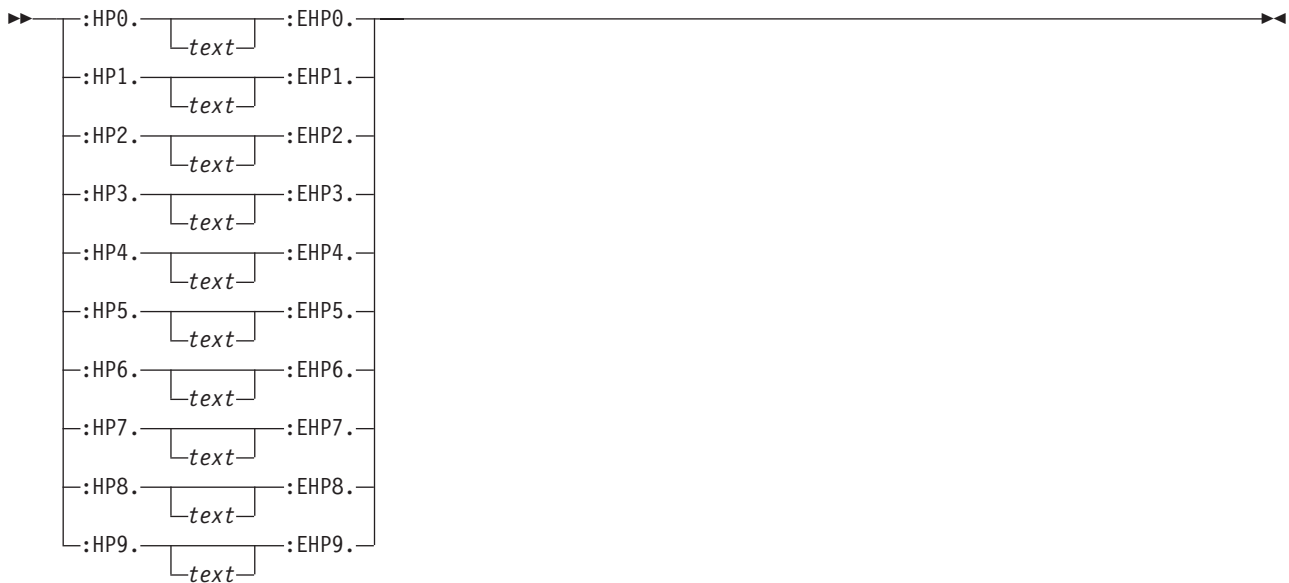
```
:help name=hmain.Main System Menu - Help
:p.
This panel allows you to ...
  (extended description of the panel)
:ehelp.
```

HELP タグ

結果

```
HELP                                Main System Menu  
This panel allows you to ...
```

HP0 から HP9 (強調表示句)



これらの強調表示句 (HP0-HP9) タグは、強調表示される語または句を識別します。すべての HP n タグには対応する終了タグが必要です。これらのタグは、印刷ヘッド (PRTHEAD) または印刷パネル (PRTPNL) タグ内にはないヘルプ域および情報域でのみ許可されます。

HP n および EHP n タグの句は、ワード境界で指定する必要があります。EHP n タグの直後に続く 2 文字が句読記号とブランクである場合、UIM は強調属性を自動的に拡張して、句読記号を含めます。これにより、句読記号とそれに関連したテキストを、同じ強調を使用して表示できます。

強調表示タグはネストできます。たとえば、:HP1.text:HP2.text2:EHP2.text3:EHP1. は有効です。

オプションのテキスト

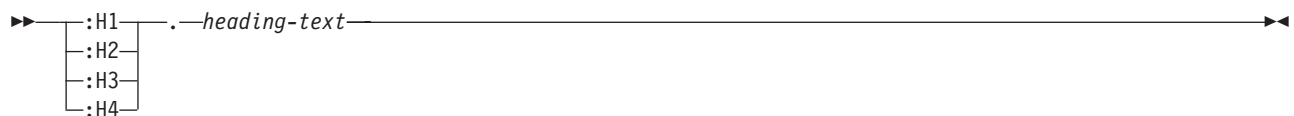
text

強調表示される語または句は必須ではありません。ただし、タグはテキストが指定されていないときには意味を持ちません。

タグ	カラー装置での書式設定	モノクローム装置での書式設定	ヘルプ用の印刷の書式設定
HP0	緑	通常	通常
HP1	緑、下線	下線	下線付き
HP2	白	高輝度	太字
HP3	白、下線	高輝度、下線	太字、下線
HP4	緑	通常	通常
HP5	緑、反転表示	通常、反転表示	下線付き
HP6	緑、下線、反転表示	下線、反転表示	下線付き
HP7	白、反転表示	高輝度、反転表示	太字、下線
HP8	白、反転表示	高輝度、反転表示	太字、下線
HP9	白、反転表示	通常、反転表示	下線付き

RV2W062-0

H1 から H4 (見出し)



見出し (H1-H4) タグは、情報のメイン・トピックおよびサブトピックを識別します。これらのタグは、情報域とヘルプ領域でのみ許可されます。

ヘルプ領域では、見出しとテキストを区切るために、見出しの後のテキストがマージンから 4 スペース字下げされます。情報域では、テキストは左マージンに対して左揃えにフォーマットされます。

見出しでは、見出しテキストの前後に 1 行の空白行がフォーマット設定されています。

個々のフォーマット規則は以下のとおりです。

H1 テキストを下線付きの強調表示されたテキストとして中央揃えし、フォーマット設定します (HP3 タグと同様)。このタグを設定すると、印刷されたヘルプ・モジュールで表示されるときにページ排出が行われます。

H1 から H4 タグ

- H2 テキストを下線付きの強調表示されたテキストとして左揃えし、フォーマット設定します (HP3 タグと同様)。
- H3 テキストを強調表示されたテキストとして左揃えし、フォーマット設定します (HP2 タグと同様)。
- H4 テキストを下線付きのテキストとして左揃えし、フォーマット設定します (HP1 タグと同様)。

必須テキスト

heading-text

見出し用のテキスト。見出しのテキストは単一行に入力する必要があり、入力したとおりにフォーマット設定されます。その他のタグを含めることはできません。一般的な方法は、意味のある語の最初を大文字にし、現行の発行スタイルに従ってテキストを入力することです。

例: 見出しタグ

この例は、さまざまな見出しが位置調整されて、フォーマットされる様子を示します。

UIM ソース

```
:H1.A One Heading
:p.Here's a paragraph.
:H2.A Two Heading
:p.Another paragraph.
:H3.A Three Heading
:p.Still another paragraph.
:H4.A Four Heading
:p.Still another paragraph.
```

結果

A One Heading

Here's a paragraph.

A Two Heading

Another paragraph.

A Three Heading

Still another paragraph.

A Four Heading

Still another paragraph.

IMHELP (組み込みヘルプ)

▶▶:IMHELP—NAME—=*help-module-name*—▶▶

- | 組み込みヘルプ (IMHELP) タグは、ヘルプ・モジュール内にヘルプ・モジュールを組み込みます。ヘルプ・モジュールはヘルプ (HELP) タグで始まり、ヘルプ終了 (EHELP) タグで終わります。

必須属性

NAME=*help-module-name*

このタグが出現する、組み込みヘルプ・モジュールの名前。命名規則については、515 ページの『名前構文』を参照してください。名前は、同じパネル・グループ内で定義された別のヘルプ・モジュールの名前であるか、またはインポート (IMPORT) タグを使用して別のパネル・グループからインポートされる必要があります。IMHELP タグのネスト限度は 16 です。

索引検索 (ISCH) または索引検索同義語 (ISCHSYN) タグが組み込みヘルプ・モジュールの一部である場合、それらは IMHELP タグが出現するモジュールの一部ではありません。

アクティブな強調表示句 (HP*n*) タグは、IMHELP タグがコード化されているときには許可されません。IMHELP タグをコード化する前に、アクティブな HP*n* タグが EHP*n* タグで終わっている必要があります。

- | 組み込みヘルプ (IMHELP) タグは、以下のタグとともに組み込むことができます。
- | • 定義リスト (DL)。IMHELP タグの前には DD (定義) タグ、または DD タグのテキストが置かれている必要があります。
- | • 組み込みヘルプ (IMHELP)。
- | • 注記 (NT)。
- | • 番号付きリスト (OL)。IMHELP タグの前には LI (リスト項目) タグ、または LI タグのテキストが置かれている必要があります。
- | • パラメーター・リスト (PARML)。IMHELP タグの前には PD (パラメーター定義) タグ、または PD タグのテキストが置かれている必要があります。
- | • 単純リスト (SL)。IMHELP タグの前には LI (リスト項目) タグ、または LI タグのテキストが置かれている必要があります。
- | • 順不同リスト (UL)。IMHELP タグの前には LI (リスト項目) タグ、または LI タグのテキストが置かれている必要があります。
- | IMHELP タグの直後にテキストを続けることはできません。この要件を満たすために、構成内で許可される任意のタグを使用できます。ただし、HP*n* (強調表示句)、RT (テキスト反転)、または PK (プログラミング・キーワード) タグを除きます。

例: 組み込みヘルプ

- | この例は、機能キーのヘルプ用にインポートされたヘルプを使用します。2 番目のヘルプ・パネルは、リスト項目内で共通して使用される情報をインポートします。

UIM ソース

```
:HELP name='menu1'.
:H2.Purpose of MENU1
Menu1 is intended for the use of...
:
:IMHELP name='keydefs'.
:
:EHELP.
:
:HELP name='keydefs'.
:PARML.
:PT.F1=Help
:PD.This key...
```

IMHELP タグ

```
:PT.F3=Exit
:PD.This key...
:EPARML.
:EHELP.

| :HELP name='lib1'.
| :P.The valid values for library are:
| :o1.
| :li.APP1LIB
| :li.APP2LIB
| :li.Any IBM-supplied library.
| :imhelp name='ibmlib'.
| :p.
| Some of the above libraries ...
| :li.APP3LIB
| :
| :eol.
| :ehelp.

| :HELP name='ibmlib'.
| :o1.
| :li.QSYS
| :li.QUSRSYS
| :
| :eol.
| :ehelp.
```

IMPORT (インポート)

▶▶ :IMPORT—NAME—== imported-name PNLGRP—== panel-group-name →

▶ NEWNAME—== SAME private-name PRDLIB—== product-library-name .

インポート (IMPORT) タグは、別のパネル・グループ・オブジェクトで定義されたヘルプ・モジュールを現在のパネル・グループで使用できるようにします。このタグは、パネル・グループ (PNLGRP) タグおよび著作権 (COPYR) タグの後に続くパネル・グループ・ソースのプロローグ・セクションに出現する必要があります。IMPORT タグは、それが定義するヘルプ・モジュールを他のタグが参照する前に指定する必要があります。

また、このタグはインポートされたヘルプ・モジュールに専用の名前を割り当てることもできます。専用の名前が割り当てられる場合、現行のパネル・グループの有効範囲で、その専用の名前はインポートされた名前を置き換えます。これにより、現行のパネル・グループ内の他の名前と競合する可能性のあった名前をインポートできます。

必須属性

NAME=*imported-name* | '*'

現行のパネル・グループ内で使用するためにインポートされたヘルプ・モジュールの内部名。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

'*' がコード化される場合、未解決の名前はすべて、このパネル・グループからインポートされたものと想定されます。1 つの IMPORT タグだけを NAME='*' でコード化できます。

PNLGRP=panel-group-name

NAME で指定されたヘルプ・モジュールを含むパネル・グループの名前。これは、すべてのオブジェクト名規則に従う i5/OS オブジェクト名です。この名前は完全修飾にすることができます。

オプションの属性**NEWNAME=SAME | private-name**

現行のパネル・グループ内のインポートされた名前の代わりに使用される新規名。新規名はインポートされた名前が表示されないようにし、その代わりにの役割を果たします。NAME='*' が指定されているときには、NEWNAME=SAME だけが許可されます。

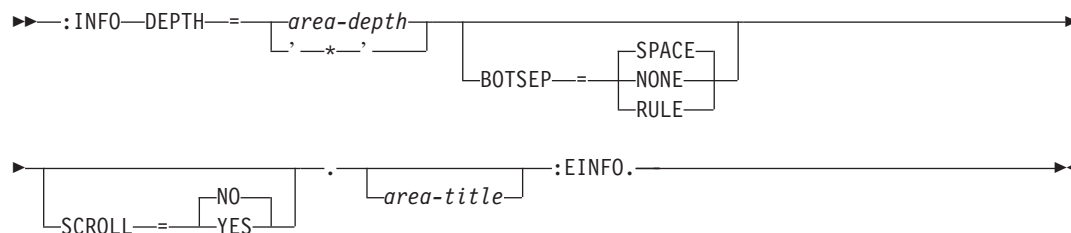
PRDLIB=product-library-name

ライブラリー検索リストに製品ライブラリーとして追加されるライブラリーの名前。このタグの NAME 属性で指定されたヘルプ・モジュールを定義するパネル・グループを見付けるために使用されます。PNLGRP 属性が完全修飾されている場合には、この属性は許可されません。

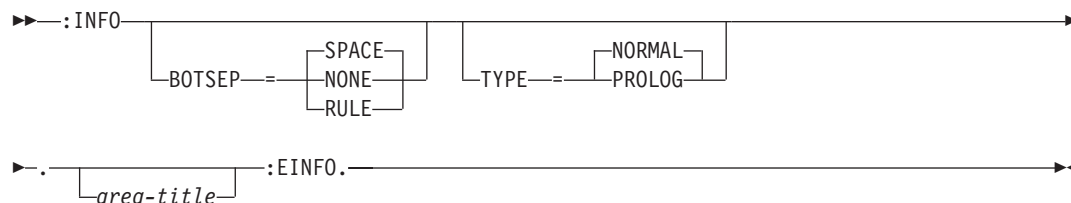
組み込みヘルプ (IMHELP) タグを使用してヘルプ情報を埋め込む場合にのみ、製品ライブラリーが使用されます。その他の UIM タグの HELP 属性を使用してヘルプ・モジュールを参照しているときには、製品ライブラリーは無視されます。

INFO (情報域)

表示パネルの構文:



印刷パネルの構文:



情報域 (INFO) タグは、アプリケーションまたはパネルの操作を説明するテキスト情報を提供します。このタグは、表示パネルおよび印刷パネルで許可されます。これは、パネル上の詳細な説明域を構成することができます。

パネル・タグの幅属性が 74 列より小さくない限り、情報域の幅は 72 列にフォーマットされます。小さい場合は、情報域の幅は、パネル・タグの列属性よりも 2 列小さくフォーマットされます。

必須属性

DEPTH=*area-depth* | '*'

領域の深さを行で示したものの。区切り記号が指定されていれば、それも含まれます。この属性は表示パネルには必要ですが、印刷パネルでは許可されません。 '*' が指定されている場合、その他のパネル・エレメントが割り振られた後で画面上に残っているスペースは、この領域に割り当てられます。パネル内の 1 つの領域のみ、 '*' でコード化できます。

オプションの属性

BOTSEP=SPACE | NONE | RULE

情報域の最下部区切り記号を定義します。 SPACE が指定されている場合、スペースから成る 1 行が使用されます。

NONE は、区切り行が存在しないことを示します。

RULE が指定されている場合、下線付きスペースから成る 1 行が使用されます。

SCROLL=NO | YES

この領域がスクロール可能かどうかを示します。この属性は表示パネルにのみ許可されます。 NO は、領域がスクロール可能でないことを示します。

YES は、領域がスクロール可能であることが意図されていることを示します。 SCROLL=YES の領域の場合、スクロール情報の行を提供するために、UIM によって 1 行のスペースが使用されます。また、BOTSEP=SPACE が指定されている場合、区切り記号およびスクロール情報の行の両方のために 1 行が使用されます。

TYPE=NORMAL | PROLOG

この情報域がプロローグ域かどうかを示します。プロローグ域は、先頭ページのタイトル行の後に 1 回限り印刷されます。この属性は、印刷ヘッド (PRTHEAD) タグにのみ許可されます。 INFO タグが PRTHEAD パネルの情報域用の PRTHEAD および EPRTHEAD タグの間に出現する場合にのみ、この属性は有効です。 NORMAL がデフォルト値です。

オプションのテキスト

area-title

領域のタイトル。テキストが指定されていない場合、領域にタイトル行は割り振られません。テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみを含めることができます。最大長の 55 文字を超えることはできません。

印刷のフォーマットの考慮事項

印刷情報域は表示域と同様にフォーマットされます。ただし、印刷時に 4 行の情報域が 1 ページに収まらない場合を除きます。この場合、ページ排出が行われ、情報域は次のページに印刷されます。情報域は、PRTHEAD または印刷パネル (PRTPNL) タグで指定されている幅に関係なく、表示パネル上の情報域で行われるように、幅が 72 列にフォーマットされます。

ISCH (索引検索)

▶▶: ISCH—ROOTS—'—'root-word-list' —.—index-entry-text—▶▶

索引検索 (ISCH) タグは、ユーザーが同義語と一致する単語を入力した場合に索引検索機能により表示されるテキストを示します。 ISCH タグはパネル・グループ・オブジェクトのソース・コード内でのみ許可され、メニュー・オブジェクトのソースでは許可されません。 ISCH タグの配置により、索引項目が選択されたときに表示されるヘルプ・モジュールが決定されます。 ISCH タグは、参照先のヘルプ・モジュール (HELP) タグの直後に配置する必要があります。

必須属性

ROOTS='root-word-list'

索引項目に適用される最大 50 のルート・ワードのリスト。それぞれのルート・ワードは最大 20 文字まで可能で、文字 A から Z、a から z、および 0 から 9 のみを含めることができます。単語と単語の間に 1 つ以上の空白を指定する必要があり、リスト全体はアポストロフィで囲む必要があります。 ROOTS 属性は繰り返すことができるので、ソース行 1 行に収まる長さより多くのルート・ワードを索引項目に定義することができます。

ルート・ワードは ISCH タグと索引検索同義語 (ISCHSYN) タグの間のリンクとして使用され、ユーザーには表示されません。ユーザーが索引項目を検索できるようにするには、ここで指定されたそれぞれのルート・ワードには、一致するルート・ワードが ISCHSYN タグ上になければなりません。

必須テキスト

index-entry-text

同義語の 1 つが選択された場合に表示される索引項目のテキスト。テキストには 72 文字まで含めることができますが、他のタグを含めることはできません。タグ定義を終了させるピリオドと同じ行またはその次の行にこれを置く必要があり、ソース内で 2 行にまたがって配置することはできません。

索引項目がサブトピックである場合、*index-entry-text* は、索引項目の階層内で 1 つ高位の索引項目のテキストから 2 スペース分下げられます。

提供されるテキストは、トピックが選択用に表示される時のトピック・タイトルとして使用されます。トピックの HELP タグにテキストが指定されていない場合、ISCH タグで提供されたテキストは、表示するためにトピックを選択したときのパネル・タイトルとしても使用されます。パネル・タイトルとして使用される場合、テキストは 55 文字までしか表示されません。テキストに 55 文字より多く含まれる場合、空白の所で切り捨てられ、切り捨てが行われたことを示すためにテキストの後に省略記号 (...) が置かれます。

例: 索引検索

以下の例は、いくつかの ISCHSYN タグと、それらのタグを使用する ISCH タグを示しています。

UIM ソース

```
:ISCHSYN ROOT='copy'.copy copying copies
:ISCHSYN ROOT='delete'.delete deleting deletes
:ISCHSYN ROOT='delete'.remove removes removing
:ISCHSYN ROOT='folder'.folder folders
:ISCHSYN ROOT='folder'.document documents
:
:help name=fldcpy.
:ISCH ROOTS='copy folder'.Copying folders
:
:ehelp.
:help name=flddlt.
:ISCH ROOTS='delete folder'.
Deleting folders
:ehelp.
```

ISCHSUBT (索引検索サブトピック)

▶▶: ISCHSUBT—TOPICS—='help-module-name-list' —. —▶▶

索引検索サブトピック (ISCHSUBT) タグは、索引検索 (ISCH) タグで指定された先行トピックの下にあるサブトピックであるヘルプ・モジュールを、同じパネル・グループ内で識別します。ISCHSUBT タグは ISCH タグの後に出現する必要があります。このタグは反復可能です。

ISCHSUBT タグによって識別されない ISCH タグを持つヘルプ・モジュールは、索引検索階層の主トピックになります。したがって、ISCHSUBT タグを使用しないと、すべてのヘルプ・モジュールが主トピックになり、索引検索に階層はありません。ISCHSUBT タグはパネル・グループ・オブジェクトのソース・コードでのみ許可され、メニュー・オブジェクトのソース・コードでは許可されません。

必須属性

TOPICS='help-module-name-list'

索引検索 (ISCH) タグに指定された先行するトピックの下にあるサブトピックであるヘルプ・モジュールを、同じパネル・グループ内で識別します。ヘルプ・モジュールが TOPICS 属性に指定される順序が、索引検索階層で表示される順序になります。ヘルプ・トピック名は、このリストで 2 回指定することはできません。ヘルプ・モジュール名の規則について詳しくは、515 ページの『名前構文』を参照してください。この属性は反復可能です。

1 つのトピックを複数のトピックのサブトピックにすることができます。

トピックは 16 までのレベルにネストできます。

例: 索引検索階層

以下の例は、ISCH タグと ISCHSUBT タグと一緒に作用して、索引検索階層を形成する様子を示しています。

UIM ソース

```
:HELP name=mainhelp.
:ISCH roots='root words'.
Main Help Topic
:ISCHSUBT topics='help1'
      topics='help2'.
  :
:EHELP.
:HELP name=help1.
:ISCH roots='root words'.
Help number 1
:ISCHSUBT topics='help3 help4'.
  :
:EHELP.
:HELP name=help2.
:ISCH roots='root words'.
Help number 2
:ISCHSUBT topics='help3'
  :
:EHELP.
:HELP name=help3.
:ISCH roots='root words'.
Help number 3
  :
:EHELP.
```

```
:HELP name=help4.
:ISCH roots='root words'.
Help number 4
:
:EHHELP.
```

この UIM ソースは、以下の索引検索階層を作成します。

```
Title of this index
Main Help Topic
  Help number 1
    Help number 3
    Help number 4
  Help number 2
    Help number 3
```

ISCHSYN (索引検索同義語)

▶▶: ISCHSYN—ROOT=—*root-word*—.—*synonym-words*————▶▶

索引検索同義語 (ISCHSYN) タグは、索引検索関数で使用されるルート・ワードの変形および同義語を識別します。ISCHSYN タグはパネル・グループ・オブジェクトのソース・コードでのみ許可され、メニュー・オブジェクトのソースでは許可されません。

テキストは 1 行で入力する必要があります。複数の行が必要な場合は、複数の ISCHSYN タグをコード化することも可能です。このテキストを索引検索 (ISCH) タグと組み合わせると、ユーザーが索引検索関数のための用語を入力したときに表示される索引項目が決定されます。

ISCHSYN タグは同義語の表を作成します。これは、ISCH タグへのリンクとなります。索引検索用の語が入力されると、それらは同義語表の語と突き合わされ、表示される項目にリンクします。

ISCHSYN タグの配置に制約はありませんが、保守および変換を容易にするためには、それらを 1 つの領域 (たとえば、パネル・グループの先頭または ISCHSYN タグのみを含むパネル・グループ・オブジェクト内など) に配置すべきです。

必須属性

ROOT='*root-word*'

同義語が適用されるルート・ワード。ルート・ワードに実語が使用される場合 (『copy』 など)、その語も *synonym-words* フィールドに入力する必要があります。

ルート・ワードは ISCH タグの ROOTS 属性で使用され、最高 20 文字になります。ルート・ワードには、文字 A から Z、a から z、および 0 から 9 のみ含めることができます。

必須テキスト

synonym-words

特殊形式。 ルート・ワードに対する変形および同義語を指定します。同義語は空白で区切る必要があります。ルート・ワードに追加の同義語が必要な場合、同じルート・ワードを指定する追加の ISCHSYN タグを入力しなければなりません。追加の同義語は前の同義語に追加されます。

それぞれの同義語は 40 文字以下でなければならず、スペースを含めることはできません。ピリオド、左括弧または右括弧、セミコロン、コンマ、疑問符 (?)、またはコロンを含めることはできません。

ISCHSYN タグ

言語が異なれば、各語の同義語の数も異なります。意味のある結果を得るために、ルート・ワードの変換は、同義語のリストを用意し、それぞれの英語の同義語を変換しないことによって行う必要があります。

UIM は、パネル・グループ (PNLGRP) タグの TXTCHRID 属性で指定されたコード・ページを使用して、小文字の 1 バイト文字セット (SBCS) の同義語を自動的に大文字語と同じように扱います。

例: 索引検索同義語

この例では、いくつかの ISCHSYN タグと、それらが ISCH タグによって参照される様子を示します。

UIM ソース

```
:pnlgrp...
:
:ISCHSYN ROOT='copy'.copy copying
:ISCHSYN ROOT='copy'.duplicate duplicating
:ISCHSYN ROOT='copy'.model
:ISCHSYN ROOT='root1'.remove removing
:ISCHSYN ROOT='root1'.delete deleting
:ISCHSYN ROOT='root1'.trash discard
:ISCHSYN ROOT='folder'.folder folders
:ISCHSYN ROOT='folder'.document documents
:ISCHSYN ROOT='folder'.data information
:
:help name=fldcpy.
:ISCH ROOTS='copy folder'.Copying a folder
:h2.Copying a Folder
:
:ehelp.
:
:help name=flddlt.
:ISCH ROOTS='root1 folder'.Deleting a folder
:h2.Deleting a folder
:
:ehelp.
```

この例では、ISCHSYN タグは以下の項目を含む同義語表を作成します。

ルート・ワード

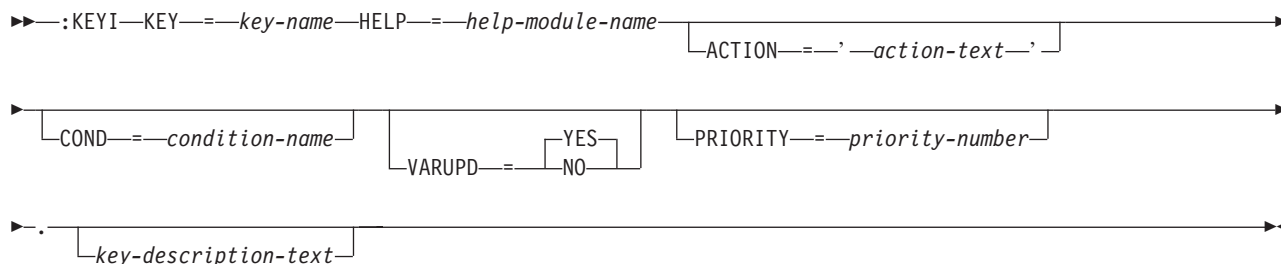
同義語

copy	copy copying duplicate duplicating model
root1	remove removing delete deleting trash discard
folder	folder folders document documents data information

『Copying a folder』 項目用の ISCH タグは、『copy』 および 『folder』 検索同義語リストを指します。

ユーザーが "copy folders" を要求すると、項目が検索されて表示されます。なぜなら、"copy" と "folders" はどちらも対応するものとして ISCHSYN 表にリストされているからです。ユーザーが "copy folders"、"folders, copy"、または "copying documents" を入力した場合、項目 "Copying a folder" が表示されます。

KEYI (キー・リスト項目)



キー・リスト項目 (KEYI) タグは、単一の機能キーを識別します。このタグは、KEYL タグと EKEYL タグの間に出現する必要があります。これは、指定されたキーに表示可能テキストを割り当て、そのキーが押されたときに起こるアクションを識別します。

必須属性

KEY=key-name

刻印された、またはソフトウェア定義済みの機能キーの名前は、Attn、Enter、F1-F24、Help、Home、Print、Pagedown、Pageup、および Sysreq です。

HELP=help-module-name

キーの説明に関連したヘルプ・モジュール。ヘルプ・モジュールの名前は別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説された名前の規則に従っていません。命名規則については、515 ページの『名前構文』を参照してください。

ACTION='action-text'

機能キーが押されたときに生じるアクション。この属性は条件付きで必須です。キーによっては、UIM がその特定のキーを扱わないため、ACTION 属性を指定してはなりません。それ以外のキーの場合、ACTION 属性は必須です。

アクション・テキストの有効な形式を以下に示します。

- 'ACTIONS'
- 'CALL program-reference' UIM と機能キー CALL プログラムの間のインターフェースの説明については、iSeries Information Center で『APIs』のトピックを参照してください。
- 'CANCEL'
- 'CHGVIEW'
- 'CMD command-string' コマンド・ストリング内のダイアログ変数の前には、変数置換を示すアンパサンドが付いていなければなりません。
- 'CMDLINE'
- 'ENTER'
- 'EXIT'
- 'HELP'
- 'HOME'
- 'MENU qualified-menu-name RTNPNT\NORTNPNT'
- 'MOREKEYS'
- 'MOVETOP'

KEYI タグ

- 'PRINT'
- 'PROMPT'
- 'RETRIEVE'
- 'RETURN positive-integer'
- 'PAGEUP'
- 'PAGEDOWN'

それぞれのアクションの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

刻印されたキーの割り当てはコンパイラーによって施行されます。これらのキーは、F1 から F24 を除く、リストされたキーです。刻印されたキーがそれらに対応するダイアログ関数以外のものに割り当てられている場合 (または場合によってはアクションなし)、コンパイル・エラーが起こります。次のリストは、刻印されたキーとそれに対応するアクションを示しています。

キー	アクション
Attn	アクションなし
Enter	'ENTER'
Help	'HELP'
Home	'HOME'
Print	'PRINT'
Pageup	'PAGEUP'
Pagedown	'PAGEDOWN'
Sysreq	アクションなし

オプションの属性

COND=*condition-name*

指定された条件が真の場合にのみ、キーはパネル上で有効になります。条件は、パネル・グループ内で条件定義 (COND) タグを使用して定義する必要があります。

VARUPD=YES | NO

YES がコード化される場合、妥当性検査が行われ、アクションが取られる前にユーザーが入力した値で変数プールが更新されます。

NO がコード化される場合、アクションが即時に実行され、変数プールの更新は行われません。

VARUPD 属性が指定されておらず、このタグの ACTION 属性に EXIT または CANCEL ダイアログ・コマンドが指定されている場合、VARUPD=YES が想定されることを示す警告メッセージがコンパイル時に生成されます。これは注意を喚起するために行われます。なぜなら、これらのダイアログ・コマンドを使用して画面を終了することをユーザーに許可する前に、入力データが有効かどうか検査することを望まない場合があるからです。このメッセージが表示されないようにするには、キー割り当てで VARUPD=YES または VARUPD=NO を指定してください。

ACTION 属性が指定されていない場合、この属性は無視されます。

PRIORITY=*priority-number*

MOREKEYS ダイアログ・コマンドが実行されるときにキーが表示される順序。

優先順位番号の低いキー項目は、優先順位番号の高いキー項目の前に表示されます。指定される優先順位番号は、1 以上 99 以下でなければなりません。

キーの優先順位が指定されていない場合、デフォルト値は指定されたアクションに応じて決まります。

アクション	デフォルト
EXIT	1
CANCEL	2
その他すべて	99
アクションなし	99

PRIORITY 属性は、KEYI タグのいずれかで ACTION=MOREKEYS が指定されているキー・リスト内でのみ使用されます。そうでない場合は、優先順位属性は無視されます。

オプションのテキスト

key-description-text

関連したキーの表示可能な記述。たとえば、"F5=Refresh" または "F12=Cancel" です。テキストが指定されていない場合、キーがアクティブになっていてもキーの記述はパネルに表示されません。テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみを含めることができます。

ファンクション・キー域は、さまざまなキー・リスト項目のテキストを、少なくとも 3 つのスペースで区切った各キーと連結することにより、形成されます。テキストが行の残りの部分に収まらない場合は、次の行に置かれます。

例: キー定義

以下の例は、ファンクション・キー F3、F5、F11、および F12 を定義します。

UIM ソース

```
:keyl name=keylist
      help=hkeylist.
:keyi key=f3
      action='exit set'
      help=exit
      .F3=End task
:keyi key=f5
      action='return 5'
      help=refresh
      .F5=Refresh
:keyi key=f11
      action=chgview
      help=chgview
      .F11=Alternate view
:keyi key=f12
      action='cancel set'
      help=cancel
      .F12=Cancel
:ekeyl.
```

結果

F3=End task F5=Refresh F11=Alternate view F12=Cancel

KEYL (キー・リスト)

```
▶▶:KEYL—NAME==—key-list-name—└─HELP==—help-module-name—┘.—:EKEYL.—▶▶▶▶
```

キー・リスト (KEYL) タグは、キー定義のリストの先頭を示します。各リストは、1 つ以上のパネル定義によって参照されることがあります。EKEYL タグは、キー・リストの末尾に指定する必要があります。

他のタグを KEYL タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、各タグについて詳しく説明しているページを示しています。

表 62. KEYL タグと EKEYL タグの間で許可されるタグ

タグ名	順序	ページ
KEYI (キー・リスト項目)	1	596

必須属性

NAME=key-list-name

キー・リストに割り当てられた名前。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

オプションの属性

HELP=help-module-name

機能キーの目的を説明するオンライン・ヘルプ情報を識別します。ヘルプ・モジュールの名前は、別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説した名前の規則に従っていなければなりません。名前の規則について詳しくは、515 ページの『名前構文』を参照してください。

例: キー・リスト

これは、キー・リストの例です。

UIM ソース

キー・リストの例は以下のとおりです。

```
:keyl name=keylist.
:keyi key=f1
      help='key/helpf1'
      action=help.
:keyi key=f3
      help='key/exit'
      action='exit set'.F3=Exit
:keyi key=f12
      help='key/cancel'
      action='cancel set'.F12=Cancel
:keyi key=f24
      help='key/morekeys'
      action=morekeys.F24=More keys
:keyi key=enter
      help='key/enter'
      action=enter.
:keyi key=help
      help='key/help'
```

```

        action=help.
:keyi key=pagedown
        help='key/pagedown'
        action=pagedown.
:keyi key=pageup
        help='key/pageup'
        action=pageup.
:keyi key=print
        help='key/print'
        action=print.
:ekey].

```

LINES (不定形式行)

▶▶—:LINES.—└──────────────────┘:ELINES.—▶▶

└──────────┘

unformatted-lines

不定形式行 (LINES) タグは、行が自動的に連結されないユーザー制御の行項目の領域を識別します。テキストは入力したとおりにフォーマットされます。このタグは情報域とヘルプ領域でのみ許可されます。LINES タグには、対応する終了タグが必要です。

注: リスト内の不定形式行を使用するときには注意が必要です。なぜなら、1 行に収まらないテキストは次の行の列 1 に折り返されるからです。ネストされたリストがフォーマットされると、現行左マージンは変わります。さまざまな位置 (リスト内を含む) に組み込まれた不定形式行を含むオンライン・ヘルプ情報がある場合、それは毎回同じように見えるわけではありません。

フルスクリーン・フォーマットまたは拡張ヘルプ・ウィンドウで表示されるヘルプ・モジュールは、4 バイト下げられます。そのため、最初のフォーマット幅は、ヘルプ・モジュールで指定された幅より 6 バイト小さくなります。LINES タグが情報域に表示される場合、最初のフォーマット幅は、パネル・タグで指定された幅より 2 バイト小さくなります。

オプションのテキスト

unformatted-lines

不定形式行のテキストは必須ではありません。ただし、タグはテキストが指定されていないときには意味を持ちません。

例: 不定形式行

この例は、テキストが入力されたとおりにフォーマットされる様子を示しています。

UIM ソース

```

:LINES.
First line
Second line
:ELINES.

```

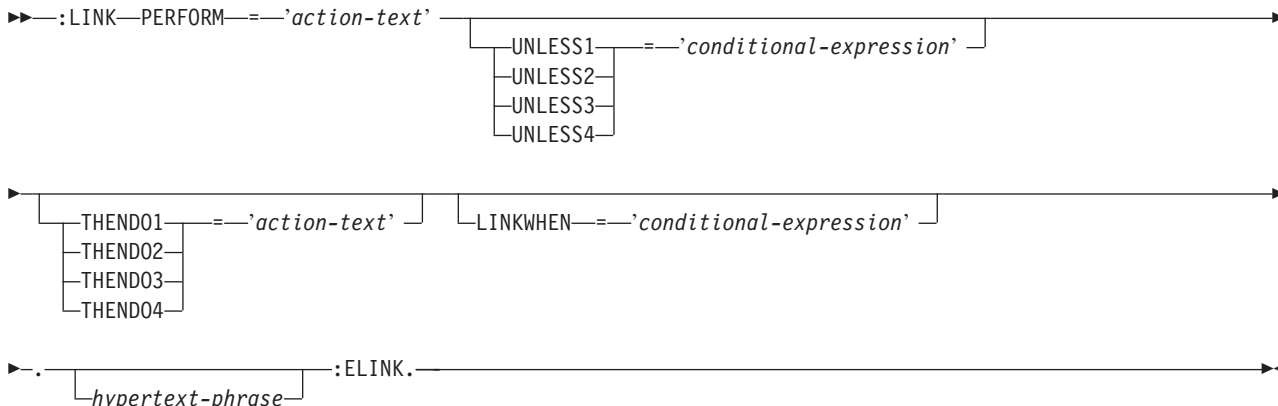
結果

```

First line
Second line

```

LINK (ハイパーテキスト・リンク定義)



ハイパーテキスト・リンク定義 (LINK) タグは、静的ハイパーテキスト・リンクのアンカーである参照句テキストを識別します。参照句は UIM ヘルプ領域に表示できますが、情報域には表示できません。このタグには、対応する終了タグが必要です。また、参照句が選択されるときにとられるアクションを識別します。サポートされるアクションのタイプは、ヘルプ・モジュールの表示のみです。

LINK および ELINK タグはワード境界で指定する必要があります。ELINK タグの直後に続く 2 文字が句読記号と空白である場合、UIM は強調属性を自動的に拡張して、句読記号を含めます。これにより、句読記号とそれに関連したテキストを、同じ強調を使用して表示できます。

単色デバイスでは、参照句テキストは強調表示された下線付きテキストとして表示されます。カラー・デバイスでは、参照句テキストは下線付きの黄色いテキストとして表示されます。ハードウェア制限のため、下線は PS/2 コンピューターまたはカラー・デバイスを搭載したパーソナル・コンピューターでは見ることができません。

リンク・タグ内で使用できるタグはテキスト反転 (RT) タグのみです。LINK タグはその他の LINK タグ内では使用できませんが、以下のいずれかのタグ内では使用することが可能です。

CIT	タイトルの引用
DD	定義
FIG	図表
FIGCAP	図表表題テキスト
HP_n	強調表示句
LI	リスト項目
LINES	不定形式行
LP	リストの一部
NT	注記
P	段落
PC	段落の連結
PD	パラメーター記述
PK	プログラミング・キーワード

PV プログラミング変数

XMP 例

必須属性

PERFORM=*'action-text'*

参照句が選択され、UNLESS n 属性の条件式がすべて偽と評価されているときに取られるアクション。アクション・テキストの有効な形式を以下に示します。

- *'DSPHELP help-module-name [panel-group-name]'*

注: 上記のアクション・テキスト内のブラケットは、パネル・グループ名がオプションであることを示します。それらは、UIM ソースでは必須ではありません。

DSPHELP アクションの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

オプションの属性

UNLESS n ='conditional-expression'.

n に許可される値は 1 から 4 のみです。

UNLESS n と THENDO n 属性は対でコード化する必要があります。UNLESS n 属性は数値順に評価されます。あるものが真と評価される場合、対応する THENDO n アクションが実行され、このタグで番号の大きい UNLESS n および THEN n 属性と、PERFORM 属性はすべて無視されます。

UNLESS n 属性を持つ条件式の使用については、600 ページの『条件式』を参照してください。

THENDO n ='action-text'

n に許可される値は 1 から 4 のみです。

この属性は、参照句が選択され、対応する UNLESS n 属性の条件式が真と評価され、かつ小さい番号の UNLESS n 属性の条件式がすべて偽と評価されるときに取られるアクションを指定します。

UNLESS n と THENDO n 属性は対でコード化する必要があります。UNLESS n 属性は数値順に評価されます。あるものが真と評価される場合、対応する THENDO n アクションが実行され、このタグで番号の大きい UNLESS n 属性および PERFORM 属性はすべて無視されます。

アクション・テキストの有効な形式を以下に示します。

- *'DSPHELP help-module-name [panel-group-name]'*

注: 上記のアクション・テキスト内のブラケットは、パネル・グループ名がオプションであることを示します。それらは、UIM ソースでは必須ではありません。

DSPHELP アクションの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

LINKWHEN='conditional-expression'.

LINKWHEN 式が真と評価されるか、またはそれがコード化されていない場合、参照句が使用可能になり、ユーザーはそれを選択することができます。LINKWHEN が偽と評価されると、参照句はアクティブにされず、それを選択できません。

LINKWHEN 属性を持つ条件式の使用については、600 ページの『条件式』を参照してください。

LINK タグ

条件式

LINKWHEN 属性で形成できる条件のセットは、UNLESS n 属性で形成できる条件のセットと同じですが、それらの条件の重要度は異なります。LINKWHEN 属性は LINK タグをアクティブにしたり非アクティブにしたりしますが、UNLESS n 属性は、アクティブな LINK タグが選択されたときに実行されるアクションを選択します。条件式は、次の形式で真または偽の式になります。



ここで、A は以下のいずれかになることができるオペランドです。

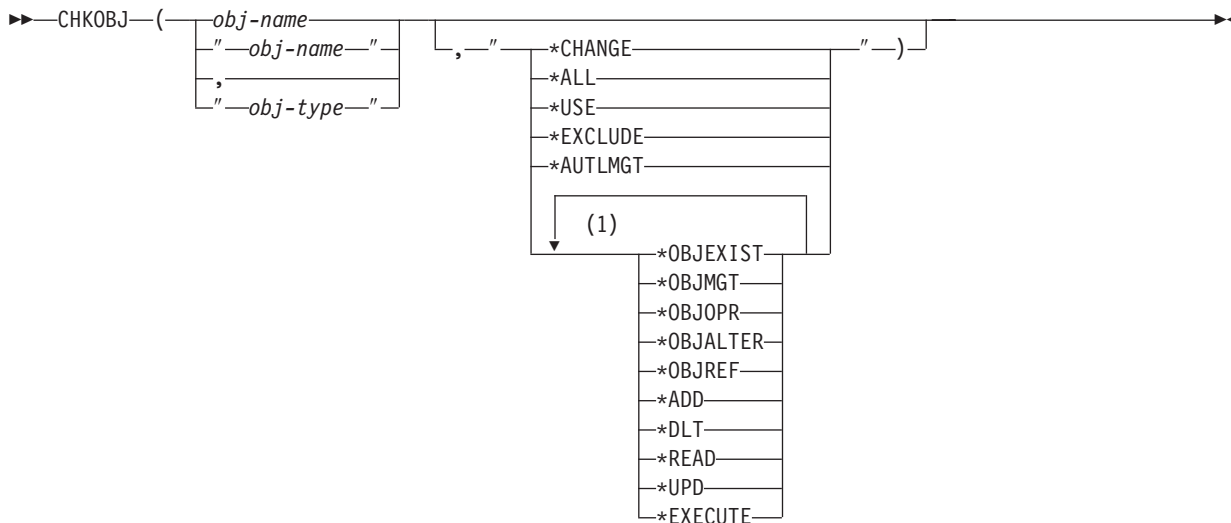
- 条件式
- 組み込み関数

論理 OR 文字 (|) は *OR の代わりに使用できます。アンパーサンド文字 (&) は *AND の代わりに使用できます。また、論理 NOT 文字 (!) は *NOT の代わりに使用できます。論理 OR および論理 NOT 文字は不変文字セットにないため、これらの使用は推奨されていません。コード・ページ 00037 (共通 USA コード・ページ) の場合、論理 OR 文字の 16 進値は X'4F' で、論理 NOT 文字の 16 進値は X'5F' です。ソースのコード・ページにかかわらず、UIM コンパイラーはこれらの 16 進値を使用します。

以下の 3 つの組み込み関数があります。

CHKOBJ

オブジェクトがシステム上にあり、現行ジョブが、少なくとも権限によって指定された許可レベルをオブジェクトに対して所有している場合に、真と評価されます。引数は、二重引用符 (") で囲まれた文字ストリングでなければなりません。オブジェクト名は i5/OS オブジェクト命名規則に従います。オブジェクト・タイプは DSPOBJD コマンドの許容オブジェクト・タイプのいずれかであり、権限は検査対象の単一値または権限リストでなければなりません。これらの権限の値はブランクで区切られます。以下の構文図に、権限とその使用方法を示します。



注:

1 各値は 1 回のみ使用でき、最大値は 7 です。

権限が指定されていない場合、許可検査は実行されず、関数は存在検査になります。

プログラムがその所有者の権限を借用すると、プログラムによって実行される操作を許可するために、通常はその権限が使用されます。しかし、ヘルプが要求されると、それらの借用権限は無視され、UIM はオンライン・ヘルプ情報を表示します。そのため、LINK タグの UNLESS または LINKWHEN 属性で CHKOBJ 関数が使用されるときには、借用権限は使用されません。

所有者の権限を借用するプログラムについて詳しくは、CRTCLPGM コマンドの USRPRF(*OWNER) パラメーターを参照してください。

CHKPGM

CHKPGM 関数は、出口プログラムとして UIM から呼び出される修飾プログラム名を受け入れます。プログラムを呼び出すことができない場合、関数は偽と評価されます。出口プログラムは、関数が真または偽のいずれに設定されるかを判別し、標識を UIM に戻します。プログラム名は二重引用符 (") で囲まなければなりません。*LIBL 特殊値をライブラリー名の代わりに使用できます。ライブラリー名が入力されない場合、*LIBL がデフォルトです。

以下の構文図に、CHKPGM 関数の有効な引数値を示します。

```
▶▶CHKPGM(—"pgm-name"—)
```

CHKUSRCLS

関수에指定されたユーザー・クラス引数は、現行ジョブのユーザー・プロファイルからのユーザー・クラス・パラメーターと比較されます。ユーザー・プロファイルの値が関数の引数と同じかそれ以上の場合は、関数は真と評価されます。関数の引数は二重引用符 (") で囲まなければなりません。

以下の構文図に、ユーザー・クラスの有効な引数値を示します。

```
▶▶CHKUSRCLS(—" *SECOFR "
  " *SECADM "
  " *PGMR "
  " *SYSOPR "
  " *USER ")
```

条件式の例は以下のとおりです。

```
CHKOBJ("OBJECT", "*FILE", "*USE")
```

```
CHKOBJ("PANELGRP", "*PNLGRP")
  *AND CHKUSRCLS(" *PGMR")
```

```
CHKOBJ("DOCUMENT", "*DOC", "*READ *UPD")
  *OR CHKUSRCLS(" *SYSOPR")
```

```
*NOT(CHKOBJ("PROGRAM", "*PGM"))
CHKPGM(" *LIBL/PROGRAM")
CHKPGM("PROGRAM")
```

双方向の考慮事項

ハイパーテキスト・リンクを取り込むことによって取得できる、考えられるすべてのパネル・グループのパネル・グループ (PNLGRP) タグの BIDI 属性の値は同じでなければなりません。同じでない場合、参照句が選択されるときに、表示される画面またはポップアップ・ウィンドウは反対方向で表示されます。

例: ハイパーテキスト・リンク

この例は、用語の定義を提供するためにハイパーテキスト・リンクをどのように使用できるかを示します。

LINK タグ

UIM ソース

```
:HELP NAME='hyper/help'.Hypertext in i5/OS
:P.
Hypertext lets users explore
the online help information in a way that is most
natural for them.
Hypertext links can be used within :LINK
PERFORM='DSPHELP item/specific/help'
        .item specific help:ELINK.,
:LINK PERFORM='dsphelp extended/help'
        .extended help:ELINK., and
:LINK PERFORM='dsphelp index/search'
        .index search help:ELINK..
:EHELP.

:HELP NAME='item/specific/help'
        .Definition of Item Specific Help
:P.
Item specific help is . . .
:EHELP.

:HELP NAME='extended/help'
        .Definition of Extended Help
:P.
Extended help is . . .
:EHELP.

:HELP NAME='index/search'
        .Definition of Index Search
:P.
Index search allows you to tell the
system to search for specific information.
Index search information is made more
useful by the addition of
:LINK PERFORM='dsphelp hyper/help'
        .hypertext:ELINK. because
it allows you to link to additional help
topics.
:EHELP.
```

Results

```
.....
:                               Hypertext on i5/OS                               :
:                               :                                               :
: Hypertext lets users explore the online information in a way                 :
: that is most natural for them. Hypertext links can be used                 :
: within item specific help, extended help, and                               :
: index search help.                                                         :
:                               終わり                                         :
: F2=拡張ヘルプ      F10=最初へ移動      F11=見出し検索                     :
: F12=取消し        F13=情報援助        F24=キーの続き                       :
:                               :                                               :
:.....
```

カーソルが *INDEX SEARCH HELP* に移動されて、Enter キーが押されると、次の画面が表示されます。

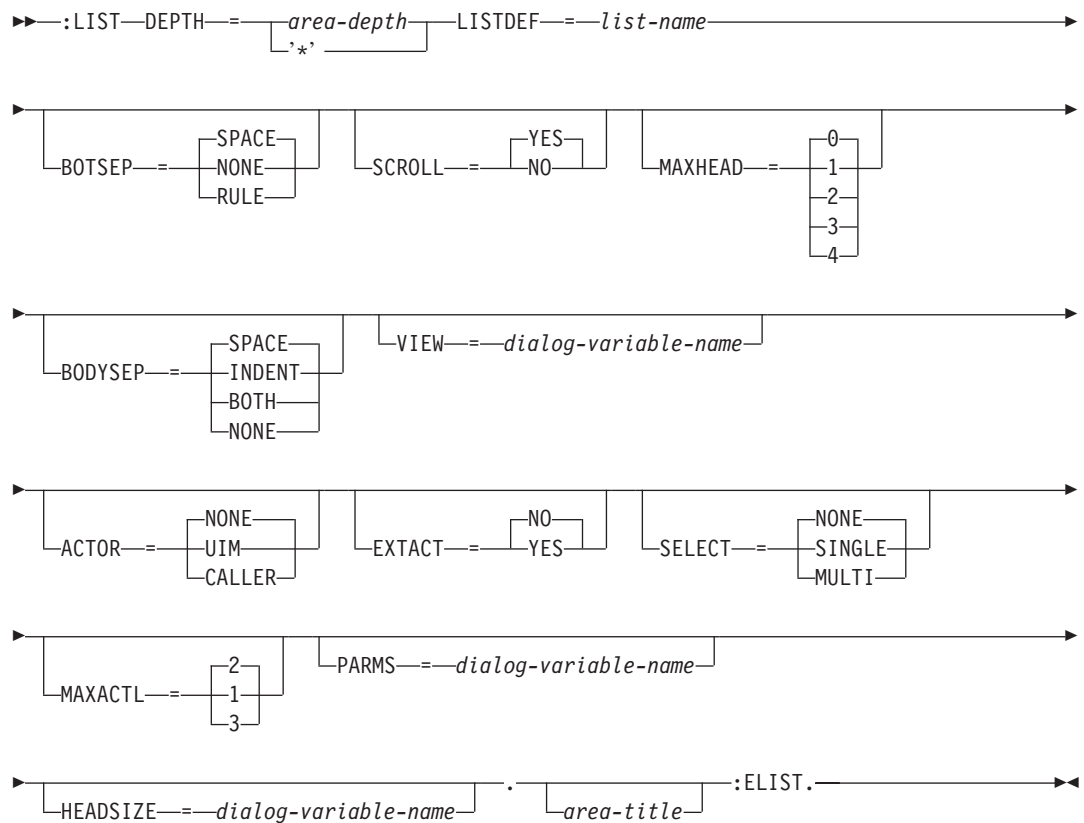
```

: .....
:           Definition of Index Search           :
: .....
: Index search allows you to tell the system to search for specific :
: information. Index search information is made more useful by the :
: addition of > hypertext because it allows you to link to      :
: additional help topics.                                     :
:                                                                       :
:                                                                       :
: F6=表示済みトピック F10=最初へ移動   F11=見出し検索           :
: F12=取消し           F13=情報援助     F24=キーの続き           :
:                                                                       :
: .....

```

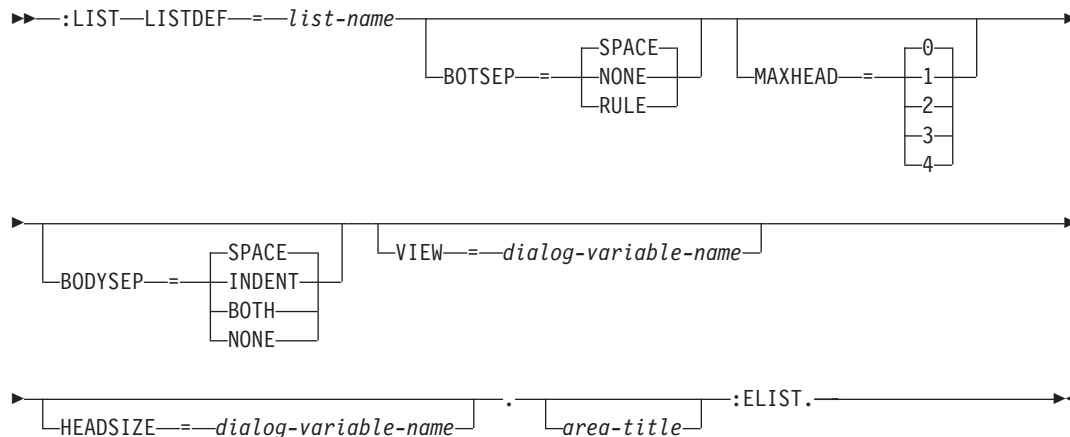
LIST (リスト域)

表示パネルの構文:



LIST タグ

印刷パネルの構文:



リスト域 (LIST) タグは、パネル上のリスト域を定義します。

このタグは、印刷ヘッド・パネル (PRTHEAD) タグを除いて、表示パネルと印刷パネルの両方で許可されます。リスト域は、列の類似した任意の数の行で構成され、行数がリストの表示される領域を超える場合には、スクロールアップしたり、スクロールダウンすることができます。

リスト域は UIM リストのビューを表示します。これは、UIM アプリケーション・プログラミング・インターフェース (API) によって扱われます。UIM リストは、リスト定義 (LISTDEF) タグによって名前が付けられ、記述されます。UIM はリスト内の項目を表示し、スクロール操作を処理して、ユーザーがすべての項目を見ることができるようになります。

領域は複数のビューを持つことがあります。それらは CHGVIEW ダイアログ・コマンドによって選択されます。CHGVIEW ダイアログ・コマンドを、ビューの間を行き来するファンクション・キーに割り当てることができます。

他のタグを LIST タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 63. LIST タグと ELIST タグの間で許可されるタグ

タグ名	順序	使用法	ページ
TOPINST (最上部指示行)	1	D	687
LISTACT (リスト・アクション)	2	D	614
LISTGRP (リスト列グループ)	3	B	628
LISTCOL (リスト列)	3	B	621
LISTVIEW (リスト・ビュー)	4	B	631

表 63. LIST タグと ELIST タグの間で許可されるタグ (続き)

タグ名	順序	使用法	ページ
BOTINST (最下部指示行)	5	D	521

必須属性

DEPTH=*area-depth* | '*'

この属性は表示パネルには必要ですが、印刷パネルでは許可されません。領域の深さを行で示したものの。区切り記号が指定されていれば、それも含みます。 '*' が指定されている場合、その他のすべてのパネル・エレメントが割り当てられた後で画面上に残っているスペースは、この領域に割り当てられます。パネル内の 1 つの領域のみ、 '*' でコード化できます。

LISTDEF=*list-name*

このリスト域のデータが選ばれる UIM リストの名前。リストは、パネル・グループ内で LISTDEF タグを使用して定義する必要があります。

オプションの属性

BOTSEP=SPACE | NONE | RULE

リスト域の下部区切り記号を定義します。SPACE が指定されている場合、スペースから成る 1 行が使用されます。

NONE は、区切り行が存在しないことを示します。

RULE が指定されている場合、下線付きスペースから成る 1 行が区切り行として使用されます。

SCROLL=YES | NO

リスト域がスクロール可能かどうかを示します。この属性は表示パネルにのみ許可されます。YES は、リスト域がスクロール可能であることが意図されていることを示します。SCROLL=YES の場合、スクロール情報の行を提供するために、UIM によって 1 行のスペースが使用されます。

NO は、リストがスクロール可能でないことを示します。通常、NO は使用されませんが、短縮リストの場合、リスト域のスクロール・キーを使用不可にすることを指定できます。

BOTSEP=SPACE が指定されており、区切り記号とスクロール情報の行の両方に最下部指示がない場合、1 行が使用されます。

MAXHEAD= 0 | 1 | 2 | 3 | 4

列見出しの最大行数。0 から 4 行を指定できます。デフォルトは 0 で、列見出しが許可されていないことを示します。列見出しについて詳しくは、621 ページの『LISTCOL (リスト列)』を参照してください。この属性には、列グループ見出しで必要とされる見出し行が含まれます。

列見出しが使用される場合、各国語変換用の拡張スペースが提供されます。これは、必要な見出し行数より大きい MAXHEAD 値を指定することによって行われます。この拡張スペースが提供されない場合、各リスト列 (LISTCOL) タグの MAXWIDTH 属性で決定された列幅の範囲内で提供する必要があります。

このタグの HEADSIZE 属性が変数列見出しとともに使用するために指定されている場合、この属性は指定できません。

BODYSEP=SPACE | INDENT | BOTH | NONE

領域の本体と領域内のその他のエレメント (特に、領域内の最上部指示行および最下部指示行) を区別する、目に見える区切りのタイプ。

LIST タグ

SPACE は、最上部指示行の後および最下部指示行の前では空白行のままです。領域の本体にあるリスト列は、レイアウト列の左端の位置で始まり、指示行については字下げされません。領域に最上部指示行または最下部指示行が含まれない場合、領域本体の前後に空白行は予約されません。

INDENT は、領域の本体にあるリスト列を、指示行が始まるレイアウト列の左端の位置から 2 桁ずつ字下げするために使用します。領域に最上部指示行または最下部指示行が含まれる場合、領域がスクロール可能である場合を除いて、指示と領域本体の間に空白行は予約されません。領域がスクロール可能な場合、空白行はスクロール情報用に予約されます。

BOTH は、最上部指示行の後および最下部指示行の前では空白行のままになり、レイアウト列で指示行が始まる左端の位置から本体を 2 桁字下げします。領域に最上部指示行または最下部指示行が含まれない場合、領域本体の前後に空白行は予約されません。

NONE は、指示行と本体の間を空白行のままにせず、また領域がスクロール可能である場合を除き、レイアウト列については本体を字下げしません。領域がスクロール可能な場合、空白行はスクロール情報用の行を提供するために予約されます。

VIEW=dialog-variable-name

これは、ユーザーに表示されるリストのビューを判別する BIN 15 ダイアログ変数です。有効な値は 0 から、定義されたビューの数より 1 小さい数までです。値が有効ではない場合、または CHGVIEW ダイアログ・コマンドがリスト域に対して実行される場合、UIM はこの変数をアクティブなビューの数と一致するように変更します (0 が最初のビューです)。

リスト域の最初のリスト・ビュー (LISTVIEW) タグがビュー 0 を定義し、リスト域の 2 番目の LISTVIEW タグがビュー 1 を定義します。このようにして、リスト域の各 LISTVIEW タグについて定義されます。

リスト・ビューについて詳しくは、631 ページの『LISTVIEW (リスト・ビュー)』を参照してください。リスト域について複数の LISTVIEW タグが指定される場合、VIEW 属性を指定する必要があります。

ACTOR=NONE | UIM | CALLER.

ACTOR=NONE が使用される場合、リストは処置リストではありません。この属性は表示パネルにのみ許可されます。リスト・アクション (LISTACT) タグは指定できません。そのため、UIM はリスト域に対して処置リスト処理を行いません。

ACTOR=UIM が指定される場合、LISTACT タグで指示されるアクションが UIM によって実行され、ENTER 属性を指定する必要があります。

ACTOR=CALLER の場合、LISTACT タグで指示されるアクションが display panel (QUIDSPP) API の呼び出し側プログラムによって実行される必要があります。

EXTACT=NO | YES

処置リストがアクション機能を拡張しているかどうかを指定します。この属性は表示パネルにのみ許可されます。拡張処置リストがある場合、列見出しの下の最初の行は拡張アクション項目のために使用されます。この行には、追加のリスト項目用の入力可能フィールドに加えて、アクション・オプション列が含まれます。

ユーザーは拡張アクション項目用のオプション・フィールドにリスト・アクション・オプションを入力し、その他の列にデータを入力できます。リスト・アクション・タグによって識別されるアクションは、その他のリスト列用に入力されたデータを使用して実行されます。拡張アクション項目列に入力されたデータは、既存のリスト項目内のデータと一致している必要はありません。

EXTACT=NO は、リストに拡張アクション機能がないことを示します。

EXTACT=YES は、リストに拡張アクション機能があることを示します。これは、ACTOR=UIM または ACTOR=CALLER がこのタグに指定されている場合にのみ許可されます。それぞれのリスト・ビューで定義されている処置リスト・オプション列と少なくとも 1 つのその他のリスト列は、LISTCOL タグに EXTACT=YES を指定して拡張アクションに使用されるように定義する必要があります。拡張アクション・フィールドで作用できる LISTACT タグでは、ACTFOR=BOTH または ACTFOR=EXTACTE が指定されている必要があります。

拡張処置リストを持つパネルが表示されていても、そのリストがオープン・アプリケーションで現在アクティブになっていない場合、リストは QUIDSPP API によって活動化されます。

SELECT=NONE | SINGLE | MULTI

SELECT=NONE が使用される場合、リストは選択リストではありません。この属性は表示パネルにのみ許可されます。

SELECT=SINGLE が指定されている場合、リスト域は単一選択項目の選択リストです。単一選択項目の選択リストが表示される場合、各リスト項目の前にピリオド (.) が付きます。ユーザーは、スラッシュ (/) または各国指定文字をピリオドの上に入力することにより、1 つだけのリスト項目を選択できます。

SELECT=MULTI が指定されている場合、リストは複数項目選択リストです。複数リスト項目を選択するには、スラッシュ (/) または各国指定の文字を、選択する各リスト項目の前にある入力フィールドに入力します。選択項目を選択解除するには、ユーザーはリスト項目にブランクを入力するか、スラッシュ (/) または各国指定の文字を削除する必要があります。

選択したリスト項目のアクション変数には、値 1000 が設定されます。

SELECT が SINGLE か MULTI のいずれかである場合には、このタグに ACTOR=NONE を指定するか、またはこれをデフォルトする必要があります。リストのビューごとに、アクション変数を宣言する必要があります。

MAXACTL= 2 | 1 | 3

リスト・アクション記述に使用される最大行数。1 から 3 行を指定できます。2 がデフォルトです。必要な数のリスト・アクション行だけが使用されます。この属性は表示パネルにのみ許可されます。

MAXACTL=1 が有用になるのは、複数のリスト・アクションがあり、各国語変換の後でそのリスト・アクションのうちの 2 行が必要でないときに限ります。

UIM が最初の 2 つのリスト・アクション行をフォーマットすると、各アクションの先頭をその上か下にある行に位置合わせしようとします。これは、最初の行か 2 番目の行でテキストを右に数列移動させることによって行われます。3 番目のリスト・アクション行のテキストがフォーマットされると、最初の 2 行のテキストは移動されませんが、UIM は 3 番目の行のテキストを 2 番目の行のテキストと位置合わせしようとします。

PARMS=*dialog-variable-name*

この属性は、CHAR 255 ダイアログ変数を指定する必要があります。これは、UIM が処置リスト処理に関するパラメータ情報を保管するために使用されます。この属性は表示パネルにのみ許可されます。また、このタグの ACTOR 属性に NONE 以外の値があるときにのみ有効になります。同じパネル上でリスト域がコマンド行とともに作用するときに、この属性を使用する必要があります。これにより、ユーザーは処置リスト処理に影響を与えるパラメータを指定できるようになります。

コマンド行に処置リスト処理用のパラメータが含まれている場合、UIM はコマンド行の内容をこの変数に保管します。コマンド行にパラメータ情報が含まれていない場合、UIM は処置リスト処理の前に変数をブランクに設定します。

このダイアログ変数は、LISTACT タグの ENTER、EXTENTER、PROMPT、または EXTPROMPT 属性に指定された CMD ダイアログ・コマンドのアクション・ストリングで使用するよう意図されています。

LIST タグ

注: コマンド行が処置リスト処理に指定するパラメーターとして解釈されると、この変数にはコマンド行の内容のみが含まれます。この変数は、コマンド行の内容に対する汎用アクセスを提供しません。

HEADSIZE=*dialog-variable-name*

このダイアログ変数は、リスト列見出しに使用される LISTCOL タグの COLHEAD 属性に指定されたダイアログ変数の数を指定します。MAXHEAD が指定されている場合、この属性は許可されません。

このダイアログ変数には 0 から 10 の値のみ含めることができます。これらの値が見出しテキストに有効な行数です。このダイアログ変数が 0 から 10 までの有効な値ではない場合、最大値の 10 が使用されます。LISTCOL タグの COLHEAD 属性に指定されたダイアログ変数をリストから使用する場合、その変数は COLHEAD 属性で定義されている順序で使用されます。このダイアログ変数は、クラス定義 (CLASS) タグの BASETYPE を 'BIN 15' にして定義する必要があります。

LISTCOL タグの HEADSIZE 属性および COLHEAD 属性に指定されたダイアログ変数は、通常ダイアログ変数と同様に評価されます。QUIDSPP API を呼び出す前に HEADSIZE ダイアログ変数を設定し、ビューを変更しているときにはそのダイアログ変数を変更しないことが推奨されています。リストおよび見出しをパネル上でシフトアップおよびシフトダウンさせないようにするのは、アプリケーション・プログラマーの責任です。

リスト域にリスト列グループが含まれている場合には、この属性を指定できません。

オプションのテキスト

area-title

領域のタイトル。テキストが指定されていない場合、領域にタイトル行は割り振られません。テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみを含めることができます。最大長の 55 文字を超えることはできません。

印刷のフォーマットの考慮事項

印刷されたリスト域は、以下の変更を加えて表示されたものと同様にフォーマットされます。

- 印刷は常にリスト内の最初の項目から開始します。
- 上部にある不完全なリストの場合、UIM はリスト内の最初の項目から印刷を開始し、リストの上部に完全であるというマークが付けられている必要はありません。
- 下部にある不完全なリストの場合、UIM はリスト項目がなくなるまでフォーマットと印刷を続け、不完全なリストでさらに多くの項目を処理するために出口プログラムを呼び出します。一般に、出口プログラムは大量のリスト項目を処理するように依頼され、リストの下部に完全というマークが付けられていない場合、出口プログラムはさらに多くのリスト項目を処理するために再び呼び出されます。
- 複数のレイアウトの場合、UIM は 1 ページに印刷できる数の項目をフォーマットしてから、そのページを印刷します。ページを満たす項目が足りない場合、レイアウト列間で項目の平衡が保たれます。最初の列には常に最低 2 つの項目があり、その後で項目は後続の列に置かれます。これは、ユーザーがリストを読み取る方法を理解するのに役立ちます。たとえば、LAYOUT=2 が指定され、2 つだけのリスト項目を持つリストは、次のように印刷されます。

```
Column 1      Column 2
xxxxxxx
xxxxxxx
```

リスト・パネルの場合、ページ上には、列見出しのほかに最低 2 行のリスト項目のための余地がなければなりません。2 行のリスト項目がページに収まらない場合、ページ排出が行われ、リスト項目は次のページに印刷されます。次の例は、1 ページに収める必要があります。そうしない場合は、新規ページが使用されます。

Column 1	Column 2
xxxxxxx	xxxxxxxxx
xxxxxxx	xxxxxxxxx

リスト域が別のページに続く場合、リスト列見出しとグループ見出しは各ページで繰り返されます。

例 1: リスト域

この例は、UIM がユーザーに要求されたアクションを実行するというアクションを示します。

UIM ソース

```
:listdef name=outflist
  vars='opt fil nbr usr pri pg sts co'.
:panel topsep=space
  ...
  .Output Files
:list depth='*'
  listdef=outflist
  maxhead=2
  actor=uim
  parms=pvar.
:topinst.Type options, press Enter.
:listact
  enter='CMD CHGSPLFA FILE(&FIL)'
  enter='JOB(&USR/&SPID) SPLNBR(&NBR) &PVAR'
  help=opt2
  option=2.2=Change
:listact
  enter='CMD CNLSPLF FILE(&FIL)'
  enter='JOB(&USR/&SPID) SPLNBR(&NBR)'
  help=opt4
  option=4.4=Cancel
:listact
  enter='CMD DSPSPLF FILE(&FIL)'
  enter='JOB(&USR/&SPID) SPLNBR(&NBR)'
  help=opt5
  option=5.5=Display
:listact
  enter='CMD HLDSPLF FILE(&FIL)'
  enter='JOB(&USR/&SPID) SPLNBR(&NBR)'
  help=opt7
  option=7.7=Hold
:listact
  enter='CMD RLSSPLF FILE(&FIL)'
  enter='JOB(&USR/&SPID) SPLNBR(&NBR)'
  help=opt8
  option=8.8=Release
:listcol var=opt
  usage=inout
  maxwidth=6
  help=hopt.Opt
:listcol var=fil
  usage=out
  maxwidth=10
  help=hfil.File
:listcol var=nbr
  usage=out
  maxwidth=6
  help=hnbr.Nbr
:listcol var=usr
  usage=out
  maxwidth=10
  help=huser.User
:listcol var=pri
  usage=out
```


LIST タグ

```
maxwidth=7
.Read
:listcol var=add
usage=out
maxwidth=7
.Add
:listcol var=update
usage=out
maxwidth=7
.Upd
:listcol var=delete
usage=out
maxwidth=7
.Dlt
:elistgrp.
:listview layout=1
cols='user objrights dtarights'
:elist.
:epanel.
```

結果

User Authorizations								System: XXXXXXXX
User Name	----Object Rights----			-----Data Rights-----				
	Oper	Mgmt	Exist	Read	Add	Upd	Dlt	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	
XXXXXXXXXX	X	X	X	X	X	X	X	

F3=Exit F12=Cancel

More...

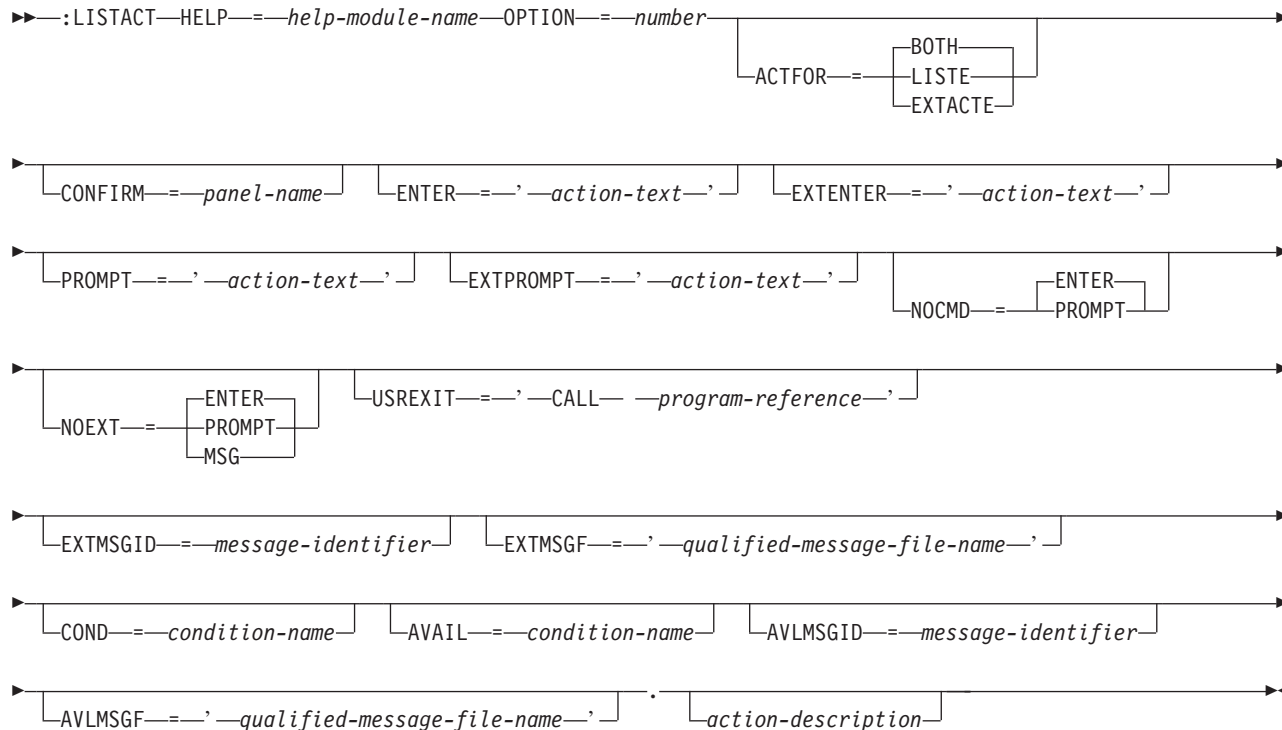
例 4: 動的リスト列見出しのフォーマット

この例は、列見出しをダイアログ変数に含める方法を示します。

UIM ソース

```
:listdef name=xmp4
vars='var1 var2 var3'.
:panel panel-attributes.
Example of Dynamic Column Headings
:list listdef=xmp4
depth='*'
headsize=colhsize.
:listcol var=var1
usage=inout
maxwidth=10
colhead='A B C D E F'.
:listcol var=var2
usage=inout
maxwidth=14
colhead='G H'.
```


LISTACT (リスト・アクション)



リスト・アクション (LISTACT) タグは、リストの個々の項目に対して行われる操作を定義します。このタグは、表示パネルでのみ許可されます。ユーザーが入力したオプションは、Enter または Prompt 機能キーが押されるまで、実際には実行されません。そのため、ユーザーは、Page Up または Page Down キーを使用しながら、複数のリスト項目を選択することができます。

ACTOR=UIM がリスト域 (LIST) タグに指定される場合、LISTACT タグで指示されるすべてのアクションが UIM によって実行されます。これらのアクションには、ENTER 属性が指定されている必要があります。また、PROMPT および USREXIT 属性を指定することも可能です。ACTOR=CALLER が LIST タグに指定されている場合、アクションは Display Panel (QUIDSPP) API を呼び出したプログラムによって扱われます。この場合、以下の属性のどれも LISTACT タグに指定することはできません。

- CONFIRM
- ENTER
- PROMPT
- USREXIT
- EXTENTER
- EXTPROMPT
- NOEXT
- EXTMSGID
- EXTMSGF

UIM にいくつかのアクションを実行させ、呼び出し側プログラムにその他のアクションを実行させる方法はありません。

CMD ダイアログ・コマンドが、処置リスト内の拡張アクション項目について実行されるアクションとして使用される場合、コマンド・テキストは System/36 Environment OCL ではなく、制御言語コマンドでなければなりません。コマンド・テキスト内の値 *N は、ブランク値を持つ拡張アクション項目のダイアログ変数の代わりとなります。例外となるのは、このタグに NOEXT=MSG が指定される場合です。これは、拡張アクション項目の値のいずれかがブランクであれば、コマンドをサブミットすべきでないことを示します。この考慮事項は、このタグの以下の属性に適用されます。

- ENTER
- PROMPT
- EXTENTER
- EXTPROMPT

必須属性

HELP=*help-module-name*

このリスト・アクションに関連したヘルプ・モジュールを識別します。すべてのアクティブな条件付きリスト・アクションに関するオンライン・ヘルプ情報が、常にアクション列に関するオンライン・ヘルプ情報に追加されます。

OPTION=*number*

リスト内のこの項目について選択されるオプションに関連した値。値は 1 から 999 の範囲の整数値でなければなりません。アクション・フィールド変数は、クラス定義 (CLASS) タグの BASETYPE を ACTION にして定義する必要があります。アクション・フィールドの値は、UIM によって自動的に有効なオプションに制限されます。アクティブな LISTACT タグがないアクション・フィールドに数値を入力すると、エラーが発生します。

オプションの属性

ACTFOR=BOTH | LISTE | EXTACTE

定義されたリスト・アクションが既存のリスト項目、拡張アクション項目、またはその両方に許可されるかどうかを示します。デフォルトは BOTH です。

LISTE が指定される場合、定義されたアクションは、画面上の既存のリスト項目に関連したアクション・フィールドでのみ有効です。

EXTACTE は、定義されたアクションが、拡張アクション項目のアクション・フィールドでのみ有効であることを示します。たとえば、ACTFOR=EXTACTE は、Option 1=Create の場合に使用されます。なぜなら、作成オプションは既存の項目には許可されないからです。

CONFIRM=*panel-name*

リスト・アクションが実行される前に表示される確認パネル。確認パネルでは、リスト・アクションを確認するか、確認しないかのオプションがユーザーに提供される必要があります。アクションが確認される場合、このタグの ENTER、EXTENTER、および USREXIT 属性から該当するアクションが実行されます。アクションが確認されない場合、それらのアクションはいずれも実行されません。

このタグの PROMPT および EXTPROMPT 属性でコード化されるアクションは確認されません。PROMPT および EXTPROMPT アクションの結果として表示されるプロンプト画面は確認パネルとして働き、ユーザーが考えを変えて、アクションを取り消すことができますようにします。確認パネルは、このパネル・グループ内で定義された別のパネルでなければなりません。

CONFIRM 属性は、ACTOR=UIM が LIST タグに指定されている場合にのみ許可されます。

LISTACT タグ

CONFIRM 属性が指定されている場合、EXTACT=YES が LIST タグに指定され、ACTFOR=BOTH または ACTFOR=EXTACTE がこのタグに指定されている場合、NOEXT=MSG もこのタグに指定する必要があります。

単一の処置リストについて最高 20 個の異なる LISTACT タグが CONFIRM 属性を指定することがあります。これは、アクションのいずれか 1 つに適用されるリスト項目の数に制限を課すものではありません。

この属性の必須の規則および推奨される規則について詳しくは、619 ページの『確認パネルの要件』および 619 ページの『確認パネルの規則』を参照してください。

ENTER='action-text'

ENTER ダイアログ・コマンドを介してリスト・アクションが要求されるときに取られるアクションを指定します。この属性は、ACTOR=UIM が LIST タグに指定される場合は必須です。action-text の有効な形式を以下に示します。

- 'CALL program-reference' UIM と処置リスト・オプション用の出口プログラム間のインターフェースの説明については、iSeries Information Center の『APIs』のトピックを参照してください。
- 'CMD command-string' コマンド・ストリング内のダイアログ変数名は、変数置換を示すために、前にアンパーサンドが付いており、ピリオドで終わらなければなりません。

それぞれのアクションの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

この属性は長いコマンド・ストリングの構造を許可しているため、反復可能です。

LIST タグの PARMS 属性は、CMD ダイアログ・コマンドのコマンド・ストリングに置換できるダイアログ変数を識別します。CMD ダイアログ・コマンドを使用するリスト・アクションのために、追加の関数が提供されます。コマンド行のパラメーターがコマンド・ストリングに置換されるときに、コマンド・ストリング内のパラメーターの前に "?<" または "??" 選択プロンプト文字が付く場合、コマンド行のパラメーターはコマンド・ストリングで定義された同じパラメーターをオーバーライドします。

この関数により、コマンドのプロンプトはコマンド行からのパラメーター値を示したり、オーバーライドするパラメーターがコマンド行にないときにオブジェクトの現行パラメーター値を示すことができます。"?<" または "??" 選択プロンプト文字が使用される場合、現行パラメーター値をコマンド・ストリングに置換することはご自身で行ってください。

EXTENTER='action-text'

ENTER ダイアログ・コマンドを介して拡張アクション・フィールドのリスト・アクションが要求されるときに取られるアクションを指定します。この属性は、ACTFOR=LISTE が LISTACT タグに指定されているとき、または EXTACT=NO が LIST タグに指定されているときには許可されません。

この属性の構文および使用法は、このタグの ENTER 属性と同じです。EXTENTER 属性が指定されない場合、ENTER 属性に指定されたアクションが使用されます。

PROMPT='action-text'

PROMPT ダイアログ・コマンドを介してリスト・アクションが要求されるときに取られるアクションを指定します。アクション・テキストの有効な形式を以下に示します。

- 'CALL program-reference' UIM と処置リスト・オプション用の出口プログラム間のインターフェースの説明については、iSeries Information Center の『APIs』のトピックを参照してください。
- 'CMD command-string' コマンド・ストリング内のダイアログ変数の前には、変数置換を示すアンパーサンドが付いていなければなりません。

これらのアクションの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

この属性は長いコマンド・ストリングの構造を許可しているため、反復可能です。

LIST タグの PARMs 属性は、CMD ダイアログ・コマンドのコマンド・ストリングに置換できるダイアログ変数を識別します。CMD ダイアログ・コマンドを使用するリスト・アクションのために、追加のコマンド実行依頼サポートが活動化されます。コマンド行のパラメーターがコマンド・ストリングに置換されるときに、コマンド・ストリング内のパラメーターの前に "?<" または "??" 選択プロンプト文字が付く場合、コマンド行のパラメーターはコマンド・ストリングで定義された同じパラメーターをオーバーライドします。

この関数により、コマンドのプロンプトはコマンド行からのパラメーター値を示したり、オーバーライドするパラメーターがコマンド行にないときにオブジェクトの現行パラメーター値を示すことができます。"?<" または "??" 選択プロンプト文字が使用される場合、現行パラメーター値をコマンド・ストリングに置換することはご自身で行ってください。

EXTPROMPT='action-text'

PROMPT ダイアログ・コマンドを介して拡張アクション・フィールドのリスト・アクションが要求されるときに取られるアクションを指定します。EXTPROMPT 属性は、ACTFOR=LISTE が LISTACT タグに指定されているとき、または EXACT=NO が LIST タグに指定されているとき、あるいは PROMPT 属性がこのタグに指定されていないときには許可されません。

この属性の構文および使用法は、このタグの PROMPT 属性の場合と同じです。EXTPROMPT 属性が指定されない場合、PROMPT 属性に指定されたアクションが使用されます。

NOCMD=ENTER | PROMPT

コマンド行がブランクのときに UIM が ENTER ダイアログ・コマンドについて実行するアクションを指定します。

ENTER は、このタグの ENTER 属性に対するアクションが実行されることを示します。

PROMPT は、このタグの PROMPT 属性に対するアクションが実行されることを示します。

これにより、コマンド行が空のときにはコマンド・プロンプトを表示させ、コマンド行にパラメーターが含まれている場合には、プロンプトを表示させないようにすることができます。

この属性がこのタグの NOEXT 属性と相互作用する様子の説明については、表 64を参照してください。

NOEXT=ENTER | PROMPT | MSG

拡張アクション項目の 1 つ以上の入力フィールドがブランクであるときに、UIM が ENTER ダイアログ・コマンドについて実行するアクションを指定します。

ENTER は、このタグの EXTENTER 属性に対するアクションが実行されることを示します。

PROMPT は、このタグの EXTPROMPT 属性に対するアクションが実行されることを示します。

MSG は、アクションが実行されず、このタグの EXTMSGID 属性に指定されたメッセージがユーザーに送信されることを示します。MSG が指定されるときには、EXTMSGID 属性が必須です。

この属性が NOCMD 属性と相互作用する様子の説明については、表 64を参照してください。次の表は、コマンド行がブランクで、さらに拡張アクション項目の 1 つ以上の入力フィールドがブランクであるときに、NOCMD と NOEXT 属性が作用する様子を示しています。

表 64. NOCMD と NOEXT 属性の相互作用

NOCMD	NOEXT	結果
ENTER	ENTER	EXTENTER 属性が使用される
PROMPT	ENTER	EXTPROMPT 属性が使用される
ENTER	PROMPT	EXTPROMPT 属性が使用される
PROMPT	PROMPT	EXTPROMPT 属性が使用される

LISTACT タグ

表 64. NOCMD と NOEXT 属性の相互作用 (続き)

NOCMD	NOEXT	結果
ENTER	MSG	EXTMSGID メッセージが表示される
PROMPT	MSG	EXTMSGID メッセージが表示される

この表は、次のすべてが当てはまる場合にのみ使用されます。

1. リスト・アクション・オプションが拡張アクション項目に対して処理されている。
2. コマンド行がブランクである。
3. 拡張アクション項目の 1 つ以上の入力フィールドがブランクである。

USREXIT='CALL program-reference'

このタグの ENTER、EXTENTER、PROMPT、または EXTPROMPT 属性のいずれかで定義されているアクションが実行された後で、UIM がこのリスト項目を更新または削除するために呼び出すリスト出口プログラムを指定します。プログラムには、オプションが成功したか失敗したかを示す内容を含む情報が渡されます。

CALL ダイアログ・コマンドの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

UIM と処置リスト用のプログラムの間のインターフェースの説明については、iSeries Information Center の『APIs』のトピックを参照してください。

EXTMSGID='message-identifier'

拡張アクション項目の 1 つ以上の入力フィールドがブランクであるため、このタグによって定義されたアクションが実行できない場合に表示されるメッセージのメッセージ ID。この属性は、NOEXT=MSG がこのタグに指定されているときには必須です。それ以外の場合は許可されません。

EXTMSGF='qualified-message-file-name'

メッセージ ID が入っているメッセージ・ファイル名。この属性を指定できるのは、EXTMSGID 属性をこのタグに指定する場合だけです。パネル・グループ (PNLGRP) タグに DFTMSGF 属性が指定されていない場合で、このタグに EXTMSGID 属性が指定されている場合は、この属性を指定する必要があります。

COND=condition-name

指定された条件が真の場合にのみ、リスト・アクションはパネル上で有効になります。条件は、パネル・グループ・プロローグ内で条件定義 (COND) タグを使用して定義する必要があります。すべての LISTACT タグの条件が偽の場合、アクション列は引き続き画面に表示されますが、オプション番号は有効ではありません。リストに関して、オプションが引き続き有効であるかどうかを確認することは、ご自身で行ってください。

AVAIL=condition-name

指定された条件が真と評価される場合にのみ、リスト・アクションは使用可能です。条件は、パネル・グループ・プロローグ内で COND タグを使用して定義する必要があります。

使用不可のアクションは、アクション記述の最初の部分をアスタリスク (*) がオーバーレイした状態で表示されます。使用不可のリスト・アクションについて、オンライン・ヘルプ情報が表示されます。

条件が真の場合、リスト・アクションは使用可能です。条件が偽の場合、リスト・アクションは使用不可です。COND 属性で指定されている条件があれば、その条件がこの属性より優先して使用されます。

AVLMSGID=message-identifier

数値が入力されているが、このタグの AVAIL 属性で指定されたとおりにそれを使用できないときに表示されるメッセージのメッセージ ID。この属性を指定できるのは、AVAIL 属性が指定されている場合だけです。

この属性が指定されない場合、UIM はオプション番号が現在使用可能でないことを告げるデフォルト・メッセージを表示します。

AVLMSGF='qualified-message-file-name'

メッセージ ID が入っているメッセージ・ファイル名。属性を指定できるのは、このタグの AVLMSGID 属性が指定されている場合だけです。DFTMSGF 属性を PNLGRP タグに指定しない場合で、このタグの AVLMSGID 属性を指定するときには、この属性を指定する必要があります。

オプションのテキスト

action-description

リスト域の最上部指示の一部として表示されるテキスト。テキストが提供されない場合、このアクションに関するリスト域の指示部には、何も表示されません。テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみを含めることができます。

指示行は、さまざまなリスト・アクションからのテキストを 3 スペースで区切って連結することにより、形成されます。テキストが行の残りの部分に収まらない場合は、次の行に置かれます。

確認パネルの要件

このタグの CONFIRM 属性で指定されたパネルは、以下の規則に従っていなければなりません。

- 確認パネルの表示パネル (PANEL) タグ上の ENTER 属性は、'RETURN 100' でコード化されていなければなりません。これは、アクションがユーザーによって確認されることを UIM に示します。
- キー・リスト項目 (KEYI) タグ上の ACTION 属性は、RETURN ダイアログ・コマンドに割り当ててはできません。これは、UIM はパネルの表示を処理するため、戻り値を解釈できないことによります。
- 確認パネルにはリスト域がなければなりません。
- 確認リストとして使用されるリストは、処置リスト・パネルに表示されるリストと同じであってはなりません。
- 確認パネルには、メニュー・バー、処置リスト、コマンド行、またはメニュー域があってはなりません。

確認パネルの規則

このタグの CONFIRM 属性で指定されたパネルは、以下の規則に従っているべきです。

- パネルについて説明するタイトル・テキストおよび最上部指示が必要です。
- 元の処置リストで定義されたものと同じ列 (またはそのサブセット) を含むリスト定義を持つ、単一のリスト域があるべきです。
- リストはスクロール可能であるべきです。
- 処置リストに複数のビューがある場合、確認リストにも複数のビューが必要であり、同じダイアログ変数が両方のリスト上のビュー変数に使用される必要があります。確認パネルには、CHGVIEW ダイアログ・コマンド用に定義された機能キーが必要です。両方のリストに同じビュー変数が使用される場合、確認パネル上のビュー操作を変更して、処置リスト・パネルが再表示されるときにそのパネルのビューを変更することができます。
- ENTER、CANCEL、PAGEDOWN、PAGEUP、HELP、および PRINT ダイアログ・コマンドの機能キー定義が必要です。

LISTACT タグ

- 汎用出口プログラムが定義される場合、処置リストへの変更が行われないようにし、また元の処置リスト・パネルの再表示に影響を与える条件値を変更しないようにすべきです。
- 確認パネルで使用されるリストは、確認パネル以外では使用すべきではありません。UIM はそのリストの内容を削除および変更するためです。複数の確認パネルが UIM アプリケーション内で使用される場合、それらは同じリスト定義を使用します。
- 処置リスト・パネルのヘルプ情報の大半は、確認パネルで使用できますが、アクション列および拡張ヘルプに対して、個別にヘルプ情報を提供する必要があります。
- CLASS タグの BASETYPE を ACTION に指定して定義されたリスト・オプション変数のクラス定義は、CLASS タグに WIDTH 属性を持っており、処置リスト・パネル上に最大のリスト・アクション・オプションを収容するべきです。たとえば、処置リスト・パネル上で最大のオプション番号が 12 の場合、アクション変数のクラス定義に WIDTH=2 が指定されているべきです。これにより、オプション番号は、確認パネル上の列見出しの下の、以前に処置リスト・パネルで表示されていた場所と同じ位置に表示されます。
- 拡張リスト・アクション項目の入力可能フィールドの値だけが確認パネルに表示されます。その他の列はすべて、そこに値があるとしてもブランクで表示されます。

例: リスト・アクション

この例は、画面上のアクション・オプションが定義される様子を示します。

UIM ソース

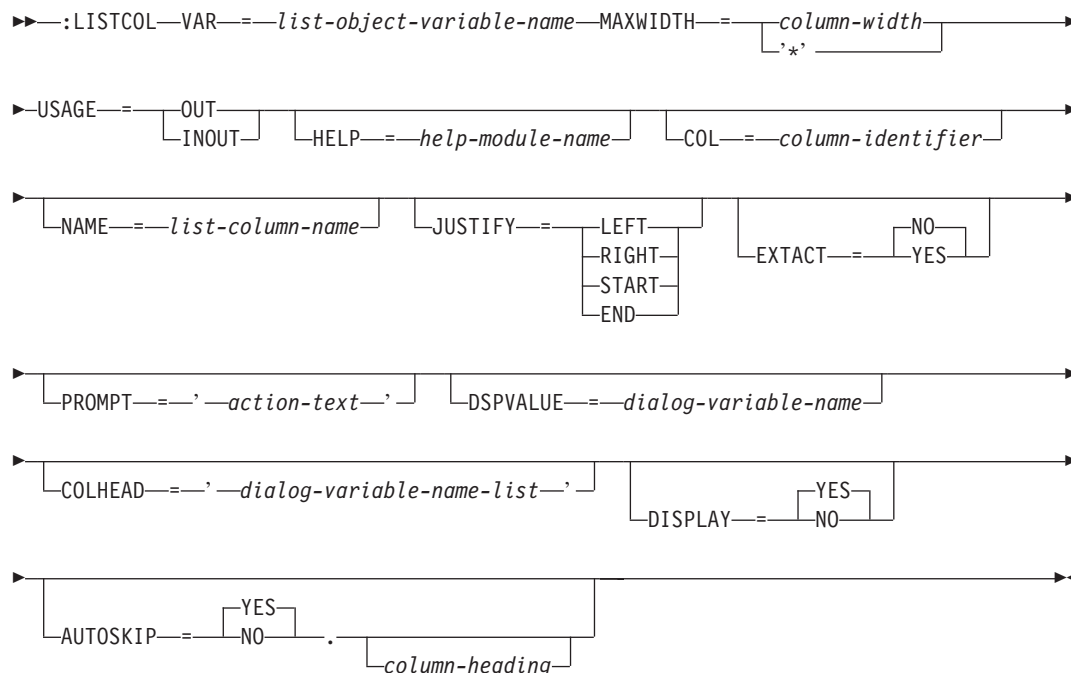
```
:listact option=2
    help='option/2'
    enter='CALL CHGPGM'
    .2=Change
:listact option=4
    help='option/4'
    enter='CALL DLTPGM'
    confirm=CONFDLT
    .4>Delete
:listact option=5
    help='option/5'
    enter='CALL DSPPGM'
    .5=Display
```

結果

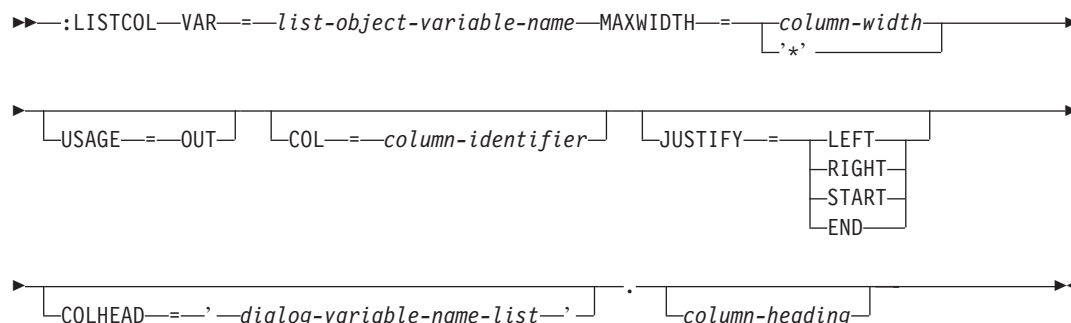
```
Type choices, press Enter.
 2=Change 4>Delete 5=Display
```

LISTCOL (リスト列)

表示パネルの構文:



印刷パネルの構文:



リスト列 (LISTCOL) タグは、複数のリスト・ビューのいずれかに属している列を定義します。このタグは、表示パネルおよび印刷パネルで許可されます。これはダイアログ変数、フィールドがユーザーによるデータの入力を許可するか、それとも出力専用か、列幅、および列見出しを指定します。

必須属性

VAR=*list-object-variable-name*

この列に表示する基礎となるリスト・オブジェクトの変数の名前。

MAXWIDTH=*column-width* | '*'

- | 列の最大幅。幅については、列見出しを他の言語に変換することを考慮に入れることができます。列見出しの最小サイズは 4 文字です。クラス定義 (CLASS) タグで定義された値より長い幅を指定すると、列見出しのフォーマットに余裕ができますが、より多くのデータをフィールドに入力できるわけで

LISTCOL タグ

はありません。CLASS タグのクラス幅より短い幅を指定することは、CHAR、IGC、または特定の TIME 変数にのみ許可され、入力列には許可されていません。

TIME 変数の時間帯部分のみ切り捨てることができます。そのため、TIME 変数のクラス定義の BASETYPE に ZONE オプションが指定されていなければなりません。また、LISTCOL に指定される幅は 8 以上でなければなりません。

'*' がコード化されている場合、領域幅の残りが列に使用され、その列の中の値は切り捨てることができます。1 つの列のみ、'*' をコード化できます。'*' が指定されている列がすべてのビューの最後の列ではない場合、警告メッセージがコンパイラー・リストに書き込まれます。2 から 5 文字の区切り記号が列の間で保守されます。

USAGE=OUT | INOUT

列の表示の使用法。この属性は表示パネルには必要ですが、印刷パネルではオプションです。

USAGE=OUT は、出力データ列を定義します。OUT は、表示される変数が出力専用であり、ユーザーが変更できないことを示します。

USAGE=INOUT は、データ入力列を定義します。INOUT は、変数がデータ入力用であり、ユーザーが変更できることを示します。USAGE=INOUT フィールドは印刷パネルでは許可されませんが、アクティブ・リストのアクション列には必須です。

オプションの属性

HELP=*help-module-name*

このリスト内の列の目的を説明するオンライン・ヘルプ情報を識別します。この属性は表示パネルにのみ許可されます。ヘルプ・モジュールの名前は別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説された名前の規則に従っていなければなりません。名前の規則については詳しくは、515 ページの『名前構文』を参照してください。

この列がリスト・グループ (LISTGRP) タグで定義されたリスト列グループの一部である場合、HELP 属性は使用できません。しかし、列がグループの一部でない場合はこの属性が必要です。

COL=*column-identifier*

列の ID。リスト・ビュー (LISTVIEW) タグの COLS 属性で参照できます。ID が提供されていない場合、デフォルトでこのタグの VAR 属性で指定されたダイアログ変数に設定されます。これらの名前は、その他のすべての名前と別のネーム・スペースにあり、パネル・グループではなく、リスト域内でのみ意味があります。

列が LISTGRP タグで定義されたリスト列グループの一部である場合は、リスト列 ID を指定できず、デフォルトの列名も想定されません。

NAME=*list-column-name*

列に関連した名前。命名規則については詳しくは、515 ページの『名前構文』を参照してください。

この名前は、Add Pop-Up Window (QUIADDPW) API とともに使用して、このリスト列の近くにウィンドウを配置することができます。この属性は表示パネルにのみ許可されます。

リスト列が LISTGRP タグで定義されたリスト列グループの一部であり、LISTCOL タグと LISTGRP タグの両方で NAME 属性が指定されている場合、カーソルがリスト列内の任意の場所に置かれているときに、リスト列名は表示パネル (PANEL) タグの CSRNAME 属性で指定されたダイアログ変数内に設定されます。カーソルが列の間、または指定されたリスト・グループ内の不特定のリスト列に置かれている場合、リスト・グループ名は PANEL タグの CSRNAME 属性について設定されます。CSRNAME 属性の説明については、649 ページの『PANEL (表示パネル)』を参照してください。

JUSTIFY=LEFT | RIGHT | START | END

ダイアログ変数の値がリスト列内で左揃えされるか、右揃えされるかを示します。このタグの ALIGN 属性が省略される場合、デフォルトでは数値ダイアログ変数の値を右揃えし、非数値ダイアログ変数を左揃えします。

ダイアログ変数が左揃えされているときには、先行ブランクは保持されます。右揃えではすべての末尾ブランクを取り外します。数値ダイアログ変数を編集するための UIM 規則により、先行ブランクを含まない表示可能な値が生成されます。

位置調整のデフォルトをオーバーライドできる一般的な例は、BASETYPE が CLASS タグで数値として定義されている値の列の場合です。しかし、その列の表示可能な値は時折、変換リストから生成される特殊値であることがあります。変換リストについて詳しくは、686 ページの『TL (変換リスト)』を参照してください。

START は LEFT の同義語で、END は RIGHT の同義語です。

EXTACT=NO | YES

このリスト列が拡張アクション項目で入力可能かどうかを指定します。この属性は表示パネルにのみ許可されます。

EXTACT=NO は、このリスト列が拡張アクション項目に入力可能フィールドを持たないことを指定します。

EXTACT=YES は、このリスト列が拡張アクション項目に入力可能フィールドを持つことを指定します。EXTACT=YES がリスト域 (LIST) タグで指定される場合、EXTACT=YES を各リスト・ビュー内の少なくとも 2 列 (アクション列を含む) に指定する必要があります。

EXTACT=YES が指定されている列は、リスト域のすべてのビューになければなりません。

PROMPT='action-text'

PROMPT ダイアログ・コマンドを介して F4=List が要求されるときに取られるアクション。この属性は表示パネルにのみ許可されます。この属性は、CLASS タグの BASETYPE を ACTION にして定義されたダイアログ変数、またはこのタグの USAGE=OUT リスト列には許可されません。アクション・テキストの有効な形式を以下に示します。

- 'CALL program-reference' UIM と、カーソル移動に影響されるプロンプト用の出口プログラムの間インターフェースの説明については、iSeries Information Center の『APIs』のトピックを参照してください。
- 'RETURN positive-integer'

これらのアクションの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

DSPVALUE='dialog-variable-name'

カーソルが置かれているリスト項目に入力された現行データを含むダイアログ変数。この属性は表示パネルにのみ許可されます。また、このタグで PROMPT 属性がコード化されるときに許可されます。この変数は、変数プールが更新されるかどうかにかかわらず、更新されます。変数プールは、キー項目 (KEYI) タグまたはプルダウン・フィールド選択項目 (PDFLDC) タグの VARUPD 属性に基づいて更新されます。この更新は、変数プールの通常の更新とは別に行われます。

指定されるダイアログ変数は、CLASS タグで CHAR または IGC 変数として定義される必要があります。それらの長さは、このタグの VAR 属性に指定されたダイアログ変数の幅と同じです。値がこの変数に置かれる前には、この値の変換処理または妥当性検査は実行されません。このタグの VAR 属性で指定された変数のクラスが、文字セットとコード・ページの変換を実行すべきであると指定する場合、この変数について文字セットとコード・ページの変換が実行されます。

LISTCOL タグ

COLHEAD='dialog-variable-name-list'

ブランクで区切られたダイアログ変数のリスト。リスト域内のリスト項目の上に置かれた列見出しを指定します。10 個までのダイアログ変数を指定できます。これらの変数では、CLASS タグで Basetype を CHAR または IGC として指定している必要があります。また、このタグの VAR 属性に指定されたダイアログ変数のクラス幅より大きいか等しく、このタグの MAXWIDTH 属性に指定された幅より小さいか等しくなければなりません。単一の COLHEAD 属性に指定されるすべての変数は、同一のクラスで定義する必要があります。

それぞれのダイアログ変数は、列見出しテキストの行として使用されます。UIM はこれらのダイアログ変数の内容の桁揃えまたは位置調整を行いません。

この属性が使用される場合、列見出しテキストをこのタグのピリオドの後に指定することはできません。また、LIST タグに MAXHEAD属性を指定する必要はありませんが、LIST タグの HEADSIZE 属性は指定する必要があります。

この属性は、リスト列グループの一部であるリスト列では使用できません。

DISPLAY=YES | NO

パネルが表示されるときにリスト列のフィールドが見えるかどうかを示します。この属性は表示パネルにのみ許可されます。YES は、リスト列のフィールドが見えることを示します。

NO は、リスト列のフィールドが見えないことを示します。リスト内の列について NO を指定できるのは、USAGE=INOUT がこのタグに指定されており、ACTOR=NONE が LIST タグに指定されている場合に限りです。

AUTOSKIP=YES | NO

リスト列フィールドの最後の位置にデータが入力されるときに、カーソルを次の入力フィールドに自動的に移動するかどうかを示します。この属性は表示パネルにのみ許可されます。YES がデフォルトです。

YES は、カーソルが自動的に次の入力フィールドに移動することを示します。

NO は、カーソルが自動的に次の入力フィールドに移動しないことを示します。リスト内の列について NO を指定できるのは、USAGE=INOUT がこのタグに指定されており、ACTOR=NONE が LIST タグに指定されている場合に限りです。

オプションのテキスト

column-heading

リスト域の予約済み部分で、列の上部に置かれる列見出し。列見出しまたは COLHEAD 属性が指定されていない場合、列見出しは表示されません。

テキストは複数の行に表示されることがあります。また、テキスト反転 (RT) タグのみを含めることができます。

列見出しのそれぞれの語は、新規の行に置かれます。列見出しの行に複数の語が必要な場合は、アポストロフィ (') で囲む必要があります。それぞれの語または引用符付きストリングは、このタグの MAXWIDTH 属性で定義された最大列に収まらなければなりません。許可された語または引用符付きストリングの最大数は、MAXHEAD 属性によって LIST タグに定義されます。リスト域に MAXHEAD=0 が使用される場合は、列見出しは許可されません。

列見出しがテキストとして指定される場合に限り、それはこのタグの JUSTIFY 属性に従って位置調整されます。このタグに COLHEAD 属性が指定されている場合、位置調整は行われません。

列見出しテキストまたは COLHEAD 属性が領域の LISTCOL タグに指定されていない場合、見出し情報用の画面上で行は予約されません。

必須属性

NAME=*list-name*

リスト・オブジェクトの名前。この名前はパネル・グループ内で固有でなければなりません。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

VARS='*dialog-variable-list*'

リストの列を構成する、50 個までの変数のリスト。このリスト内のすべてのダイアログ変数は、このパネル・グループで変数定義 (VAR) タグを使用して以前に定義されている必要があります。これらの変数のすべてを、リスト・ビュー (LISTVIEW) タグで定義されたビューに表示する必要はありません。リスト内の変数名はブランクで区切られます。

UIM は、VARS 属性で指定され、クラス定義 (CLASS) タグで ACTION の BASETYPE が定義されている変数を識別することにより、どの列が処置リスト処理または選択リスト処理に使用されるかを自動的に判別します。NONE 以外の ACTOR または SELECT 値が指定される場合、またはリスト域 (LIST) タグで SELECT=SINGLE または MULTI が指定される場合は、リスト・ビューごとにアクション列が必要です。リストの列の 1 つだけをアクション列として指定できます。

オプションの属性

CHGVAR=*dialog-variable-name*

ユーザーが入力したリスト項目値を、すでにリスト項目にある値と比較する必要があることを指定します。また、この属性は、比較の結果を保管する場所も示します。指定されるダイアログ変数は、このタグの VARS 属性にリストされた変数の 1 つでなければなりません。また、CLASS タグの BASETYPE を CHAR 1 にして定義する必要があります。

リスト値は、その表示形式ではなく、各ダイアログ変数の内部形式を使用して比較されます。リスト項目内の値が変更される場合、この属性に指定されたダイアログ変数に関連するリスト値は '1' に設定されます。新規リスト値がすべて旧リスト値と等しい場合には、特殊な処理は実行されません。この属性に指定されたダイアログ変数は変更されません。

注: ユーザーがリスト項目を変更したり、リストをスクロールしたり、またはその他のいくつかのファンクション・キーを使用するたびに、この属性に指定されるダイアログ変数は '1' に設定されます。リスト内の値が変更されてリストがスクロールされ、再び元の位置にスクロールして戻り、リスト項目が元の値に戻される場合、この属性に指定されたダイアログ変数に関連するリスト値は '1' に 2 回設定されます。

アプリケーション・プログラムは、この属性に指定されたダイアログ変数のリスト値を使用して、ユーザーによって変更されるリスト項目を判別します。リスト項目の値を '1' 以外のものに設定し、リセットすることは、アプリケーションが行います。これは、入力値が処理された後およびリストが再表示された後にリスト項目を更新して、比較が信頼できるようにすることにより行われます。

MSGID=*message-identifier*

UIM リストが空のときにユーザーに表示されるメッセージのメッセージ ID。メッセージ ID がこのタグの MSGID 属性または MSGIDVAR 属性で指定されていない場合、リストが空のときにメッセージは表示されません。MSGID 属性と MSGIDVAR 属性の両方を同じ LISTDEF タグに指定することはできません。

MSGIDVAR=*dialog-variable-name*

UIM リストが空のときにユーザーに表示されるメッセージのメッセージ ID を含むダイアログ変数。ダイアログ変数は、CLASS タグの BASETYPE を 'CHAR 7' にして定義する必要があります。メッセー

ジ ID がこのタグの MSGID 属性または MSGIDVAR 属性で指定されていない場合、リストが空のときにメッセージは表示されません。MSGID 属性と MSGIDVAR 属性の両方を同じ LISTDEF タグに指定することはできません。

MSGF='qualified-message-file-name'

MSGID 属性または MSGIDVAR 属性に指定されたメッセージ ID が入っているメッセージ・ファイル名。このタグに MSGID 属性または MSGIDVAR 属性が指定される場合に限り、この属性は許可されます。パネル・グループ (PNLGRP) タグに DFTMSGF 属性が指定されていない場合で、このタグに MSGID 属性または MSGIDVAR 属性が指定されている場合は、この属性を指定する必要があります。

PRTFLAG=dialog-variable-name

各リスト項目の最初に文字に印刷された文字を含む、リスト定義内のダイアログ変数の名前を指定します。印刷されたリスト項目ごとに、ダイアログ変数に保管された文字は最初に列に印刷されます。指定されるダイアログ変数は、このタグの VARS 属性にリストされた変数の 1 つでなければなりません。また、CLASS タグの BASETYPE を CHAR 1 に、幅を 1 にして定義する必要があります。この属性は、特殊文字を列 1 に印刷する必要があるリスト出力アプリケーションに使用されることを意図しています。

EMPHASIS='dialog-variable-emphasis-list.'

このタグの VARS 属性にリストされているそれぞれの変数に関連した強調または色を含む、ダイアログ変数のリスト。リスト内の項目の数は、VARS 属性に指定されたダイアログ変数の数と同じでなければなりません。リストされている変数ごとに 1 つの強調だけが許可されますが、特定の強調値が複数の変数に適用される場合があります。値 '*' は、通常の強調が VARS 属性にリストされている対応する変数とともに使用され、またどのリスト・ビューにも表示されないリスト列にも使用されることを示すために指定しなければなりません。

この属性に指定されたダイアログ変数は、このタグの VARS 属性にリストされた変数の 1 つにすることはできません。

リスト内の各変数は、CLASS タグで BASETYPE に CHAR 1 を指定して定義する必要があります。項目がリストに追加されるか、または既存の項目が更新されると、それらの変数の値により、その項目が表示されるときに使用される強調が決定されます。次の表は、強調を制御できる値を示しています。

表 65. 強調値

値	出力フィールド	入力フィールド
0	通常	通常
1	強調	強調
2	強調解除	通常

通常と強調解除は、モノクローム・デバイスでは同じです。

この属性にリストされているダイアログ変数に、上記の値のいずれかとは異なる値が指定されている場合、通常の強調が使用されます。この属性が指定されていない場合、通常の強調が使用されます。

この属性は、印刷時には強調を制御しません。

PROTECT='dialog-variable-usage-list'

このタグの VARS 属性にリストされている変数ごとに、リスト列 (LISTCOL) タグの USAGE 属性をオーバーライドするために使用される値を含むダイアログ変数のリスト。リスト内の項目の数は、VARS 属性に指定されたダイアログ変数の数と同じでなければなりません。リストされている変数ごとに 1 つの保護値だけが許可されますが、特定の保護値が複数の変数に適用される場合があります。

LISTDEF タグ

値 '*' は、VARS 属性にリストされている、対応する変数の LISTCOL タグの USAGE 属性をオーバーライドしないことを示すために指定する必要があります。値 '*' は、どのリスト・ビューにも表示されないリスト列にも使用されます。

この属性に指定されたダイアログ変数は、VARS 属性にリストされた変数の 1 つにすることはできません。

リスト内の各変数は、CLASS タグで BASETYPE に CHAR 1 を指定して定義する必要があります。項目がリストに追加されるか、または既存の項目が更新されると、それらの変数の値により、その項目が表示される際の保護が決定されます。この属性は、LISTCOL タグからの USAGE 属性をオーバーライドします。この属性に指定されたダイアログ変数に有効な値は以下のとおりです。

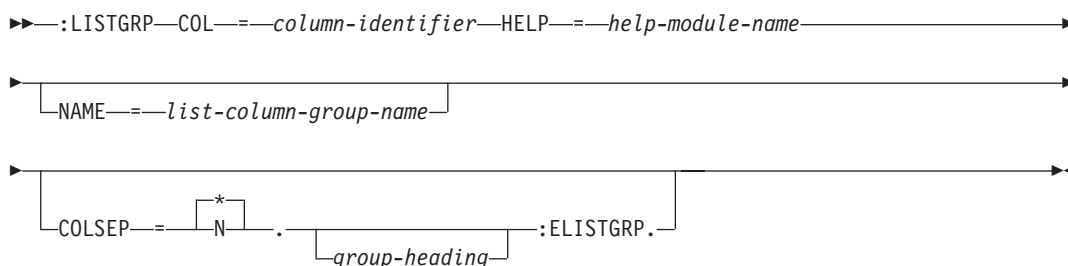
- '1' 変数の使用は出力のみです。
- '0' 使用は、LISTCOL タグの USAGE 属性によって制御されます。

処置リストまたは選択リストのリスト・オプション・フィールドを出力のみに変更しても、リスト処理には影響を与えません。アクション・フィールドにオプション番号または選択値が含まれていても、出力のみとして表示されている場合、オプションは処理されます。

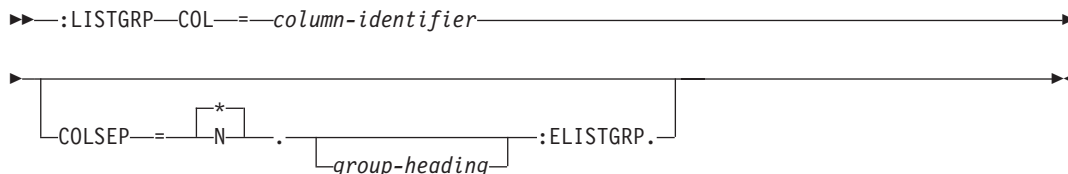
この属性にリストされているダイアログ変数が指定されていないか、または '1' か '0' 以外の値が指定されている場合、使用は LISTCOL タグの USAGE 属性によって制御されます。

LISTGRP (リスト列グループ)

表示パネルの構文:



印刷パネルの構文:



リスト列グループ (LISTGRP) タグは、すべての列に適用される単一見出しの下にリスト域内の列をまとめてグループ化します。このタグは、表示パネルおよび印刷パネルで許可されます。列は、どのリスト列グループにも属さないリスト域内で定義されます。このタグは、リスト域 (LIST) タグで MAXHEAD=0 が指定されている領域では使用できません。

列グループ内に存在できる列の種類には、以下のような制限があります。

- リストのアクション列は列グループに入れることができません。

- すべての列が列見出しを持つことができるか、またはどの列も列見出しを持つことができないかのどちらかです。
- 列グループはネストできません。
- 少なくとも 1 つの列が列グループ内で定義されている必要があります。

必須属性

COL=*column-identifier*.

列グループの ID。リスト・ビュー (LISTVIEW) タグの COLS 属性で参照されます。これらの名前は、その他のすべての名前と別のネーム・スペースにあり、パネル・グループ内ではなく、リスト域内でのみ意味があります。

HELP=*help-module-name*

このリスト内の列グループの目的を説明するオンライン・ヘルプ情報を識別します。この属性は表示パネルには必要ですが、印刷パネルでは許可されません。ヘルプ・モジュールの名前は別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説された名前の規則に従っていなければなりません。名前の規則については、515 ページの『名前構文』を参照してください。

オプションの属性

NAME=*list-column-group-name*

グループに関連した名前。この属性は表示パネルにのみ許可されます。命名規則については、515 ページの『名前構文』を参照してください。この名前は、このリスト列グループの近くにウィンドウを配置するために、Add Pop-Up Windows (QUIADDPW) API とともに使用することができます。

COLSEP=** | N*

リスト列グループで列を区切るためのスペースの数。COLSEP=* を指定すると、UIM は列区切り記号の幅を計算します。計算される幅は 2 から 5 の間のスペースです。COLSEP=N は任意の正数にすることができます。N の値を選択するときには、列の幅を覚えておいてください。この属性は、リスト列グループとその他の列リストの間のスペースには影響を与えません。また、リスト列グループ内の選択列の後に続くスペースにも影響を与えません。

オプションのテキスト

group-heading

リスト域の見出し部分で、列グループの上に置かれるグループ見出しテキスト。この見出しは、グループ内の列の上で 1 行に収まらなければなりません。見出しは列のグループ上に中央揃えされ、その見出しがまたがっている列の数が図形で示されます。列グループ内の列の見出しはさまざまな数の行にまたがる場合があります。グループ見出しは、グループの最も長い列見出しのすぐ上の行に表示されません。

列グループ内の列には、リスト列 (LISTCOL) タグで MAXWIDTH='*' が指定されないことがあります。列と、列の間の区切り記号が、グループ見出しに割り振られる幅を定義します。グループ見出しはこの幅を超えてはなりません。

グループの列のどこにも列見出しが指定されていない場合、列データのすぐ上にグループ見出しが表示されます。

テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみを含めることができます。

LISTGRP タグ

例: リスト列グループ

この例では、列区切り記号 1 を使用します。

UIM ソース

```
:listdef name=list1
  vars='seq colname more table text'.
:panel panel-attributes.Select and Sequence Columns
:list listdef=list1
  depth='*'
  maxhead=3.
:topinst.Type sequence numbers (1-999) to select
columns, press Enter.
:listcol var=seq
  usage=inout
  maxwidth=3
  help=hseq.Seq
:listgrp col=column
  colsep=1
  help=hlistgrp.
:listcol var=colname
  usage=out
  maxwidth=18.Column
:listcol var=more
  usage=out
  maxwidth=1.
:elistgrp.
:listcol var=table
  usage=out
  maxwidth=20
  help=htable.Table
:listcol var=text
  usage=out
  maxwidth=25
  help=htext.Text
:listview layout=1
  cols='seq column table text'.
:elist.
:epanel.
```

結果

```
                Select and Sequence Columns

Type sequence numbers (1-999) to select columns, press Enter.

Seq  Column                                Table  Text
---  ---                                ---    ---
---  EXTRALONGCOLUMNNAM > TABLE DESCRIPTION  TEXT DESCRIPTION
---  SHORTCOLUMNNAME          TABLE DESCRIPTION  TEXT DESCRIPTION

                                                                Bottom

F3=Exit  F4=Prompt  F12=Cancel
```

LISTVIEW (リスト・ビュー)



リスト・ビュー (LISTVIEW) タグは、リスト域に表示されるリストのビューの 1 つを定義します。このタグは、表示パネルと印刷パネルの両方で許可されます。

複数のリスト・ビューを領域で定義し、CHGVIEW ダイアログ・コマンドを使用して変更することができます。このダイアログ・コマンドは、機能キーまたはプルダウン選択項目に割り当てることができます。パネル上に複数のリスト・ビューがあり、CHGVIEW ダイアログ・コマンドが実行される場合、カーソルが置かれているリスト域がビューを変更します。

また、リスト域 (LIST) タグの VIEW 属性に指定されたダイアログ変数の値を変更することにより、ビューをプログラムで設定することができます。ビューには、0 から、この領域で定義された LISTVIEW タグの数より 1 小さい数まで連続して番号が付けられます。リスト域の最初の LISTVIEW タグがビュー 0 を定義し、2 番目の LISTVIEW タグがビュー 2 を定義します。このようにして、リスト域の各 LISTVIEW タグについて定義されます。

LIST タグの ACTOR 属性に NONE 以外の値がある場合、リストのアクション列は、パネルのリスト域のすべてのビューに表示される必要があります。LIST タグの SELECT 属性に NONE または MULTI 以外の値がある場合、リストのアクション列は、リスト域のすべてのビューに表示される必要があります。

複数のビューを印刷パネルに定義できる場合でも、Print Panel (QUIP RTP) API が呼び出されるときには、LIST タグの VIEW 属性によって識別されるビューだけが印刷されます。複数のビューを印刷するには、QUIP RTP API を複数回呼び出して、QUIP RTP の毎回の呼び出しの間にビュー・ダイアログ変数を更新する必要があります。

必須属性

COLS='column-identifier-list'

ブランクで区切られた列 ID または列グループ ID のリスト。このリストは、このビューの一部である列およびグループを識別します。

このビューで使用されるすべての変数は固有でなければなりません。

オプションの属性

LAYOUT=1 | number-of-layout-columns

複数の列セットが一度に画面に表示されるかどうかを示します。このタグの COLS 属性で定義される列の幅が小さい場合、複数の列セットを画面またはページに置くことができます。

複数列レイアウトに使用されるフォーマットは、画面全体またはページ幅を、等しいサイズのレイアウト列に分割します。このとき、レイアウト列の間には少なくとも 3 文字の区切り記号を挿入します。このタグの COLS 属性に指定されるすべてのリスト列は、単一のレイアウト列に収まらなければなりません。

表示パネル (PANEL) または印刷パネル (PRTPNL) タグで WIDTH=132 が指定されているすべてのパネルでは、1 から 10 までの値が許可されます。WIDTH が 132 より小さいパネルでは、1 から 6 までの値が許可されます。

LISTVIEW タグ

次の表は、LAYOUT 属性の値に応じて、使用可能な列幅と、それぞれのレイアウト列に使用される位置を示します。最初の表は、PANEL または PRTPNL タグで WIDTH=80 が指定されているすべてのパネルに適用されます。2 番目の表は、WIDTH=132 が指定されているパネルに適用されます。

表 66. Width=80 のレイアウト値： WIDTH=80 が指定されているパネルのレイアウト列幅と位置

レイアウト	レイアウト幅	レイアウト 1 列の位置	レイアウト 2 列の位置	レイアウト 3 列の位置	レイアウト 4 列の位置	レイアウト 5 列の位置	レイアウト 6 列の位置
1	78	2-79					
2	37	2-38	43-79				
3	24	2-25	29-52	56-79			
4	17	2-18	22-38	42-58	62-78		
5	13	2-14	18-30	34-46	50-62	66-78	
6	10	2-11	15-24	28-37	41-50	54-63	67-76

表 67. WIDTH=132 のレイアウト値： WIDTH=132 が指定されているパネルのレイアウト列幅と位置

レイアウト	レイアウト幅	レイアウト 1 列の位置	レイアウト 2 列の位置	レイアウト 3 列の位置	レイアウト 4 列の位置	レイアウト 5 列の位置	レイアウト 6 列の位置
1	130	2-131					
2	63	2-64	69-131				
3	41	2-42	46-86	90-130			
4	30	2-31	35-64	68-97	101-130		
5	23	2-24	28-50	54-76	80-102	106-128	
6	19	2-20	24-42	46-64	68-86	90-108	112-130

表 68. WIDTH=132 のレイアウト値： WIDTH=132 が指定されているパネルのレイアウト列幅と位置

レイアウト	レイアウト幅	レイアウト 7 列 の位置	レイアウト 8 列 の位置	レイアウト 9 列 の位置	レイアウト 10 列 の位置
7	16	97-112			
8	13	66-78	82-94		
9	11	86-96	100-110	114-124	
10	10	80-89	93-102	106-115	119-128

PANEL タグで指定されている WIDTH が 80 より小さいか、または width が 81 と 131 の間である場合、以下の算法を使用して レイアウト幅を決定します。

1. パネルの幅から 2 を減算する。
2. ステップ 1 の結果から $3 \times (\text{レイアウト}-1)$ を減算する。これが、レイアウト間の区切り記号の合計スペースです。
3. ステップ 2 の結果をレイアウトで除算する。切り上げは行いません。これがレイアウト幅です。
4. ステップ 3 の余りが (レイアウト-1) より大きい場合、レイアウト間の区切り記号は 4 です。そうでない場合、区切り記号スペースは 3 です。

LP (リストの一部)

▶▶:LP—.—*list-part-text*—▶▶

リストの一部 (LP) タグは、リストの一部に適用されるコメントまたは説明を識別します。このタグは情報域とヘルプ領域でのみ許可されます。リスト内の任意の場所に置くことができます。

LP タグの後に続くテキストは、リストの現行レベルの左マージンで開始します。番号や文字を振ることはできません。番号付きリスト (OL) タグとともに使用される場合、LP タグが番号付きシーケンスを中断したり、増やしたりすることはできません。

例: リストの一部

この例では、リストの一部タグを使用して、リストの流れが中断されないようにします。

UIM ソース

```
:ol.
:li.First item
:lp.This is a list part.
:li.Second item
:eol.
```

結果

1. First item

This is a list part.

2. Second item

MBAR (メニュー・バー)

▶▶:MBAR—NAME—==*menu-bar-name*—
 |HELP—==*help-module-name*—|
 |MAXBARL—

1
2
3

—.:EMBAR.—▶▶

メニュー・バー (MBAR) タグは、パネルのメニュー・バー域を定義します。メニュー・バーはパネルに最上部エレメントとして表示され、1 つ以上のメニュー・バー選択項目で構成されます。ユーザーがカーソルをメニュー・バー選択項目に置き、ENTER ダイアログ・コマンドを要求すると、選択されたメニュー・バー選択項目のプルダウン・メニューがメニュー・バー選択項目のテキストの下に表示されます。

表示されるプルダウン・メニューには、1 つ以上のプルダウン選択項目を持つ選択フィールドが含まれます。ユーザーは 1 つだけのプルダウン・メニュー選択項目を選択することが許可されます。各選択フィールドには 1 つ以上のプルダウン・メニュー選択項目があります。

すべてのメニュー・バーは、条件ステートメントの後およびキー・リストの前で定義する必要があります。

メニュー・バーがリスト・アクション列を持つパネルに指定される場合、スラッシュ (/) または各国指定選択文字がリスト・アクション列に入力され、メニュー・バー選択項目の処理対象となるエレメントを示しま

MBAR タグ

す。スラッシュは通常のリスト・アクション処理と同じ上から下への処理で処理されます。スラッシュ (/) は、処置リストの拡張アクション項目では許可されません。

他のタグを MBAR タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、各タグについて詳しく説明しているページを示しています。

表 69. MBAR タグと EMBAR タグの間で許可されるタグ

タグ名	順序	ページ
MBARC (メニュー・バー選択項目)	1	636

必須属性

NAME=menu-bar-name

メニュー・バーに割り当てられた名前。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

オプションの属性

HELP=help-module-name

メニュー・バーの目的を説明するヘルプ情報の名前。ヘルプ・モジュールの名前は別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説された名前の規則に従っていなければなりません。名前の規則について詳しくは、515 ページの『名前構文』を参照してください。

MAXBARL=1 | 2 | 3

メニュー・バーの記述に使用できる最大行数。1 から 3 行を指定できます。1 がデフォルトです。MAXBARL によって許可される最大数までの必要とされるメニュー・バーの行数だけが使用されます。すべてのメニュー・バー選択項目が、指定された行数に収まらない場合、コンパイラー・エラーが発生します。

例: メニュー・バー

この例は、「Work with Programs」パネルに表示されるメニュー・バーを定義します。

UIM ソース

```
.* Menu bar for Work with Programs panel
:mbar name=mbarpgm
      help='mbarpgm/'
      .
.*
.* Menu bar choice for the "File" pull-down
:mbarc help='mbarpgm/file'
      .File
:pdfld.
:pdfldc option=2
      help='mbarpgm/file/change'
      action='cmd ?CHGPGM PGM(&var2/&var1)'
      actfor=list
      .Change
:pdfldc option=4
      help='mbarpgm/file/delete'
      action='cmd DLTPGM PGM(&var2/&var1)'
      actfor=list
      confirm=confpgm
      usrexit='call exitpgm'
      .Delete
```

```

:pdfldc    option=5
           help='mbarpgm/file/display'
           action='cmd DSPPGM PGM(&var2/&var1)'
           actfor=list
           .Display
:pdfldc    option=6
           help='mbarpgm/file/exit'
           action=exit
           varupd=no
           .Exit
:pdaccel.F3
:epdfld.
:embarc.
.*
.* Menu bar choice for the "Help" pull-down
:mbarc    help='mbarpgm/help'
         .Help
:pdfld.
:pdfldc    option=1
           help='mbarpgm/help/helphelp'
           action=helphelp
           varupd=no
           .Help for help...
:pdfldc    option=2
           help='mbarpgm/help/help'
           action=exthelp
           varupd=no
           .Extended help...
:pdfldc    option=3
           help='mbarpgm/help/keyshelp'
           action=keyshelp
           varupd=no
           .Keys help...
:pdfldc    option=4
           help='mbarpgm/help/schidx'
           action=helpidx
           varupd=no
           .Help index...
:pdfldc    option=5
           help='mbarpgm/help/about'
           action='call logopgm'
           varupd=no
           .About...
:epdfld.
:embarc.
:embar.

```

結果

```

File  Help
-----
                                Work with Programs
                                System:  SYSTEM01
Select items in list, press F10 to select action.

Opt  Program    Library    Text
-    P11111111  L11111111  Description text
-    P22222222  L22222222  Description text
-    P33333333  L33333333  Description text
-    P44444444  L44444444  Description text

```

File プルダウンは以下に示すように表示されます。

MBAR タグ

```
File  Help
-----
:  2. Change      :          Work with Programs
:                :
:  4. Delete      :  ress F10 to select action.
:  5. Display     :
:  6. Exit        F3 : y      Text
:.....: LLLL  Description text
- P P P P P P P P 2 L L L L L L L L L L Description text
- P P P P P P P P 3 L L L L L L L L L L Description text
- P P P P P P P P 4 L L L L L L L L L L Description text
-
```

Help プルダウンは以下に示すように表示されます。

```
File  Help
-----
:  1. Help for help... : with programs
:  2. Extended help... :
Select :  3. Keys help...  : select action.
:  4. Help index...   :
Opt Pr :  5. About...      :
PP :.....: ption text
- P P P P P P P P 2 L L L L L L L L L L Description text
- P P P P P P P P 3 L L L L L L L L L L Description text
- P P P P P P P P 4 L L L L L L L L L L Description text
-
```

F10 (ACTIONS) を押すと、カーソルは最初のメニュー・バー選択項目の前のスペースに置かれます。

メニュー・バー選択項目が選択されると、カーソルはプルダウン・フィールドに置かれ、タブ・キーは最初の未選択のメニュー・バー選択項目に移動します。タブ・キーを繰り返し押すと、カーソルはメニュー・バー域を移動し、プルダウン・メニューに戻ります。

MBARC (メニュー・バー選択項目)

▶▶:MBARC—HELP—=*help-module-name*—.—*menu-bar-choice-text*—.EMBARC.—▶▶

メニュー・バー選択項目 (MBARC) タグは、メニュー・バー内の 1 つの選択項目を定義します。メニュー・バー選択項目の末尾は EMBARC タグで示されなければなりません。

他のタグを MBARC タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、各タグについて詳しく説明しているページを示しています。

表 70. MBARC タグと EMBARC タグの間で許可されるタグ

タグ名	順序	ページ
PDFLD (プルダウン・フィールド)	1	659
PDFLDC (プルダウン・フィールド選択項目)	2	660

必須属性

HELP=*help-module-name*

メニュー・バー選択項目の目的を説明するヘルプ情報の名前。ヘルプ・モジュールの名前は別のパネ

ル・グループからインポートされた名前にすることができますが、この章の前の部分で概説された名前の規則に従ってなければなりません。名前の規則について詳しくは、515 ページの『名前構文』を参照してください。

必須テキスト

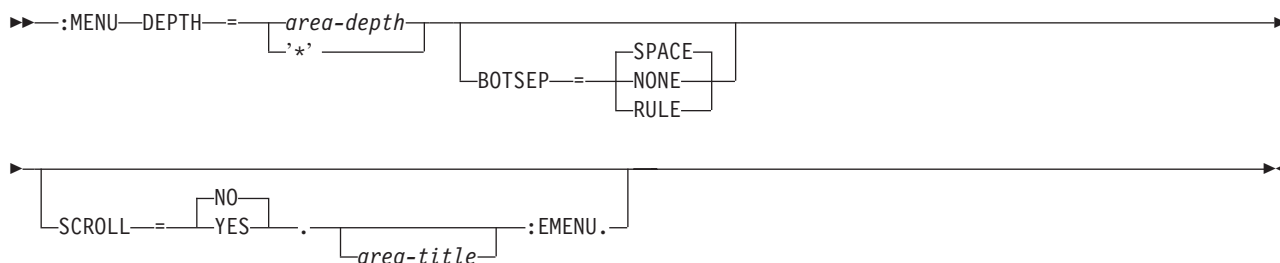
menu-bar-choice-text

メニュー・バー選択項目の表示可能な記述。テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみを含めることができます。テキストは、32 バイトを超えてはなりません。

メニュー・バーは、さまざまなメニュー・バー選択項目からのテキストを連結することにより、形成されます。テキストが行の残りの部分に収まらない場合は、次の行に置かれます。メニュー・バーは、MBAR タグの MAXBARL 属性で指定された行数に制限されます。

テキストは、その前にタブ移動可能なスペース付きで表示されます。メニュー・バー選択項目のいずれかにカーソルを置くためにタブ・キーを使用すると、テキストから 1 文字左にカーソルが表示されます。メニュー・バーの各テキスト・フィールドの間には 2 スペースが挿入されます。その結果、選択項目の間には 3 スペースが見えることになります。

MENU (メニュー域)



メニュー域 (MENU) タグは、メニュー域を定義します。このタグは、表示パネルでのみ許可されます。メニュー域の末尾には EMENU タグがなければなりません。

メニュー域は、パネルで定義されたその他すべての領域とともにパネルに組み込まれます。どのパネルでも、メニュー域は 1 つだけ存在することができます。

メニュー・パネルには、コマンド行 (CMDLINE) タグかオプション行 (OPTLINE) タグのいずれかを指定する必要があります。ユーザーは、コマンド行またはオプション行にオプション番号を入力することにより、メニュー・オプションを選択します。

他のタグを MENU タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

MENU タグ

表 71. MENU タグと EMENU タグの間で許可されるタグ

タグ名	順序	使用法	ページ
TOPINST (最上部指示行)	1	D	687
APPFMT (アプリケーション・フォーマット済み領域)	2	D	518
MENUGRP (メニュー列グループ)	3	D	641
MENUI (メニュー項目)	3	D	642
BOTINST (最下部指示行)	4	D	521

必須属性

DEPTH=area-depth | '*'

領域の深さを行で示したものの。区切り記号が指定されていれば、それも含まれます。

'*' が指定されている場合、その他のすべてが割り振られた後で画面に残っているスペースは、この領域に割り当てられます。パネル内の 1 つの領域のみ、'*' でコード化できます。

オプションの属性

BOTSEP=SPACE | NONE | RULE

メニュー域の最下部区切り記号を定義します。SPACE が指定されている場合、スペースから成る 1 行が使用されます。

NONE は、区切り行が存在しないことを示します。

RULE が指定されている場合、下線付きスペースから成る 1 行が区切り行として使用されます。

SCROLL=NO | YES

NO は、メニューがスクロール可能でないことを示します。

YES は、メニュー域がスクロール可能であることが意図されていることを示します。通常、YES は使用すべきではありませんが、スクロール可能な長いメニューは SCROLL=YES を指定して構成することが可能です。SCROLL=YES の領域の場合、スクロール情報の行を提供するために、UIM によって 1 行のスペースが使用されます。また、BOTSEP=SPACE もこのタグに指定されている場合、この領域に最下部指示も含まれていない限り、1 行のスペースだけが使用されます。

オプションのテキスト

area-title

領域のタイトル。テキストが指定されていない場合、領域にタイトル行は割り振られません。テキストはタグと同じ行または次の行に表示される必要があります。また、テキストにはテキスト反転 (RT) タグのみを含めることができます。最大長の 55 文字を超えることはできません。

例 1: 単純なメニュー域

この例は、3 つのオプションを持つメニュー域を定義します。

UIM ソース

```
:panel name=main
      help='menu/main'
      keyl=small
      panelid=zmenu
```

```

        topsep=space
        .My Main Menu
:menu depth='*'.
:topinst.Select one of the following:
:menui help=MNUSER
        option=1
        action='MENU X'
        .Display menu X
:menui help=MNSYS
        option=2
        action='MENU SYSOPR'
        .System operations
:menui help=MNOFF
        option=90
        action='CMD SIGNOFF'
        .Sign off
:emenu.
:cmdline size=long.Selection or command
:epanel.

```

結果

```

MAIN                               My Main Menu

Select one of the following:

    1. Display menu X
    2. System operations

    90. Sign off

Selection or command
===> _____

F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel

```

例 2: グループを持つメニュー域

この例には 3 つのメニュー域があります。それぞれには、選択可能なオプションがあります。

UIM ソース

```

:panel name=cfg2
        help='menu/cfg2'
        panelid=zmenu
        keyl=small
        .Configuration Templates
:menu depth='*'.
:topinst.Select one of the following:
:menugrp.Local Hardware:
:menui help='opt/lcl/wsprt'
        option=1
        action='MENU LCLWSPRT'
        .Work stations and printers
:menui help='opt/lcl/ctl'
        option=2
        action='MENU LCLCTL '

```

MENU タグ

```
.Controllers
:menui help='opt/lcl/tapdkt'
      option=3
      action='MENU LCLTAPDKT'
      .Tape drives and diskette drives
:emenugrp.
:menugrp.Remote Hardware:
:menui help='opt/rmt/wsprt'
      option=4
      action='MENU RMTWSPRT'
      .Work stations and printers
:menui help='opt/rmt/ctl'
      option=5
      action='MENU RMTCTL '
      .Controllers
:menui help='opt/rmt/tapdkt'
      option=6
      action='MENU RMTTAPDKT'
      .Tape drives and diskette drives
:emenugrp.
:menugrp.Communications Support:
:menui help='opt/cmn/lines'
      option=7
      action='MENU CMNLINES'
      .Communications lines
:menui help='opt/cmn/ctl'
      option=8
      action='MENU CMNCTL '
      .Communications controllers
:emenugrp.
:emenu.
:cmdline size=long.Selection or command
:epanel.
```

結果

```
CFG2                Configuration Templates                System:  xxxxxxxx
Select one of the following:

Local Hardware:
  1. Work stations and printers
  2. Controllers
  3. Tape drives and diskette drives

Remote Hardware:
  4. Work stations and printers
  5. Controllers
  6. Tape drives and diskette drives

Communications Support:
  7. Communications lines
  8. Communications controllers

Selection or command
===> _____
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel
```


MENUGRP (メニュー・グループ)

```
▶▶ :MENUGRP [COND=condition-name] . [group=heading] :EMENUGRP. ▶▶
```

メニュー・グループ (MENUGRP) タグは、メニュー域のメニュー項目をまとめてグループ化し、そのグループの見出しを提供します。このタグは、表示パネルでのみ許可されます。

メニュー・グループの最初の 2 項目を表示するのに必要な行数が、現行ページに残っている行数を超える場合、グループ全体が強制的に次のページに送られます。メニュー・グループ内のすべての項目が 1 ページに完全に表示されないときには、すべての項目を表示するのに必要な数のページまでグループは続けられます。

他のタグを MENUGRP タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 72. MENUGRP タグと EMENUGRP タグの間で許可されるタグ

タグ名	順序	使用法	ページ
MENUI (メニュー項目)	1	D	
	642		

オプションの属性

COND=condition-name

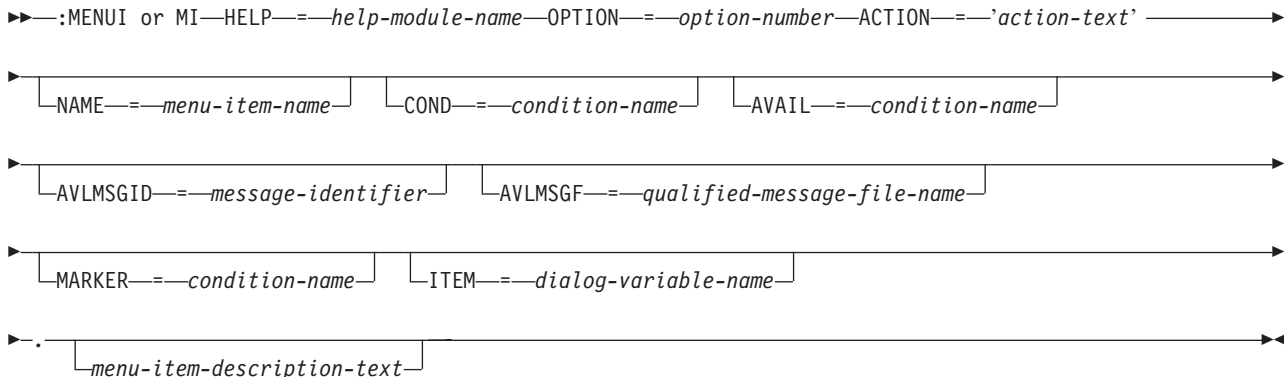
指定された条件が真の場合にのみ、グループが表示され、パネル上でアクティブになります。条件が偽の場合、グループ全体が表示されません。条件は、パネル・グループ・プロローグ内で条件定義 (COND) タグを使用して定義する必要があります。

オプションのテキスト

group-heading

グループのメニュー項目の上にある見出し。テキストはタグと同じ行または次の行に表示される必要があります。また、テキスト反転 (RT) タグのみを含めることができます。グループ見出しは常に列 4 で始まる表示行に表示されます。

MENUI または MI (メニュー項目)



必須属性

HELP=help-module-name

メニュー項目の目的を説明するオンライン情報を識別します。ヘルプ・モジュールの名前は別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説された名前の規則に従っていなければなりません。名前の規則について詳しくは、515 ページの『名前構文』を参照してください。

カーソルがこのタグのテキスト上にある場合、またはカーソルがコマンド行かオプション行上にあり、有効なメニュー項目番号が入力されている場合に、ヘルプが要求されると、この属性によって識別されるオンライン情報が表示されます。

OPTION=option-number

このオプションに割り当てる番号。オプション番号は 0 から 999 の範囲の整数です。

メニュー項目は、メニュー域で定義した順序で表示されます。2 つの項目の番号付けが連続していない場合は、2 つの項目の間に空白行が自動的に挿入されます。

ACTION='action-text'

メニュー項目がユーザーによって選択されるときに取られるアクション。action-text の有効な形式を以下に示します。

- 'CALL program-reference'
- 'CANCEL'
- 'CMD command-string' コマンド・ストリング内のダイアログ変数名は、変数置換を示すために、前にアンパーサンドが付いており、ピリオドで終わらなければなりません。
- 'EXIT'
- 'MENU qualified-menu-name RTNPNT\NORTNPNT'
- 'RETURN positive-integer'

これらの各アクションについては、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

オプションの属性

NAME=*menu-item-name*

メニュー項目に関連した名前。この名前は、このメニュー項目の近くにウィンドウを配置するために、Add Pop-Up Window (QUIADDPW) API とともに使用することができます。命名規則については、515 ページの『名前構文』を参照してください。

COND=*condition-name*

指定された条件が真の場合にのみ、メニュー項目がパネル上で有効になります。条件は、パネル・グループ・プロローグ内で条件定義 (COND) タグを使用して定義する必要があります。

AVAIL=*condition-name*

メニュー項目をユーザーが選択可能であるかどうかを UIM に示す条件の名前。条件は、パネル・グループ・プロローグ内で COND タグを使用して定義する必要があります。

使用不可のメニュー項目は、メニュー項目番号の最初の部分がアスタリスク (*) でオーバーレイされた状態で表示されます。

条件が真の場合、メニュー項目は使用可能です。条件が偽の場合、メニュー項目は使用不可です。このタグの COND 属性で指定されている条件があれば、その条件がこの属性より優先して使用されます。

AVLMSGID=*message-identifier*

プルダウン選択項目が選択されたが、このタグの AVAIL 属性での指定によってその項目が使用可能でない場合に表示されるメッセージのメッセージ ID。この属性を指定できるのは、AVAIL 属性が指定されている場合だけです。

この属性が指定されない場合、UIM は選択項目が現在使用可能でないことを告げるデフォルト・メッセージを表示します。

AVLMSGF=*'qualified-message-file-name'*

メッセージ ID が入っているメッセージ・ファイル名。この属性が許可されるのは、このタグで AVLMSGID 属性が指定されている場合だけです。パネル・グループ (PNLGRP) タグに DFTMSGF 属性が指定されていない場合で、このタグに AVLMSGID 属性が指定されている場合は、この属性を指定する必要があります。

MARKER=*condition-name*

メニュー項目がより大符号 (>) でマークされるかどうかを示す条件の名前。指定された条件が真の場合にのみ、マーカーが表示されます。条件は、パネル・グループ・プロローグ内で COND タグを使用して定義する必要があります。

条件が真の場合、メニュー項目のオプション番号の左に、より大符号が表示されます。条件が偽の場合、メニュー項目のオプション番号の前は、行がブランクになります。この属性が省略される場合、マーカーはメニュー項目に表示されません。

ITEM=*dialog-variable-name*

パネルが表示されるときにメニュー項目記述のテキスト含むダイアログ変数の名前。ダイアログ変数は、クラス定義 (CLASS) タグで BASETYPE に CHAR、IGC、または BIN を指定して定義する必要があります。ダイアログ変数の宣言された長さは、列 10 で始まる、パネルの 1 行に収まる長さを超えてはなりません。アプリケーション・フォーマット域がある場合、それが許可されなければなりません。

この属性が使用される場合、このタグのピリオドの後に *menu-item-description-text* は指定できません。

オプションのテキスト

menu-item-description-text

オプションに伴って、メニュー上に表示される記述テキスト。テキストはソース内の複数行に表示されることがあります。テキストにはテキスト反転 (RT) タグのみを含めることができます。長さが 70 文字を超えることはできません。

項目記述テキストはパネルの列 10 で始まります。オプション番号、ピリオド、および 1 個のブランクが項目記述テキストの直前に表示されます。項目番号は、オプション番号が 1、2、3 桁のどれかに応じて、列 7、6、5 で始まります。テキストは必要に応じて追加行にフォーマットされ、その追加行はパネルに合うように 2 スペース字下げされることはありません。

NT または NOTE (注記)

▶▶:NT. or :NOTE. note-text :ENT. or :ENOTE.◀◀

注記 (NT) タグは、単一または複数の段落の注記を識別します。このタグには、対応する終了タグが必要です。:NOTE および :ENOTE を :NT および :ENT タグの代わりに使用することもできます。これらのタグは情報域とヘルプ領域でのみ許可されます。注記はネストできません。

注記は、段落が許可されているテキストの任意の場所に出現できます。注記には、基本のテキスト項目を含めることができますが、見出し、図表、線、例、またはどのタイプのリストも含めることはできません。見出しタグが見つかったときにこのタグが終了していない場合、注記は終了し、警告メッセージがコンパイラによって出されます。

注記は暗黙の段落です。これはブロックとしてフォーマットされ、現行マージンから 6 スペース字下げされます。Note という語は強調表示句 2 (HP2) でフォーマットされ、コロンが後に続いており、これが段落の始まりです。注記タグかテキストのどちらかが 1 バイト文字である場合は、注記タグとテキストの間に 2 スペースがあります。注記タグとテキストの両方が 2 バイト文字である場合には、両者の間には 4 スペースがあります。

注記がリスト内で使用されるときには、リスト項目のテキストと位置合わせされます。

他のタグを NOTE タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 73. NOTE タグと ENOTE タグの間で許可されるタグ

タグ名	順序	使用法	ページ
P (段落)	1	B	648
PC (段落の連結)	1	B	658

オプションのテキスト

note-text

注記のテキストは必須ではありません。ただし、タグはテキストが指定されていないときには意味を持ちません。

例: 注記の使用

注記タグは、この例にあるように、段落内の重要な情報を目立たせます。

UIM ソース

```
:P.Here's a paragraph.
:NT.
Here's the first paragraph
of the note.
:P.
Here's the second paragraph
of the note.
:ENT.
:P.Here's a paragraph.
```

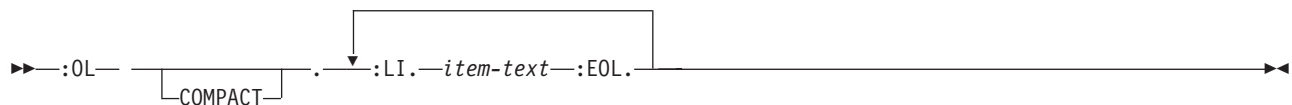
結果

Here's a paragraph. Note: Here's the first paragraph of the note.

Here's the second paragraph of the note.

Here's a paragraph

OL (番号付きリスト)



番号付きリスト (OL) タグは、項目の番号付きリストを識別します。これには、対応する終了タグが必要です。これらのタグは、情報域とヘルプ領域でのみ許可されます。

番号付きリストはテキスト内の任意の場所に出現でき、他のリスト内でネストできます。

注: 順序リスト内で不定形式行 (LINES)、図表 (FIG)、および例 (XMP) タグを使用するときには注意が必要です。なぜなら、1 行に収まらないテキストは次の行の列 1 に折り返されるからです。行および図表は現行左マージンから始まり、例は現行左マージンから 4 スペース字下げされます。ネストされたリストがフォーマットされると、現行左マージンは変わります。オンライン・ヘルプ情報のさまざまな場所 (リスト内を含む) に組み込まれた LINES、FIG、または XMP タグは、毎回同じように見えるわけではありません。

OL タグは、ぶら下げインデント・リストとしてフォーマットされ、左マージンから見て 4 桁目に項目 ID (1,2,...a,b,...)があります。テキストは、左マージンから見て 8 桁目で始まります。

順序リストは、各レベルの定義に応じて、定様式文書では順序番号または文字によって識別されます。定義のレベルは以下のとおりです。

1. 1., 2., 3., . . . ,11., 12., . . . , 99.

OL、LI タグ

2. a., b., c., . . . , z., aa., bb., . . . , zz.
3. 1), 2), 3), . . . ,11), 12), . . . , 99)
4. a), b), c), . . . , z), aa), bb), . . . , zz)

追加レベルは、先頭から順序を繰り返します。

他のタグを OL タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 74. OL タグと EOL タグの間で許可されるタグ

タグ名	順序	使用法	ページ
FIG (図表)	1	B	577
LINES (不定形式行)	1	B	597
XMP (例)	1	B	697
NT (注記)	1	B	644
P (段落)	1	B	648
PC (段落の連結)	1	B	658
LP (リストの一部)	1	B	633
OL (番号付きリスト)	1	B	645
SL (単純リスト)	1	B	678
UL (順不同リスト)	1	B	690
PARML (パラメーター・リスト)	1	B	656
DL (定義リスト)	1	B	575

オプションの属性

COMPACT

この属性は、項目間にブランク行を入れずにリストをフォーマットします。

必須タグ

:LI.item-text

リスト項目のテキスト。数値または文字およびピリオドがテキストの前に付きます。

例: 番号付きリスト

この例には、2 つの番号付きリストがあり、一方が他方の中に組み込まれています。2 番目の番号付きリストは COMPACT 属性を使用します。

UIM ソース

```
Some normal text...
:o1.
:li.First item (number)
:o1.
:li.First item (letter)
:li.Second item (letter)
:eol.
:li.Second item (number)
:eol.
```

結果

```
Some normal text...

1. First item (number)
    a. First item (letter)
    b. Second item (letter)

2. Second item (number)
```

OPTLINE (オプション行)

```
▶▶:OPTLINE [NAME=option-line-name] . [instruction-text] ▶▶
```

オプション行 (OPTLINE) タグは、コマンドまたはオプションを入力できるコマンド行とは対照的に、メニュー・オプション・フィールドを持つパネルを指定します。このタグは、表示パネルでのみ許可されます。OPTLINE タグは、メニュー域を持つパネルでのみ使用できます。

このタグに関連したテキストは指示行として使用され、パネル上のオプション・フィールドのすぐ上に表示されます。

このタグは、パネル内の最終タグでなければならず、EPANEL タグの直前に置く必要があります。コマンド行 (CMDLINE) タグと OPTLINE タグは相互に排他的です。

オプションの属性

NAME=*option-line-name*

オプション行に関連した名前。この名前は、オプション行の近くにウィンドウを配置するために、Add Pop-Up Window (QUIADDPW) API とともに使用することができます。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

オプションのテキスト

instruction-text

オプション行の指示として表示されるテキスト。このテキストは暗黙の段落です。

画面のフォーマット時に、1 表示行に収まらないテキストは、2 行目以降のブランク行に 2 桁字下げしてフォーマットされます。テキストは最大 255 文字で、テキスト反転 (RT) タグだけを含めることができます。テキストが提供されない場合、指示行は割り振られることも、オプション行の上に表示されることもありません。

P タグ

P (段落)

▶▶:P.—paragraph-text—◀◀

段落 (P) タグは、主題によって関連付けられた 1 つ以上の文である段落を識別します。このタグは情報域とヘルプ領域でのみ許可されます。

段落はテキスト内の任意の場所に出現でき、テキスト項目を含むことができます。対応する終了タグは許可されません。段落は別の段落またはさらに高いレベルの要素によって終了されます。

各段落は、最初の行を字下げしないテキストのブロックとしてフォーマットされます。段落は 1 つのブランク行によって別のテキスト項目から分離されます。

リスト内に挿入される段落は、リスト項目のテキストと位置合わせされます。

オプションのテキスト

段落のテキストは必須ではありません。ただし、タグはテキストが指定されていないときには意味を持ちません。

例: 段落タグ

この例は、段落タグを使用する方法を示します。

UIM ソース

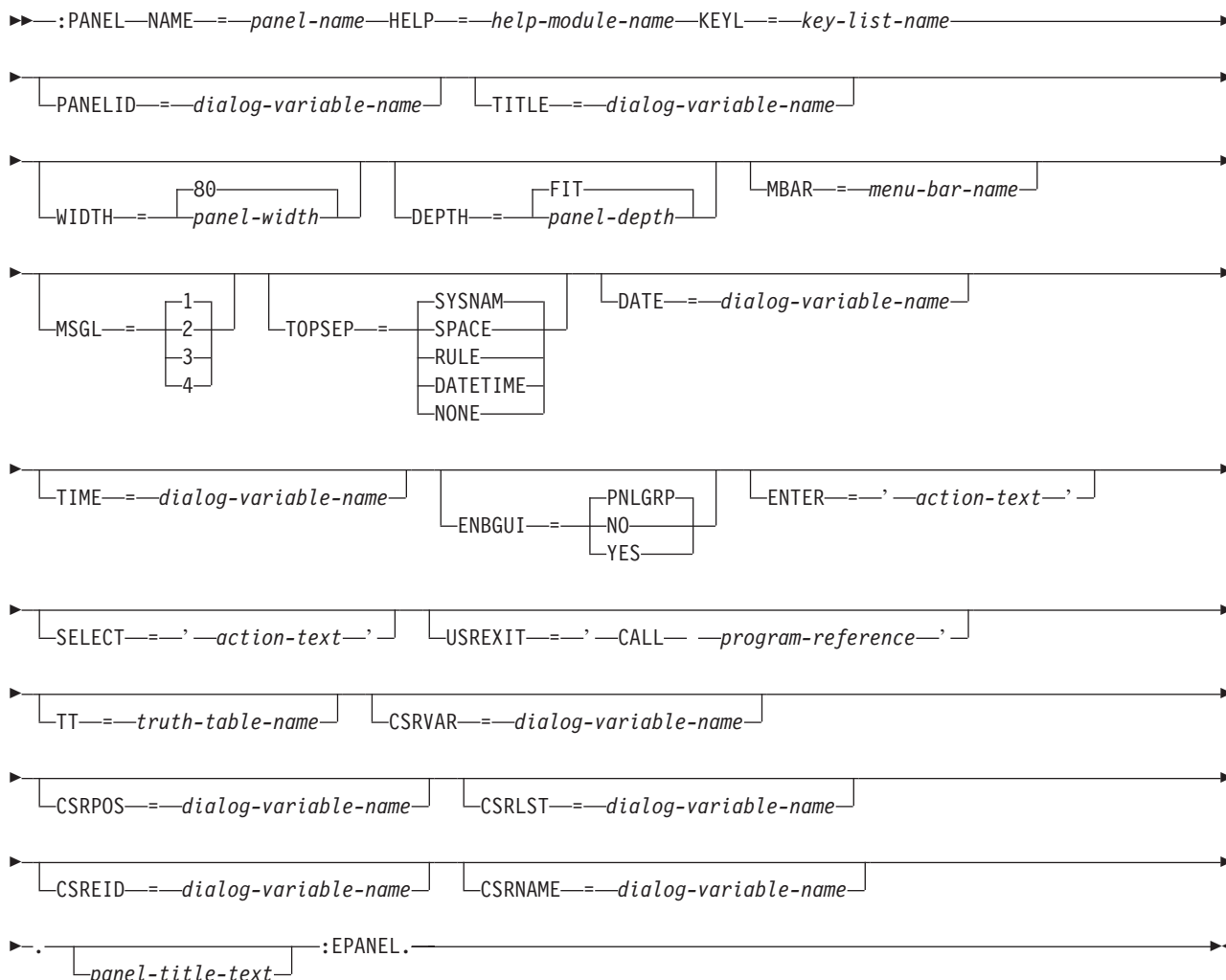
```
:P.Here's a paragraph.  
Lines are formatted to  
fill the column.  
:P.Here's another  
paragraph.
```

結果

Here's a paragraph. Lines are formatted to fill the column.

Here's another paragraph.

PANEL (表示パネル)



表示パネル (PANEL) タグは PANEL タグを使用して開かれ、EPANEL タグを使用して閉じられます。これには、1 つ以上のパネル域定義を定義するタグが含まれます。

他のタグを PANEL タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 75. PANEL タグと EPANEL タグの間で許可されるタグ

タグ名	順序	使用法	ページ
DATA (データ表示域)	1	D	544
INFO (情報域)	1	D	587

PANEL タグ

表 75. PANEL タグと EPANEL タグの間で許可されるタグ (続き)

タグ名	順序	使用法	ページ
LIST (リスト域)	1	D	603
MENU (メニュー域)	1	D	637
TEXT (テキスト域)	1	D	680
CMDLINE (コマンド行)	2	D	539
OPTLINE (オプション行)	2	D	647

必須属性

NAME=*panel-name*

パネルの名前。命名規則については、515 ページの『名前構文』を参照してください。

HELP=*help-module-name*

パネルに関する全般ヘルプの先頭を含むヘルプ・モジュールの名前。

KEYL=*key-list-name*

パネルのキー・リストの名前。

オプションの属性

PANELID=*dialog-variable-name*

パネルが表示されるときにパネル ID のテキストを含むダイアログ変数の名前。ダイアログ変数は、クラス定義 (CLASS) タグで `BASETYPE='CHAR 10'`、`'NAME 10'`、または `'OBJNAME 10'` を指定して定義しなければなりません。UIM は、変数の内容をパネルの左上隅に表示します。この属性が省略される場合、パネル ID は画面に表示されません。

メニュー作成 (CRTMNU) コマンドを使用して作成されたメニューのパネル ID を指定するには、ZMENU のダイアログ変数が PANEL タグで `PANELID=ZMENU` を指定して宣言しなければなりません。ZMENU ダイアログ変数の値がパネル ID として表示されます。

TITLE=*dialog-variable-name*

パネル・タイトルのテキストを含むダイアログ変数の名前。ダイアログ変数は、表示幅 55 か、パネルの幅から 2 を引いた値のうち、小さい方の値で定義しなければなりません。変換リストを使用して、パネル・タイトルを提供することが可能です。

UIM はストリングから末尾ブランクを除去し、作成されたテキストを画面のパネル・タイトル域で中央揃えします。この属性が使用される場合、*panel-title-text* は指定できません。

ポップアップ・パネルの場合、ダイアログ変数の宣言される長さは、ポップアップ・パネルの幅から 2 バイトを引いた値に等しいか、あるいはパネルのこのタグに `PANELID` または `TOPSEP=DATETIME` が指定されている場合には、その値より小さくなければなりません。

WIDTH=80 | *panel-width*

パネルの幅。幅が指定されない場合、パネル幅はデフォルトの 80 バイトに設定されます。指定される幅には、左右のマージン用に 2 バイトのスペースが含まれていなければなりません。パネル幅が `WIDTH=80` の場合、パネル・データ用のスペースは 78 バイトになります。

フルスクリーンで表示されるパネルにサポートされる最大幅は 132 バイトです。ポップアップ・ウィンドウにサポートされる最大幅は、デバイスがサポートしている画面のサイズによって異なります。24 行 x 80 バイトをサポートするデバイスでは、74 バイトがポップアップ・ウィンドウの最大幅で

す。27 行 x 132 バイトをサポートするデバイスでは、126 バイトがポップアップ・ウィンドウの最大幅です。ポップアップ・ウィンドウの最小幅は 20 バイトです。

DEPTH=FIT | *panel-depth*

パネルの深さ。深さが指定されない場合、パネルの深さはデフォルトの FIT に設定されます。

DEPTH=FIT の場合、UIM は表示時にデバイスの深さを使用して、パネルの深さを判別します。

DEPTH=FIT のとき、コンパイラーはすべてのパネル・エレメントが 24 行に収まることを確認します。

指定される深さには、以下が含まれます。

- パネル・タイトル用に 1 行。
- 領域データ用に 1 行。
- 機能キーの記述用に 1 または 2 行。
- メッセージ行用に 1 行。

全画面で表示されるパネルの場合、サポートされる最大の深さは 27 行です。ポップアップ・パネルにサポートされる最大の深さは、ディスプレイ装置がサポートできる内容によって異なります。24 行 x 80 バイトだけをサポートできるデバイスでは、ポップアップ・ウィンドウの最大の深さは 21 行です。27 行 x 132 バイトをサポートできるデバイスでは、ポップアップ・ウィンドウの最大の深さは 24 行です。

許可される最小の深さは 5 行です。

MBAR=menu-bar-name

パネルで使用されるメニュー・バーの名前。この属性は、確認パネル、またはポップアップ・ウィンドウとして使用されるパネルでは許可されません。

メニュー・バーは、以下のいずれかのサイズで定義されたパネルでのみ許可されます。

- WIDTH=80 または 132 バイト、および DEPTH=FIT
- WIDTH=80 バイト、DEPTH=24 行
- WIDTH=132 バイト、DEPTH=27 行

MSG1=1 | 2 | 3 | 4

このパネルに表示されるメッセージ行の数。1 メッセージ行がデフォルトです。

TOPSEP=SYSNAM | SPACE | RULE | DATETIME | NONE

パネルの上部区切り記号を定義します。SYSNAM が指定される場合、区切り行にはシステム名のみ含まれます。

SPACE が指定されている場合、スペースから成る 1 行が使用されます。

RULE が指定されている場合、下線付きスペースから成る 1 行が使用されます。このタグに MBAR 属性が指定されている場合、RULE を指定することはできません。

DATETIME が指定されている場合、区切り行には、このタグの DATE および TIME 属性で識別されたダイアログ変数からの日時が含まれます。また、DATETIME を指定すると、システム名がタイトル行の右側に表示されます。日時がパネルの上部区切り記号として使用されるとき、現在時刻クロックを提供するためにそれを使用してはなりません。この関数は、パネル上の重要なデータが収集または生成される日時を示すために使用しなければなりません。

NONE は、区切り行が存在しないことを示します。

TOPSEP=SPACE または NONE 以外の値が指定されていると、パネル幅が 80 より小さいときに、予期しない結果になることがあります。

PANEL タグ

DATE=*dialog-variable-name*

パネルの上部区切り記号に表示された日付を含むダイアログ変数の名前。この属性は、TOPSEP=DATETIME がこのタグに指定されているときには必須です。ダイアログ変数は、CLASS タグで BASETYPE='DATE' を指定して定義する必要があります。

TIME=*dialog-variable-name*

パネルの上部区切り記号に表示された時刻を含むダイアログ変数の名前。この属性は、TOPSEP=DATETIME が指定されているときには必須です。ダイアログ変数は、CLASS タグで BASETYPE='TIME' を指定して定義する必要があります。

ダイアログ変数がそれに関連した時間帯値を持つことができる場合、時間値は右側のブランクが切り取られ、画面上に右揃えされます。ダイアログ変数の時間帯の値がブランクの場合、時間帯の値は表示されません。

ENBGUI=PNLGRP | NO | YES

クライアント・プログラムにより、パネルでグラフィカル・ユーザー・インターフェース (GUI) への変換が可能かどうかを指定します。

ENBGUI=YES が指定された場合、UIMには 5250 データ・ストリーム内のパネルのレイアウトおよび内容についての情報が組み込まれます。クライアント・プログラムはこの情報を使用して、クライアント上にグラフィカル・インターフェースを作成します。表 76 では、5250 データ・ストリームに組み込まれる情報を説明しています。

表 76. UIM の指紋のレイアウト

バイト	値および説明
0	16 進 27 - 非表示属性
1	16 進 40 - ブランク
2	16 進 20 - 終了属性
3	予約済み
4	16 進 00 - Panel BIDI=NONE または LTR 16 進 40 - Panel BIDI=RTL
5	16 進 00 - パネルにはリスト域が含まれない。 16 進 40 - パネルにはリスト域が含まれる可能性がある。
6	16 進 00 - パネルにはメニュー域が含まれない。 16 進 40 - パネルにはメニュー域が含まれる可能性がある。
7	16 進 00 - パネルにはヘルプ域が含まれない。 16 進 40 - パネルにはヘルプ域が含まれる可能性がある。 これには、:TEXT 領域を含むアプリケーション・パネルだけでなく、UIM ヘルプ (UH) パネルも含まれます。
8	16 進 00 - パネルにはその他の領域は含まれない。 16 進 40 - パネルには 1 つ以上のその他の領域が含まれる可能性がある。 領域のその他のタイプ :DATA (入力または出力あるいはその両方)、:INFO (ヘルプに類似)。
9	16 進 27 - 非表示属性
10	16 進 40 - ブランク
11	16 進 20 - 終了属性

ENBGUI=NO が指定されている場合、UIM はデータ・ストリームに追加の情報を含めません。

ENTER='action-text'

パネルのデフォルトの入力アクションで、Enter キーが押され、UIM 定義のアクション (メニューまたは処置リストの処理など) が必要でない場合に取られます。この属性が省略される場合、パネルはメッセージまたはエラーの通知を含めずにユーザーに再表示されます。アクション・テキストの有効な形式を以下に示します。

- '*CALL program-reference*' UIM とファンクション・キー呼び出し用プログラム間のインターフェースの説明については、iSeries Information Center の『APIs』のトピックを参照してください。
- '*CMD command-string*' コマンド・ストリング内のダイアログ変数の前には、変数置換を示すアンパサンドが付いていなければなりません。
- '*MSG message-identifier [qualified-message-file-name]*'
- '*RETURN positive-integer*'

これらのダイアログ・コマンドの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

SELECT='action-text'

パネルのデフォルトの選択アクションで、項目が選択リストまたは処置リストで選択され、アクションがメニュー・バーから選択されないときに取られます。この属性が指定されない場合、パネルはメッセージまたはエラーの通知なしでユーザーに表示されます。アクション・テキストの有効な形式を以下に示します。

- '*ACTIONS*'
- '*MSG message-identifier [qualified-message-file-name]*'
- '*PULLDOWN*'
- '*RETURN positive-integer*'

このタグに MBAR 属性が指定されていない場合、PULLDOWN および ACTIONS ダイアログ・コマンドを指定することはできません。

USREXIT='CALL program-reference'

ダイアログ変数およびリスト項目が更新された後、およびキー・リスト項目 (KEYI)、リスト・アクション (LISTACT)、プルダウン・フィールド選択項目 (PDFLDC)、またはメニュー項目 (MENUUI) タグに指定されたアクションが実行される前に、ユーザーがファンクション・キーまたは Enter キーを押すたびに呼び出される汎用出口プログラム。

プログラムには、ユーザーによって要求される関数を含むパラメーターが渡されます。出口プログラムは、アプリケーションで定義された妥当性検査を行うか、画面上に表示されるリストの表示位置属性を変更するか、またはその両方を行います。出口プログラムが CPF6A02 状況メッセージを UIM に送信する場合、UIM は通常のアクション処理を実行しないでパネルを再表示します。

CLASS、妥当性検査 (CHECK)、または変換リスト (TL) タグに関連した妥当性検査エラーが検出される場合、またはユーザーがアクティブでないファンクション・キーを押す場合、パネルはこの出口プログラムを呼び出さずに再表示されます。KEYI タグを VARUPD=NO にして定義された機能キーをユーザーが押す場合、ユーザーがパネルに入力した値はこのプログラムには使用できません。それらは、UIM に内部的に保管されるだけです。

CALL ダイアログ・コマンドの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

UIM と汎用出口プログラム間のインターフェースの説明については、iSeries Information Center の『APIs』のトピックを参照してください。

PANEL タグ

TT=*truth-table-name*

真理値表 (TT) タグによって定義された真理値表の名前。アプリケーションの実行中に条件定義 (COND) タグによって定義された条件について、どの真理値の組み合わせが発生するかを指定します。

指定される表には、パネル定義のタグによって参照される条件のサブセットまたはスーパーセットが含まれることがあります。表に指定されている真理値の組み合わせだけが、表に指定されていない真理値について最悪の事態を想定することによって増やされて、パネル定義が使用可能かどうかを評価する上で考慮に入られます。パネルが表示されるときに発生する可能性のある真理値の組み合わせがどのようなものであっても、表はそれらを除外すべきではありません。有効な真理値の組み合わせが省略される場合、エラーが出されずにパネル・グループ・オブジェクトが作成され、パネルが表示されるときに望ましくない結果になることがあります。たとえば、UIM は、表示についての条件が設定されているパネル・エレメントを表示できない場合があります。真理値表について詳しくは、688 ページの『TT (真理値表)』を参照してください。

この属性が省略される場合、真理値のすべての組み合わせが可能であると見なされます。これにより、タグ言語コンパイラは、パネル定義が有効かどうかを評価する際に、すべての条件について最悪の事態を想定することになります。

CSRVAR=*dialog-variable-name*

この属性は、CHAR 10 ダイアログ変数を指定する必要があります。これは、ダイアログ変数の名前か、カーソルが現在置かれているかまたは置かれる予定のコマンド行の名前を含めるために UIM によって使用されるものです。

UIM は、パネルが表示された後および汎用出口プログラムまたはアクション・ルーチンが呼び出される前に、この変数を更新します。これには、フィールドに関連したダイアログ変数の名前、またはカーソルがパネルからの出口に置かれているコマンド行の名前 (指定されている場合) が含まれます。カーソルがフィールドまたは指定されたコマンド行にない場合は、CSRVAR ダイアログ変数がブランクに設定されます。

Display Panel (QUIDSPP) API は、パネルが表示される前に CSRVAR、CSRPOS、CSRLST、および CSREID パネル属性に関連したダイアログ変数を設定することにより、アプリケーション・プログラムが最初のカーソル位置を制御できるようにするオプションをサポートします。CSRVAR ダイアログ変数に、パネルに表示されるダイアログ変数またはコマンド行の名前が含まれていない場合、UIM によるデフォルト・カーソルの位置決めが使用され、エラーは報告されません。

データ項目がデータ項目エクステンダーを使用して定義される場合、カーソルがエクステンダー・フィールドに置かれていれば、CSRVAR 属性にはエクステンダー・ダイアログ変数の名前が含まれます。カーソルがフィールドの間に置かれている場合、CSRVAR はデータ表示域 (DATA) タグの VAR 属性に指定されたダイアログ変数の名前に設定され、CSRPOS ダイアログ変数はゼロに設定されます。

CSRPOS=*dialog-variable-name*

この属性は、BIN 15 ダイアログ変数を指定する必要があります。これは、このタグの CSRVAR 属性によって識別される、カーソルが現在置かれているかまたは置かれる予定のフィールドに文字位置を含めるために UIM によって使用されるものです。

UIM は、パネルが表示された後、パネル出口プログラムまたはアクション・ルーチンが呼び出される前に、この変数を更新します。これには、カーソルがパネルからの出口に置かれているフィールド内の位置が含まれます。カーソルがフィールドにない場合、ダイアログ変数はゼロに設定されます。カーソルがフィールドに関連したテキスト域 (リスト域のプロンプト・ラベルまたは列見出しなど) に置かれている場合、CSRVAR 属性には対応するデータ項目またはリスト列のダイアログ変数の名前が含まれ、CSRPOS ダイアログ変数はゼロに設定されます。

QUIDSPP API は、パネルが表示される前にこのタグの CSRVAR、CSRPOS、CSRLST、および CSREID 属性に関連したダイアログ変数を設定することにより、使用中のプログラムが最初のカーソル位置を制御

できるようにするオプションをサポートします。CSRPOS 属性が表示フィールドに有効でない位置を識別する場合、UIM によるデフォルト・カーソルの位置決めが使用され、エラーは報告されません。

CSRLST=dialog-variable-name

この属性は、CHAR 10 ダイアログ変数を指定する必要があります。これは、カーソルが現在置かれているかまたは置かれる予定のリストの名前を含めるために UIM によって使用されます。

UIM は、パネルが表示された後、パネル出口プログラムまたはアクション・ルーチンが呼び出される前に、この変数を更新します。これには、カーソルがパネルからの出口に置かれているリスト域に関連したリストの名前が含まれます。カーソルがリスト域にない場合、ダイアログ変数はブランクに設定されます。

QUIDSPP API は、パネルが表示される前にこのタグの CSRVAR、CSRPOS、CSRLST、および CSREID 属性に関連したダイアログ変数を設定することにより、使用中のプログラムが最初のカーソル位置を制御できるようにするオプションをサポートします。CSRLST 属性がブランクではなく、パネルに表示されるリストの名前が含まれていない場合、UIM によるデフォルト・カーソルの位置決めが使用され、エラーは報告されません。

CSREID=dialog-variable-name

この属性は、CHAR 4 ダイアログ変数を指定する必要があります。これは、CSRLST 属性によって識別される、カーソルが現在置かれているかまたは置かれる予定のリストにリスト項目ハンドルを含めるために UIM によって使用されるものです。

UIM は、パネルが表示された後、パネル出口プログラムまたはアクション・ルーチンが呼び出される前に、この変数を更新します。これには、カーソルがパネルからの出口に置かれていたリスト項目 ID が含まれます。カーソルがリスト項目にない場合、CSREID ダイアログ変数は X'00' に設定されます。カーソルがリスト列に関連したテキスト域 (列見出しなど) に置かれていた場合、CSRLST 属性には対応するリストの名前が含まれ、CSREID ダイアログ変数は X'00' に設定されます。

QUIDSPP API は、パネルが表示される前にこのタグの CSRVAR、CSRPOS、CSRLST、および CSREID 属性に関連したダイアログ変数を設定することにより、使用中のプログラムが最初のカーソル位置を制御できるようにするオプションをサポートします。CSREID 属性が、パネルに表示されたリスト項目を指定しない使用不可の位置を識別する場合、UIM によるデフォルト・カーソルの位置決めが使用され、エラーは報告されません。

CSRNAME=dialog-variable-name

この属性は、CHAR 10 ダイアログ変数を指定する必要があります。これは、カーソルが置かれている項目の名前を含めるために UIM によって使用されます。

UIM は、パネルが表示された後、パネル出口プログラムまたはアクション・ルーチンが呼び出される前に、この変数を更新します。これには、カーソルがパネルからの出口に置かれている項目の名前が含まれます。カーソルが項目にない場合、CSRNAME ダイアログ変数はブランクに設定されます。

この変数の値は、Add Pop-Up Window (QUIADDPW) API を使用して、ポップアップ・ウィンドウをフィールドに隣接して配置するために使用されます。

オプションのテキスト

panel-title-text

このパネルで使用されるタイトル。テキストは、タグと同じ行または次の行に表示される必要があります。テキストにその他のタグを含めることはできません。また、パネルの幅に基づいて長さが制限されます。*panel-title-text* は長さが 55 バイトを超えないで、TITLE 属性がこのタグに指定されていなければ必須です。

PANEL タグ

例: パネル定義

この例は、パネルを定義し、そのパネル内で使用されるエレメントをセットアップします。

UIM ソース

```
:panel name=main
      help=hmain
      panelid=zmenu
      key1=small
      .My Main Menu
:menu depth='*'.
:topinst.Select one of the following:
:menui help=MNUSER option=1
      action='CMD MENU X'
      .Display menu X
:menui help=MNSYS
      option=2
      action='CMD MENU SYSOPR'
      .System operations
:menui help=MNOFF
      option=90
      action='CMD SIGNOFF'
      .Sign off
:emenu.
:cmdline size=long.Selection or command
:epanel.
```

結果

MAIN My Main Menu System: SYSTEM01

Select one of the following:

- 1. Display menu X
- 2. System operations
- 90. Sign off

Selection or command
===> _____

F3=Exit F4=Prompt F9=Retrieve F12=Cancel

PARML (パラメーター・リスト)

▶▶:PARML. —:PT.—parameter-term—:PD.—description—:EPARML. —————▶▶

パラメーター・リスト (PARML) タグは、パラメーターの用語と記述を識別します。これらのタグは、情報域とヘルプ領域でのみ許可されます。このタグには、対応する終了タグが必要です。定義する用語とその定義は、それぞれパラメーター用語 (PT) タグとパラメーター定義 (PD) タグによって識別されます。

注: パラメーター・リスト内で不定形式行 (LINES)、図表 (FIG)、および例 (XMP) タグを使用するときには注意が必要です。なぜなら、1 行に収まらないテキストは次の行の列 1 に折り返されるからです。行および図表は現行左マージンから始まり、例は現行左マージンから 4 スペース字下げされます。ネストされたリストがフォーマットされると、現行左マージンは変わります。さまざまな位置 (リスト内を含む) に組み込まれた行、図表、または例を含むオンライン・ヘルプ情報がある場合、それは毎回同じように見えるわけではありません。

パラメーター・リストはテキスト内の任意の場所に出現できます。パラメーター・リストを他のリスト内にネストすることも、他のリストをパラメーター・リスト内にネストすることもできます。2 つの PT タグまたは 2 つの PD タグを連続して使用することはできません。

ネストされた PARML タグは、現行左マージンを 4 つずつ増やします。

他のタグを PARML タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 77. PARML タグと EPARML タグの間で許可されるタグ

タグ名	順序	使用法	ページ
FIG (図表)	1	B	577
LINES (不定形式行)	1	B	597
XMP (例)	1	B	697
NT (注記)	1	B	644
P (段落)	1	B	648
PC (段落の連結)	1	B	658
LP (リストの一部)	1	B	633
OL (番号付きリスト)	1	B	645
SL (単純リスト)	1	B	678
UL (順不同リスト)	1	B	690
PARML (パラメーター・リスト)	1	B	656
DL (定義リスト)	1	B	575

必須タグ

PT,parameter-term

パラメーター用語。用語は強調表示句 (HP2) でフォーマットされ、プログラミング・キーワード (PK) またはプログラミング変数 (PV) タグを含めることができます。

PARML、PT、PD タグ

用語は単独で 1 行に表示され、現行左マージンから見て 4 桁目で開始します。

PD,parameter-description

パラメーターの記述。また、PK および PV タグを含めることができます。

定義は次の行に印刷され、このとき段落は現行左マージンから 8 スペース字下げされます。これは暗黙の段落であり、テキスト項目を含めることができます。段落タグを使用して、記述段落の後に追加の段落を挿入できます。

例: パラメーター・リスト

この例は、パラメーター・リストを使用して、用語またはパラメーターを定義します。

UIM ソース

```
:PARML.  
:PT.TERM  
:PD.This is a description of  
the term. :PK.Term:EPK. is  
a programming keyword.  
:PT.:PK def.DEFAULT:EPK.  
:PD.This is a sample default  
parameter.  
:PT.:PV.variable:EPV.  
:PD.This is a parameter variable.  
:EPARML.
```

結果

TERM

This is a description of the term. **Term** is a programming keyword.

DEFAULT

This is a sample default parameter.

variable

This is a parameter variable.

PC (段落の連結)

```
▶▶:PC.—paragraph-continuation-text—▶▶
```

段落の連結 (PC) タグは、別の文書エレメントによって中断されている段落の連結を識別します。このタグは情報域とヘルプ域でのみ許可されます。

段落の連結は通常、図表、例、またはリストの後に行われます。これは、後に続くテキストが、図表、例、またはリストによって中断された段落の連結であることを示します。

対応する終了タグは許可されません。段落の連結は、別の段落またはさらに高いレベルのエレメントによって暗黙的に終了します。

例: 段落の連結

段落の連結タグは、例による中断後に段落を継続します。

UIM ソース

```
:p.If you enter the following
command
:xmp.
WRKSPLF
:exmp.
:pc.a listing of all my spool files
appears on the
terminal.
```

結果

If you enter the following command

```
WRKSPLF
```

a listing of all my spool files appears on the terminal.

PDACCEL (プルダウン・アクセラレーター)

```
▶▶:PDACCEL—.—accelerator-text—▶▶
```

プルダウン選択項目アクセラレーター (PDACCEL) タグは、表示されるテキストを、プルダウン選択項目のアクセラレーター・キーとして定義します。

このタグは、表示されるテキストだけを定義します。これは、機能キーを自動的に定義することはしません。プルダウン選択項目アクセラレーターに定義されたアクションを実行する機能キーを、キー・リスト項目 (KEYI) タグで定義するのは、ユーザーの責任です。

必須テキスト

accelerator-text

アクセラレーター・キーの表示可能な記述。テキストは、タグと同じ行または次の行に表示される必要があります。テキストには他のタグを含めることはできず、長さが 4 バイトを超えることはできません。

アクセラレーター・テキストは、そのプルダウン選択項目の右に表示されます。プルダウン・フィールドの各アクセラレーターの先頭文字は、プルダウン選択項目の最長テキストより 2 スペース後の位置に、左揃えで配置されます。

PDFLD (プルダウン・フィールド)

```
▶▶:PDFLD—[NAME—=pull-down-field-name]—.—:EPDFLD.—▶▶
```

プルダウン・フィールド (PDFLD) タグは、メニュー・バー・プルダウン内の選択フィールドを定義します。プルダウン・フィールドは、1 つ以上のプルダウン選択項目から成っています。プルダウン・フィールドは、IBM システム・アプリケーション体系 (SAA) の基本インターフェースに従ってフォーマット設定されます。

プルダウン・フィールド選択項目 (PDFLDC) タグは、プルダウン・フィールド内の 1 つの選択項目を定義します。プルダウン・フィールドは、1 つ以上のプルダウン選択項目から成っています。

必須属性

OPTION=*option-number*

このオプションに割り当てられる番号。オプション番号は、1 から 99 の整数です。

プルダウン選択項目は、プルダウン・フィールドで定義した順序で表示されます。2 つの選択項目の番号が連続していない場合は、2 つの選択項目の間に空白行が自動的に挿入されます。

HELP=*help-module-name*

プルダウン・フィールド選択項目の目的を説明するヘルプ情報を識別します。ヘルプ・モジュールの名前は、別のパネル・グループからインポートされた名前にすることができますが、この章の前の部分で概説した名前の規則に従っている必要があります。名前の規則について詳しくは、515 ページの『名前構文』を参照してください。

カーソルがこのタグのテキスト上にあるときにヘルプが要求されると、この属性によって識別されるオンライン情報が表示されます。このオンライン情報は、カーソルがこのプルダウン・メニューのメニュー・バー選択項目のテキスト上にある場合、またはカーソルがプルダウン・メニュー内にあるが、いずれかの選択項目のテキスト上にはない場合に表示される文脈ヘルプの一部としても組み込まれます。カーソルが入力フィールドにあり、ヘルプの要求時に有効な選択項目が入力されていた場合は、その選択項目のヘルプが表示されます。

ACTION=*'action-text'*

プルダウン選択項目が選択された場合に生じるアクション。

表 79. ACTION 値の有効アクション・テキスト

ダイアログ・コマンド	ACTFOR= PANEL	ACTFOR= LIST
CALL	X	X
CHGVIEW	X	
CMD	X	X
CMDLINE	X	
DSPHELP	X	
EXIT	X	
EXTHelp	X	
HELPHELP	X	
HELPIDX	X	
KEYSHelp	X	
MENU	X	
RETRIEVE	X	
RETURN	X	X

これらの各アクションについては、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

オプションの属性

ACTFOR= PANEL | LIST

PANEL が指定された場合、プルダウン選択項目は、パネル全体に対してアクションを実行します。

LIST が指定された場合、プルダウン選択項目は、選択リストまたは処置リストの項目に対してアクションを実行します。

PDFLDC タグ

プルダウン選択項目に LIST が指定されていても、処理されるリストがユーザーによって選択されていない場合、プルダウン選択項目は、オプション番号がアスタリスク (*) (プルダウン選択項目が選択不可であることを示す) でオーバーレイされた状態で表示されます。

CONFIRM=*internal-panel-name*

アクションが実行される前に表示される、確認パネルの名前。この属性が許可されるのは、このタグに ACTFOR=LIST と、ACTION=CALL または ACTION=CMD が指定された場合だけです。確認パネルはユーザーに、アクションを確認するか、確認しないかのオプションを提供しなければなりません。アクションが確認された場合は、このタグの適切なアクションが実行されます。アクションが確認されなかった場合は、それらのアクションのいずれも実行されません。

確認パネルは、このパネル・グループ内に定義された別のパネルでなければなりません。

この属性の推奨規則および必須規則のリストについては、663 ページの『確認パネルの要件』および 664 ページの『確認パネルの規則』を参照してください。

USREXIT='CALL program-reference'

ACTION 属性で定義されたアクションの実行後、リスト項目を更新するために呼び出される処置リスト・プログラムの出口プログラムを指定します。この属性が許可されるのは、このタグに ACTFOR=LIST と、ACTION=CALL または ACTION=CMD が指定された場合だけです。プログラムには、リストの名前、およびオプションが成功したか失敗したかを示す標識を含む情報が渡されます。

CALL ダイアログ・コマンドの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

UIM と処置リストの出口プログラム間のインターフェースについては、iSeries Information Center の API のトピックを参照してください。

VARUPD=YES | NO

この属性が許可されるのは、このタグの ACTFOR 属性で PANEL が指定された場合だけです。

ACTFOR=LIST が指定された場合は、変数プール更新が実行されます。YES が指定された場合、妥当性検査が行われ、アクションが実行される前に変数プールが更新されます。

NO が指定された場合、アクションが即時に実行され、変数プール更新は生じません。

ダイアログ・コマンド	VARUPD 値
CALL	YES NO
CHGVIEW	YES
CMD	YES NO
CMDLINE	NO
DSPHELP	NO
EXIT	NO YES
EXTHelp	NO
HELPHELP	NO
KEYSHELP	NO
HELPIIDX	NO
MENU	YES
RETRIEVE	NO
RETURN	YES NO

注: デフォルト値は ACTION 属性を基にしています。最初の値がデフォルト値です。

COND=*condition-name*

指定された条件が真の場合にのみ、パネル上のプルダウン選択項目が有効になります。条件は、パネ

ル・グループ・プロログ内で条件定義 (COND) タグを使用して定義する必要があります。選択項目が条件によってオフになると、それはプルダウンに表示されず、その選択項目のヘルプはヘルプを表示するための要求に組み込まれなくなります。

AVAIL=condition-name

プルダウン選択項目が使用可能かどうかを判別する、UIM によって使用される条件の名前。条件は、パネル・グループ・プロログ内で COND タグを使用して定義する必要があります。

条件が真の場合、プルダウン選択項目は使用可能です。条件が偽の場合、プルダウン選択項目は使用不可です。このタグの COND 属性で指定されている条件があれば、その条件がこの属性より優先して使用されます。

選択不可能な項目は、カラー装置では別の色で表示され、選択項目のオプション番号の最初の部分がアスタリスク (*) で表示されます。

AVLMSGID=message-identifier

AVAIL 属性での指定によって、プルダウン選択項目が使用可能でない場合に、その項目が選択されると表示されるメッセージのメッセージ ID。この属性が許可されるのは、このタグで AVAIL 属性が指定されている場合だけです。この属性が指定されない場合、UIM は選択項目が現在使用可能でないことを告げるデフォルト・メッセージを表示します。

AVLMSGF='qualified-message-file-name'

メッセージ ID が入っているメッセージ・ファイル名。この属性が許可されるのは、このタグで AVLMSGID 属性が指定されている場合だけです。パネル・グループ (PNLGRP) タグに DFTMSGF 属性が指定されていない場合で、このタグに AVLMSGID 属性が指定されている場合は、この属性を指定する必要があります。

CHOICE=dialog-variable-name

表示するプルダウン選択項目テキストを含むダイアログ変数の名前。ダイアログ変数は、32 バイトより小か等しい幅で定義しなければなりません。この属性が使用される場合、*pull-down-choice-text* は指定できません。

ダイアログ変数は、クラス定義 (CLASS) タグで BASETYPE に CHAR、IGC、または BIN を指定して定義する必要があります。

ダイアログ変数のエラー状態は、テキストの強調表示の判別には使用されません。

IGC 用の特殊フォーマット。 (省略形 IGC は、コマンドおよびキーワードで使用され、2 バイト文字セット機能を表します。) BASETYPE が IGC のダイアログ変数を CLASS タグ上で指定すると、UIM は特殊フォーマット設定を行います。変数値がシフトアウト文字 (X'0E') で始まる場合は、UIM は値を左に 1 バイト移動して、他の行の選択項目と垂直位置合わせを行います。

オプションのテキスト

pull-down-choice-text

プルダウン・フィールドに表示される説明テキスト。テキストは、ソース内の単一行に表示できます。テキストはテキスト反転 (RT) タグのみ含めることができ、長さが 32 バイトを超えることはできません。このタグに CHOICE 属性を指定しない限り、*pull-down-choice-text* は必須です。

プルダウン選択では、オプション番号、ピリオド、および 1 つのブランクが、選択記述テキストの直前に表示されます。PNLGRP タグの NBRSHAPE 属性は、BIDI=LTR または BIDI=RTL が PNLGRP タグで指定されたときに、オプション番号の表示される仕方を説明します。

確認パネルの要件

このタグの CONFIRM 属性で指定されたパネルは、以下の規則に従っていなければなりません。

PDFLDC タグ

- 確認パネルの表示パネル (PANEL) タグ上の ENTER 属性は、'RETURN 100' でコード化されていなければなりません。これは UIM に、アクションがユーザーによって確認されることを示します。
- キー・リスト項目 (KEYI) タグ上の ACTION 属性は、RETURN アクションに割り当てることはできません。これは、UIM はパネルの表示を処理するため、戻り値を解釈できないことによります。
- 確認パネルにはリスト域がなければなりません。
- 確認リストとして使用されるリストを、処置リスト・パネル上でも使用してはなりません。
- 確認パネルには、処置リスト、コマンド行、メニュー域、またはメニュー・バーがあってはなりません。

確認パネルの規則

このタグの CONFIRM 属性で指定されたパネルは、以下の規則に従っていなければなりません。

- パネルには、それを説明するのに適切なタイトル・テキストと最上部指示が必要です。
- 元のアクションまたは選択リストで定義されたものと同じ列 (またはそのサブセット) を含むリスト定義を持つ、単一のリスト域がなければなりません。
- リストはスクロール可能であるべきです。
- 処置リストに複数のビューがある場合、確認リストにも複数のビューが必要であり、同じダイアログ変数が両方のリスト上のビュー変数に使用される必要があります。確認パネルにはまた、CHGVIEW ダイアログ・コマンド用に定義された機能キーも必要です。これにより、確認パネルでのビュー変更操作が許可され、アクションまたは選択リスト・パネルが再表示されるときに、そのビューを変更できます。
- 確認パネルには ENTER、CANCEL、PAGEUP、PAGEDOWN、HELP、および PRINT ダイアログ・コマンドの機能キー定義が必要です。
- 汎用出口プログラムが定義される場合、元のパネルの再表示に影響を与えるリストまたは条件値に、変更がなされないようにすべきです。
- 確認パネルで使用されるリストは、確認パネル以外では使用すべきではありません。UIM はそのリストの内容を削除および変更するためです。複数の確認パネルが UIM アプリケーション内で使用される場合、それらは同じリストを共有できます。
- 処置リスト・パネルのオンライン・ヘルプ情報の大半は、確認パネルで使用できますが、アクション列および拡張パネル・ヘルプに対して、個別にヘルプ情報を提供する必要があります。

PK (プログラミング・キーワード)

```
▶▶:PK [DEF] .—programming-keyword-text—:EPK.▶▶
```

プログラミング・キーワード (PK) タグは、プログラミング・キーワードを識別します。このタグは、情報域とヘルプ領域でのみ許可されます。これには、対応する終了タグが必要です。

プログラミング・キーワードは、テキスト内の任意の箇所に配置できます。これは、プログラミング構文の要素の説明に役立ち、パラメーター・リスト内で頻繁に使用されます。パラメーター・リストについて詳しくは、656 ページの『PARML (パラメーター・リスト)』を参照してください。

PK タグおよび EPK タグ句は、ワード境界上で指定しなければなりません。EPK タグの直後に続く 2 文字が、句読記号とブランクである場合、UIM は強調属性を自動的に拡張して、句読記号を含めます。これにより、句読記号とそれに関連したテキストを、同じ強調を使用して表示できます。

オプションの属性

DEF

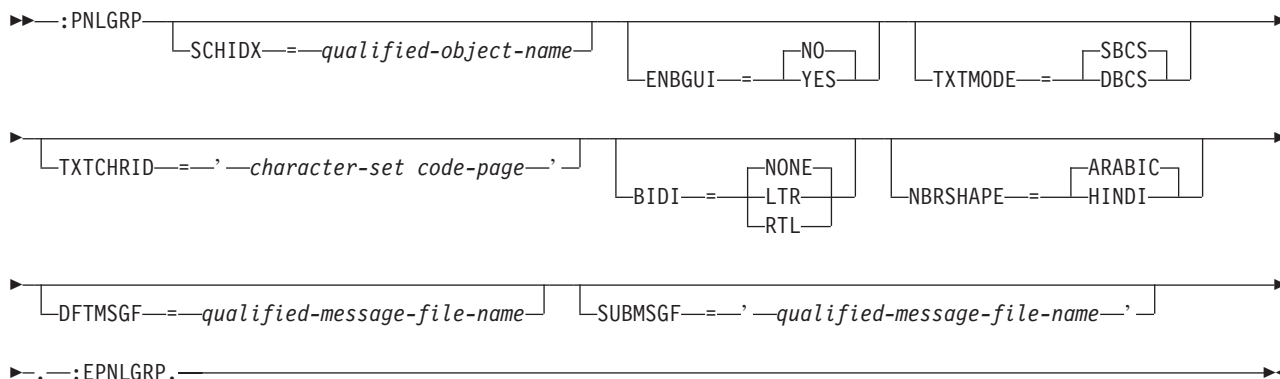
プログラミング・キーワードがデフォルト値であることを指定します。プログラミング・キーワードは、DEF 属性が指定されない限り、強調表示句 2 (HP2) でフォーマット設定されます。DEF 属性が指定された場合、プログラミング・キーワードは強調表示句 3 (HP3) としてフォーマット設定されます。

必須テキスト

programming-keyword-text

プログラミング・キーワードを指定します。

PNLGRP (パネル・グループ)



パネル・グループ (PNLGRP) タグは、パネル・グループの定義で始まります。1 つの PNLGRP タグだけが許可され、それに対応する EPNLGRP タグが必須です。

他のタグを PNLGRP タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 80. PNLGRP タグと EPNLGRP タグの間で許可されるタグ

タグ名	順序	ページ
COPYR (著作権)	1	544
ISCHSYN (索引検索同義語)	任意	591
IMPORT (インポート)	2	586
CLASS (クラス定義)	3	525
VAR (変数定義)	4	692
VARRCD (変数レコード定義)	5	695
LISTDEF (リスト定義)	6	625
COND (条件定義)	7	540

PNLGRP タグ

表 80. PNLGRP タグと EPNLGRP タグの間で許可されるタグ (続き)

タグ名	順序	ページ
TT (真理値表)	8	688
MBAR (メニュー・バー)	9	633
KEYL (キー・リスト)	10	596
PANEL (表示パネル)	11	649
PRTHEAD (印刷ヘッド・パネル)	11	668
PRTPNL (印刷パネル)	11	673
HELP (ヘルプ・モジュール)	11	579

オプションの属性

SCHIDX=*qualified-object-name*

このパネル・グループで定義されたパネルのヘルプ・モジュールから、索引検索機能が要求されたときに使用される索引検索オブジェクト。SCHIDX が指定されない場合、このパネル・グループ内のパネルのヘルプ・モジュールが表示される時、索引検索機能は利用できません。この属性は、パネル・グループ内のパネルがヘルプ・モジュールを使用するのでない限り、パネル・グループで定義されたヘルプ・モジュールに対して何の影響も及ぼしません。

ENBGUI=NO | YES

パネル・グループ内のすべてのパネルでクライアント・プログラムを使用可能にするデフォルト値を指定します。この属性は、PANEL タグで ENBGUI=PNLGRP が指定された場合、またはそれがデフォルトに設定された場合に、値を設定します。

ENBGUI=YES がパネルに指定された場合、UIM には 5250 データ・ストリーム内のパネルのレイアウトおよび内容についての情報が組み込まれます。クライアント・プログラムはこの情報を使用して、クライアントのグラフィカル・インターフェースを作成します。652 ページの表 76 では、5250 データ・ストリームに組み込まれる情報を説明しています。

ENBGUI=NO がパネルに指定された場合、UIM はデータ・ストリームに余分の情報を組み込みません。

TXTMODE=SBCS | DBCS

1 バイト文字セット (SBCS) または 2 バイト文字セット (DBCS) のどちらをタグ・テキスト内で使用するかを指定します。DBCS が指定された場合、タグ・テキストには DBCS テキストの開始および終了を示すシフトイン文字またはシフトアウト文字がなければなりません。

TXTMODE=DBCS が指定されるパネル・グループ・オブジェクトを使用できるのは、DBCS 情報を処理できるディスプレイ装置およびシステム上に限られます。DBCS をサポートしない装置上で DBCS メニューの表示、DBCS パネル・グループ・オブジェクトのオープン、または DBCS ヘルプ情報の表示を試行すると、例外が発生します。

TXTCHRID=*'character-set code-page'*

パネル・グループ・ソース内の SBCS テキスト・データ用の文字セットおよびコード・ページを指定します。この情報は、索引検索同義語 (ISCHSYN) タグを使用して定義された同義語に対して使用されます。このタグについて詳しくは、591 ページの『ISCHSYN (索引検索同義語)』を参照してください。この属性が省略される場合は、すべてのテキストが索引検索 (ISCH) タグおよび ISCHSYN タグ上にあり、パネル・グループの作成時に QCHRID システム値によって指定された文字セットおよびコード・ページのものでなければなりません。

TXTCHRID 属性は、パネル・グループ・ソース内のテキスト情報のコード・ページおよび文字セットを指定します。これを CRTPNLGRP コマンドの CHRID パラメーターと混同してはなりません。後者

は、クラス定義 (CLASS) タグで CHRID=PNLGRP を指定したダイアログ変数が、ディスプレイ装置との間でダイアログ変数の値が受け渡しされる際に処理される仕方を制御します。

BIDI=NONE | LTR | RTL

パネル・グループ内のパネルの方向を指示します。NONE は、パネル・グループ内のパネルが、左から右の単一方向で表示されることを指示します。NONE はデフォルトで、このタグに TXTMODE=DBC が指定されたときに必須です。

BIDI=NONE がコード化されると、以下の項目が無視されます。

- このタグの NBRSHAPE 属性
- CLASS タグの BIDI、NBRSHAPE、CONXTREV、および SYMSWAP 属性
- すべてのテキスト反転 (RT) タグ

LTR は、パネル・グループ内のパネルが、双方向であり、左から右の方向で表示されることを指示します。

RTL は、パネル・グループ内のパネルが、双方向であり、右から左の方向で表示されることを指示します。

LTR または RTL のいずれかがパネル・グループに指定されると、ワークステーション・コントローラーは、パネルを反転させるモード切り替えキーを使用可能にし、ユーザーが必要であればパネルを反転できるようにします。RTL が指定されると、コントローラーは、パネル・グループのすべてのパネルを、表示されるときに自動的に反転させます。

LTR は通常、左から右方向の言語で書かれ、その中にごくわずかに右から左方向の言語テキストが存在するパネルで使用されます。RTL は通常、右から左方向の言語で書かれ、その中にごくわずかに左から右方向の言語テキストが存在するパネルで使用されます。

UIM は LTR または RTL パネルが、双方向言語の正確なコード・ページを持たないディスプレイ装置で使用されることを許可しません。ヘブライ語はコード・ページ 424 を使用し、アラビア語はコード・ページ 420 を使用します。

ハイパーテキスト・リンクを介して到達できるすべてのパネル・グループについても、この属性の値は同じでなければなりません。参照句が選択されたときに、属性値が同じでない場合、表示される画面またはポップアップ・ウィンドウは、反対の方向で表示される可能性があります。

NBRSHAPE=ARABIC | HINDI

このタグで BIDI=NONE が指定されている場合、この属性は無視されます。NBRSHAPE 属性は、UIM の生成する番号の表示形状を制御します。コード・ページ 420 (アラビア語) が使用されるときには、この属性は無視されます。

ARABIC が指定されたときには、通常の数値 'XF0' から 'XF9' が番号として使用されます。ARABIC がデフォルトです。

HINDI が指定されたときには、各国語に定義された変換テーブルを使用した数値の変換が実行されます。

パネル・グループ・ソースまたはメッセージの数値は、NBRSHAPE 属性によって影響されません。なぜなら、それらはすでに正確な文字に変換されていると予期されるからです。NBRSHAPE 属性によって影響される数値は、以下のとおりです。

- メニュー項目用に UIM によって生成される番号
- 番号付きリスト番号用に UIM によって生成される番号

PNLGRP タグ

- プルダウン・フィールド選択項目用に UIM によって生成される番号
- 印刷パネルおよび印刷されたヘルプ情報上のページ番号情報

NBRSHAPE 属性によって影響されない番号は、以下のとおりです。

- システム名で使用される数字
- メッセージ行で表示されるメッセージ
- メッセージ行の著作権日付
- 印刷パネルおよび印刷されたヘルプ情報の、製品番号情報

ヒンディ語のゼロはピリオドに似ているため、メニュー項目またはプルダウン選択の前の番号の形式は、'n.'ではなく 'n' です ('n' は、メニュー項目または選択オプションの番号)。

DFTMSGF= *qualified-message-file-name*

以下のタグのいずれかで、メッセージ ID に対してメッセージ・ファイルが指定されなかったときに使用されるデフォルトのメッセージ・ファイル。

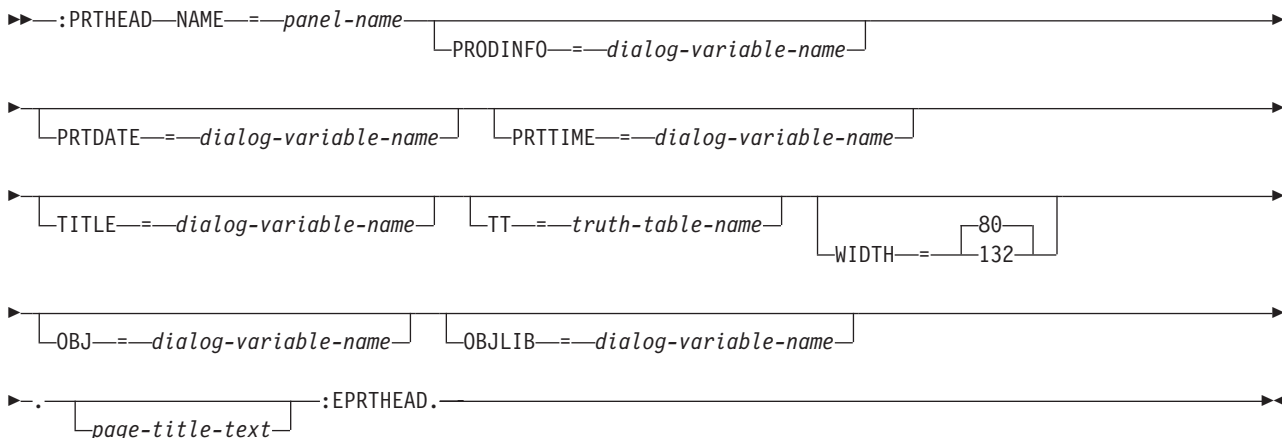
CHECK	妥当性検査
LISTDEF	リスト定義
TL	変換リスト
DATASLTC	データ選択フィールド選択項目
PDFLDC	プルダウン・フィールド選択項目
LISTACT	リスト・アクション
MENUI	メニュー項目

対応するメッセージ・ファイル属性がこれらのタグのいずれでも指定されていない場合で、このタグに MSGID 属性が指定されたときには、この属性は必須です。

SUBMSGF= *qualified-message-file-name*

&msg シンボル上でメッセージ・ファイルが指定されなかったときに使用される、デフォルトのメッセージ・ファイル。

PRTHEAD (印刷ヘッド・パネル)



印刷ヘッド・パネル (PRTHEAD) タグには、UIM 印刷アプリケーションで使用されるヘッダー情報およびデータを定義するタグが含まれます。印刷ヘッド・パネルは、最初のページのタイトル行の後に印刷されるプロローグ・セクション、およびすべてのページの上部に印刷されるヘッダー・データを設定します。プロローグ・セクションとヘッダー・セクションは、データ域および情報域から構成できます。PRTHEAD パネルは、自動ページ排出を引き起こします。

他のタグを PRTHEAD タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 81. PRTHEAD タグと EPRTHEAD タグの間で許可されるタグ

タグ名	順序	使用法	ページ
DATA (データ表示域)	1	P	544
INFO (情報域)	1	P	587
PRTRAIL (トレーラー印刷)	2	P	675

必須属性

NAME=panel-name

印刷ヘッド・パネルの名前。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

オプションの属性

PRODINFO=dialog-variable-name

印刷出力の 2 番目のタイトル行に印刷される製品情報を含む、ダイアログ変数の名前。

ダイアログ変数は、クラス定義 (CLASS) タグで BASETYPE に CHAR または IGC を指定して定義しなければなりません。最大幅は 22 バイトです。

PRTDTE=dialog-variable-name

ページ・タイトルに印刷される日付を含む、ダイアログ変数の名前。この属性が指定された場合、PRTTIME 属性もこのタグで指定しなければなりません。この属性が指定されない場合には、Open Print Application (QUIOPNPA) API または Add Print Application (QUIADDPA) API が呼び出される際に、UIM は現行システム日付を使用します。

ダイアログ変数は、CLASS タグで BASETYPE に DATE を指定したクラスで定義しなければなりません。

PRTTIME=dialog-variable-name

ページ・タイトルに印刷される時刻を含む、ダイアログ変数の名前。ダイアログ変数は、CLASS タグで BASETYPE に TIME を指定したクラスで定義しなければなりません。この属性が指定された場合、PRTDTE 属性もこのタグで指定しなければなりません。この属性が指定されない場合には、QUIOPNPA API または QUIADDPA API を呼び出す際に、UIM は現地時間を使用します。

PRTHEAD タグ

印刷幅が 80 列で、ダイアログ変数にタイム・ゾーンを含めることができる場合、このダイアログ変数のタイム・ゾーン部分は無視されます。しかし、このタグから OBJ および OBJLIB 属性が省略される場合には、タイム・ゾーン値を表示できます。

TITLE=*dialog-variable-name*

ページ・タイトル行のテキストを含む、ダイアログ変数の名前。ダイアログ変数は、CLASS タグで表示幅 55 バイトを指定して定義しなければなりません。変換リストを使用して、パネル・タイトルまたはパネルの幅から 2 引いた数のどちらか小さい方を提供することもできます。またダイアログ変数は CLASS タグで Basetype を CHAR 55 または IGC 55 に指定して定義することもできます。UIM はストリングから末尾ブランクを切り取り、その結果できたテキストをページ・タイトル域の中央に配置します。この属性が使用される場合、*title-text* は指定できません。

TT=*truth-table-name*

パネル・ヘッ드의フォーマット設定時の実行時に、生じる可能性がある真理値の組み合わせを指定する真理値表の名前。

このタグで指定される表には、印刷ヘッド・パネル内の、タグによって参照される条件のサブセットを含めることができます。表で指定される真理値の組み合わせ、および表に指定されていない真理値のワーストケースの想定だけが、印刷ヘッド・パネルが有効かどうかを評価する際に考慮されます。実行時に印刷ヘッド・パネルがフォーマット設定されるときに生じる可能性のある真理値の組み合わせを、表から除外してはなりません。有効な真理値の組み合わせが省略された場合、パネル・グループ・オブジェクトをエラーなしで作成できても、パネルの印刷時に不本意な結果が生じる可能性があります。

この属性が省略される場合、真理値のすべての組み合わせが可能であるとみなされます。この場合、タグ言語コンパイラーは、印刷ヘッド・パネルが有効かどうかを評価する際、すべての条件に対してワーストケースの想定を行います。

WIDTH=80 | 132

PRTHEAD パネルの幅を定義します (バイト単位)。デフォルト値は 80 バイトです。

OBJ=*dialog-variable-name*

その情報が印刷出力のタイトル行に印刷されるオブジェクトの名前を含むダイアログ変数の名前。この属性を指定しない場合には、オブジェクト名が印刷されません。

このタグに OBJLIB 属性が指定される場合には、このダイアログ変数のクラスを、10 文字の幅を指定して定義しなければなりません。OBJLIB 属性が指定されない場合には、このダイアログのクラスを 1 文字から 21 文字の幅を指定して定義しなければなりません。

OBJLIB=*dialog-variable-name*

このタグの OBJ 属性から指定されたオブジェクトがあるライブラリーの名前を含む、ダイアログ変数の名前。このライブラリー名は、印刷出力のタイトル行に印刷されます。この属性を指定しない場合には、ライブラリー名が印刷されません。

OBJ および OBJLIB 属性に含まれるデータの妥当性検査は、ユーザーの責任です。UIM はこれらの変数に対して妥当性検査を実行しません。OBJ 属性だけが指定された場合、OBJ だけが印刷されます。しかし、OBJLIB 属性だけが指定された場合は、コンパイル時にメッセージが発行されます。

このダイアログ変数のクラスは、10 文字の幅を指定して定義しなければなりません。

オプションのテキスト

page-title-text

タイトル行で使用されるタイトル。テキストは、タグと同じ行または次の行に表示される必要があります。テキストにはテキスト反転 (RT) タグだけを含めることができ、長さが 55 文字を超えることはできません。このタグに TITLE 属性を指定しない限り、*page-title-text* は必須です。

タイトル行のレイアウト

PRTHEAD タグのタイトル行は、それぞれの印刷ページの先頭 2 行に表示されます。タイトル行内のフィールドの配置は、このタグに指定された幅に依存しています。

表 82. 印刷幅が 132 のヘッダーの最初の行

データ	開始列	長さ
タイトル	注を参照	最大 55
区切り記号	注を参照	4
"ページ" (拡張付き)	注を参照	最大 11
区切り記号	125	2
ページ番号	127	4
区切り記号	131	2

注: このデータの開始列は可変です。タイトルは、左マージンと、タイトルに続く区切り記号の間の中央に配置されません。

表 83. 印刷幅が 80 のヘッダーの最初の行

データ	開始列	長さ
タイトル	注を参照	最大 55
区切り記号	注を参照	4
"ページ" (拡張付き)	注を参照	最大 11
区切り記号	73	2
ページ番号	75	4
区切り記号	79	2

注: このデータの開始列は可変です。

表 84. 印刷幅が 132 のヘッダーの 2 番目の行

データ	開始列	長さ
製品情報	1	22
区切り記号	23	最小 55
ライブラリー	注を参照	最大 10
スラッシュ	注を参照	1
オブジェクト	注を参照	最大 10
区切り記号	99	4
システム名	103	8
区切り記号	111	2
日付	113	8
区切り記号	121	2
時間	123	8
区切り記号	131	2

注: ライブラリーおよびオブジェクトは、OBJLIB/OBJ としてフォーマット設定され、末尾空白が OBJLIB および OBJ 値から除去されます。OBJLIB/OBJ 集合体は、システム名の前の区切り記号に対して右寄せにされます。

PRTHEAD タグ

表 85. 印刷幅が 132 でタイム・ゾーンを持つヘッダーの 2 番目の行

データ	開始列	長さ
製品情報	1	22
区切り記号	23	最小 55
ライブラリー	注 1 および 2 を参照	最大 10
スラッシュ	注 1 および 2 を参照	1
オブジェクト	注 1 および 2 を参照	最大 10
区切り記号	注 2 を参照	4
システム名	注 2 を参照	8
区切り記号	注 2 を参照	2
日付	注 2 を参照	8
区切り記号	注 2 を参照	2
時間	注 2 を参照	8
区切り記号	注 2 を参照	1
タイム・ゾーン	注 2 を参照	最大 10
区切り記号	131	2

注:

- ライブラリーおよびオブジェクトは、OBJLIB/OBJ としてフォーマット設定され、末尾ブランクが OBJLIB および OBJ 値から除去されます。OBJLIB/OBJ 集合体は、システム名の前の区切り記号に対して右寄せにされます。
- 末尾ブランクはタイム・ゾーンから除去され、ライブラリーからタイム・ゾーンまでのすべての情報が、ヘッダーの終了前の区切り記号に対して右寄せにされます。

表 86. 印刷幅が 80 のヘッダーの 2 番目の行

データ	開始列	長さ
製品コード	1	最大 22
区切り記号	23	最小 3
ライブラリー	注を参照	最大 10
スラッシュ	注を参照	1
オブジェクト	注を参照	最大 10
区切り記号	47	4
システム名	51	8
区切り記号	59	2
日付	61	8
区切り記号	69	2
時間	71	8
区切り記号	79	2

注: ライブラリーおよびオブジェクトは、OBJLIB/OBJ としてフォーマット設定され、末尾ブランクが OBJLIB および OBJ 値から除去されます。OBJLIB/OBJ 集合体は、システム名の前の区切り記号に対して右寄せにされます。

- 印刷幅が 80 列のパネルにタイム・ゾーンを印刷するためには、OBJ および OBJLIB 属性を省略しなければなりません。

表 87. 印刷幅が 80 でタイム・ゾーンを持つヘッダーの 2 番目の行

データ	開始列	長さ
製品コード	1	最大 22
区切り記号	23	最小 1
システム名	注を参照	8
区切り記号	注を参照	2
日付	注を参照	8
区切り記号	注を参照	2
時間	注を参照	8
区切り記号	注を参照	1
タイム・ゾーン	注を参照	最大 10
区切り記号	注を参照	2

注: 末尾ブランクがタイム・ゾーンから除去され、システム名からタイム・ゾーンまでのすべての情報が、ヘッダーの終了前の区切り記号に対して右寄せにされます。

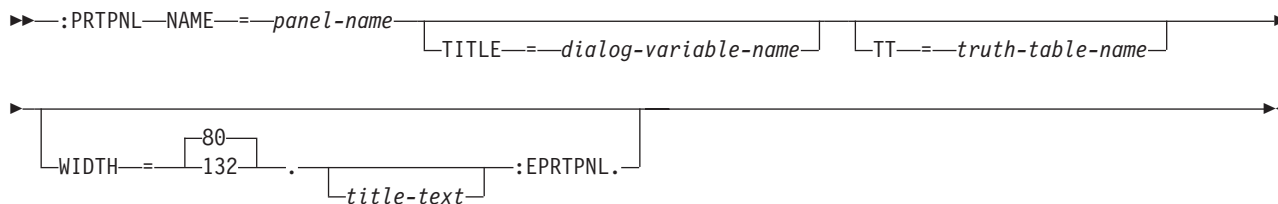
例: 印刷タイトル行

以下の例には、幅が 132 バイトの出力用のタイトル行があります。タイトルは、印刷ヘッド・パネルまたは先頭印刷パネルから取られています。ページ・テキストおよびページ番号が UIM によって提供されます。

UIM またはアプリケーションのいずれかが、日時を提供できます。アプリケーションは PRTHEAD タグ上で日時を指定するために PRDATE および PRTIME 属性を使用しなければなりません。PRDATE および PRTIME 属性が指定されない場合には、UIM は QUIOPNPA API または QUIADDPA API の呼び出し時にシステム日付および時刻を使用します。日時はジョブ属性を使用してフォーマット設定されます。

```
5722SS1 V5R4M0 060210 This is an Example of a Print Title Line Page 1
                        QNETUSER/EXAMPLE SYSTEM01 02/10/06 11:09:04
```

PRTPNL (印刷パネル)



印刷パネル (PRTPNL) タグには、印刷パネルの 1 つ以上の領域を定義するタグが含まれます。これには、対応する EP RTPNL タグが必要です。

他のタグを PRTPNL タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

PRTPNL タグ

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 88. PRTPNL タグと EPRTPNL タグの間で許可されるタグ

タグ名	順序	使用法	ページ
DATA (データ表示域)	1	P	544
INFO (情報域)	1	P	587
LIST (リスト域)	1	P	603

必須属性

NAME=panel-name

印刷パネルの名前。命名規則については、515 ページの『名前構文』を参照してください。

オプションの属性

TITLE=dialog-variable-name

パネルの印刷時のパネル・タイトルのテキストを含む、ダイアログ変数の名前。ダイアログ変数は、表示幅が 55 のクラスで定義しなければなりません。UIM はストリングから末尾空白を除去し、その結果できたテキストを印刷パネルのタイトル行の中央に配置します。この属性が使用される場合、*title-text* は指定できません。

TT=truth-table-name

パネル・ヘッドのフォーマット設定時の実行時に、生じる可能性がある真理値の組み合わせを指定する真理値表の名前。

このタグで指定される表には、印刷パネル内の、タグによって参照される条件のサブセットを含めることができます。表で指定される真理値の組み合わせ、および表に指定されていない真理値のワーストケースの想定だけが、印刷パネル定義が有効かどうかを評価する際に考慮されます。印刷パネルの印刷時に生じる可能性のある真理値の組み合わせを、表から除外してはなりません。有効な真理値の組み合わせが省略された場合、パネル・グループ・オブジェクトをエラーなしで作成できても、パネルの印刷時に不本意な結果が生じる可能性があります。

この属性が省略される場合、真理値のすべての組み合わせが可能であるとみなされます。この場合、タグ言語コンパイラは、印刷パネルが有効かどうかを評価する際、すべての条件に対してワーストケースの想定を行います。

WIDTH=80 | 132

印刷パネルの幅を定義します (バイト単位)。デフォルト値は 80 バイトです。

オプションのテキスト

title-text

この印刷パネルで使用されるタイトル。テキストは、タグと同じ行または次の行に表示される必要があります。テキストにはテキスト反転 (RT) タグだけを含めることができ、長さが 55 文字を超えることはできません。このタグに TITLE 属性が指定されている場合、*title-text* は許可されません。

PRTRAIL (トレーラー・メッセージ印刷)

▶▶:PRTRAIL.—trailer-message-text—▶▶

トレーラー・メッセージ印刷 (PRTRAIL) タグは、印刷ヘッド・パネル (PRTHEAD) タグ内でのみ使用でき、EPRTHEAD タグのすぐ前に出現します。トレーラー・メッセージは、印刷リストの末尾に印刷されません。トレーラーが指定されない場合、デフォルトのトレーラーは印刷されません。

注: トレーラーが印刷されるのは、Close Application (QUICLOA) API または Remove Application (QUIRMVPA) API のいずれかで、通常クローズのオプションを使用することによってプリンター・ファイルがクローズされた場合に限られます。異常クローズのオプションを使用してアプリケーションがクローズされた場合には、トレーラーは印刷されません。

必須テキスト

trailer-message-text

印刷リストの末尾に出現するテキスト。このテキストは、先導または後続のアスタリスク (*) なしで入力できます。UIM が、テキストの前後にアスタリスクを追加します。

テキストは、PRTHEAD タグで WIDTH=132 が指定されている場合は最大 106 バイト、WIDTH=80 が指定されている場合は最大 54 バイトを指定できます。テキストにはテキスト反転 (RT) タグのみを含めることができます。

例: トレーラー・メッセージ

この例は、トレーラー・メッセージ印刷が持つテキストの種類を示しています。

UIM ソース

```
:prtrail.END OF LISTING
```

PV (プログラミング変数)

▶▶:PV.—programming-variable—:EPV.—▶▶

プログラミング変数 (PV) タグは、プログラミング変数を識別します。これには、対応する終了タグが必要です。これらのタグは、情報域とヘルプ領域でのみ許可されます。これはパラメーター・リスト内で頻繁に使用されます。パラメーター・リストについて詳しくは、656 ページの『PARML (パラメーター・リスト)』を参照してください。

プログラミング変数は、プログラミング構文の要素を説明するヘルプ・テキスト内の任意の箇所で使用できます。

PV タグおよび EPV タグ句は、ワード境界上で指定しなければなりません。EPV タグの直後に続く 2 文字が、句読記号とブランクである場合、UIM は強調属性を自動的に拡張して、句読記号を含めます。これにより、句読記号とそれに関連したテキストを、同じ強調を使用して表示できます。

必須テキスト

programming-variable

プログラミング変数。テキストは強調表示句 1 (HP1) でフォーマット設定されます。

RT (テキスト反転)

▶—:RT.—reverse-direction-text—:ERT.—◀

テキスト反転 (RT) タグは、囲まれたテキストの方向が、パネル・グループの方向と反対であることを示します。このタグは、パネル・グループ (PNLGRP) タグで `BIDI=NONE` が指定されたパネル・グループの場合は、無視されます。`BIDI=RTL` パネル・グループでは、左から右方向の言語テキストが RT と ERT タグの間に置かれ、`BIDI=LTR` パネル・グループでは、右から左方向の言語テキストが RT と ERT タグの間に置かれます。

インポート (IMPORT) タグでは、`BIDI=RTL` が指定されたパネル・グループのヘルプ情報を、`BIDI=LTR` が指定されたパネル・グループのヘルプ情報と結合させることができます。これは推奨されてはませんが、それが生じたときには、ワークステーション・コントローラーが画面を反転させて、反対方向のヘルプ情報を読み取るよう、ユーザーはモード切り替え操作を使用する必要があります。RT と ERT タグ内のテキストは、そのパネル・グループの方向に対して正確にフォーマット設定されます。したがって、反対方向のヘルプ情報にインポートされるテキストに対して RT タグが使用されるとき、インポートされるテキストも、画面が反転されるときに読み取ることができます。

強調表示句タグとは異なり、RT タグは大半のタグのテキスト内で使用できます。以下のタグは、組み込み RT タグを許可します。

BOTINST	最下部指示
CIT	タイトルの引用
CMDLINE	コマンド行
DATA	データ域
DATASLTC	データ選択フィールド選択項目
DATAGRP	データ・グループ
DATAI	データ項目
DD	定義
DDHD	定義ヘッダー
DT	定義用語
DTHD	定義用語ヘッダー
FIGCAP	図表表題テキスト
Hn	見出し
HPn	強調表示句
INFO	情報域
KEYI	キー・リスト項目
LINK	ハイパーテキスト・リンク定義

LIST	リスト域
LISTACT	リスト・アクション
LISTGRP	リスト・グループ
LI	リスト項目
LP	リストの一部
MBARC	メニュー・バー選択項目
MENU	メニュー域
MENUI	メニュー項目
NT	注記
OPTLINE	オプション行
P	段落
PC	段落の連結
PD	パラメーター記述
PT	パラメーター条件
PRTHEAD	印刷ヘッド・パネル
PRTPNL	印刷パネル
PRTRAIL	トレーラー・メッセージ印刷
PDFLDC	プルダウン・フィールド選択項目
TOPINST	最上部指示行

RT タグは 図表 (FIG) タグ、不定形式行 (LINES) タグ、または例 (XMP) タグ内では使用できません。
RT タグとそれに対応する ERT タグとの間には、タグは許可されません。

RT タグとそれに対応する ERT タグとの間のブランクは、保存されません。これには、それぞれのソース行末尾にある末尾ブランクが含まれます。したがって、ソース行を ERT タグで開始することは推奨されません。

例 1: 右から左方向のパネル上の、左から右方向のフォーマット設定

この例は、右から左方向のパネル上でフォーマット設定され表示される、左から右方向のソースを図示しています。

UIM ソース

```
:P.For more information see the
:CIT.:RT.IBM i5/OS User Interface
Manager Reference Manual:ERT.:ECIT.
which you can find in your library.
```

結果

```
IBM i5/OS User eht ees noitamrofni erom roF
uoy hcihw Interface Manager Reference Manual
.yrarbil ruoy ni dnif nac
```

例 2: 左から右方向のパネル上の、左から右方向のフォーマット設定

この例は、左から右方向のパネル上でフォーマット設定され表示される、左から右方向のソースを図示しています。

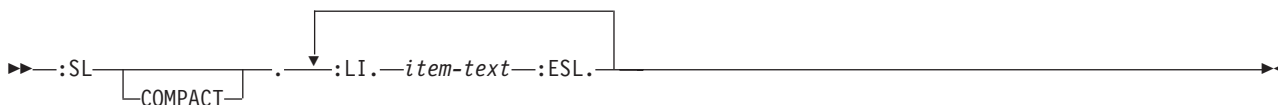
UIM ソース

```
:P.This is an example of some
:RT.PRETEND HEBREW TEXT:ERT.
imbedded within some real
English text.
```

結果

```
This is an example of some DNETERP
TXET WERBEH imbedded within some
real English text.
```

SL (単純リスト)



単純リスト (SL) タグは、項目のリストを識別します。これには、対応する終了タグが必要です。これらのタグは、情報域とヘルプ領域でのみ許可されます。

単純リストは、テキスト内の任意の場所に出現でき、他のリスト内でネストできます。

単純リスト内で不定形式行 (LINES)、図表 (FIG)、および例 (XMP) タグを使用するときには注意が必要です。なぜなら、1 行に収まらないテキストは次の行の列 1 に折り返されるからです。行および図表は現行左マージンから始まり、例は現行左マージンから 4 スペース字下げされます。ネストされたリストがフォーマットされると、現行左マージンは変わります。さまざまな位置 (リスト内を含む) に組み込まれた LINES、FIG、または XMP タグを含むヘルプ情報がある場合、外観は毎回同じとは限らないので、望ましくない結果になる可能性があります。

単純リストは、項目 ID を持たない、ぶら下げインデント・リストとしてフォーマットされます。

他のタグを SL タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 89. SL タグと ESL タグの間で許可されるタグ

タグ名	順序	使用法	ページ
FIG (図表)	1	B	577
LINES (不定形式行)	1	B	597

表 89. SL タグと ESL タグの間で許可されるタグ (続き)

タグ名	順序	使用法	ページ
XMP (例)	1	B	697
NT (注記)	1	B	644
P (段落)	1	B	648
PC (段落の連結)	1	B	658
LP (リストの一部)	1	B	633
OL (番号付きリスト)	1	B	645
SL (単純リスト)	1	B	678
UL (順不同リスト)	1	B	690
PARML (パラメーター・リスト)	1	B	656
DL (定義リスト)	1	B	575

オプションの属性

COMPACT

リストは、項目間にブランク行を入れずにフォーマット設定されます。

必須タグ

:LL.item-text

リスト項目のテキスト。テキストは、現行マージンから 4 スペース字下げされます。

例: 単純リスト

この例は、2 つの単純リストを使用します。一方は他方の内部に組み込まれています。2 番目の単純リストは COMPACT 属性を使用します。

UIM ソース

```
Some normal text...
:s1.
:li.First item
:s1.
:li.First item
:li.Second item
:es1.
:li.Second item
:es1.
```

結果

```
Some normal text...

First item

    First item
    Second item

Second item
```

TEXT (テキスト域)

```
▶▶:TEXT—VAR—=dialog-variable-name—USREXIT—='CALL program-reference'
▶
┌—ROW—=dialog-variable-name— —COL—=dialog-variable-name—┐
▶▶
```

テキスト域 (TEXT) タグは、アプリケーションによってフォーマットされるパネル内のテキスト域を定義します。各パネル内に指定できる TEXT タグは 1 つだけです。メニュー・バーおよびコマンド行は許可されますが、他の領域をパネル内で定義することはできません。

この領域の幅は、この領域を定義する PANEL タグの WIDTH 属性で指定します。

この領域の縦の長さは、パネルの縦の長さと同じです。パネルの縦の長さは、この領域を定義する PANEL タグの DEPTH 属性で指定します。DEPTH 値は、パネルの他の部分に使用される行 (メニュー・バーの行、パネル・タイトルの行、コマンド行、機能キーの行、およびメッセージ行など) の分だけ小さくなります。これらの大きさについては、PANEL タグを参照してください。

この領域に表示する目的で UIM に渡されるデータについては、以下の 682 ページの『テキスト・データ』で説明します。

カーソルがテキスト域にあるときにヘルプ・キーを押すと、パネルの全般ヘルプが表示されます。

必須属性

VAR=*dialog-variable-name*

テキスト域のデータ、またはデータへのポインターが含まれるダイアログ変数の名前。ダイアログ変数を定義するには、基本タイプに CHAR X または PTR を指定したクラス定義 (CLASS) タグを使用します。CHRID=PNLGRP 属性は、このダイアログ変数のクラス定義では許可されません。

UIM は、ダイアログ変数内のデータを $d \times w$ の文字配列と見なします。ここで、 d と w は、それぞれテキスト域の縦の長さおよび幅を表します。配列の各行は、テキスト域の 1 つの行として表示されます。

CLASS タグ上で *dialog-variable-name* の BASETYPE が CHAR の場合、このダイアログ変数にはテキスト域のデータが入ります。

CLASS タグ上で *dialog-variable-name* の BASETYPE が PTR の場合、このダイアログ変数にはテキスト域のデータへのポインターが入ります。UIM は、このポインターが指すデータを、BASETYPE が CHAR のダイアログ変数と同様に見なします。このデータへのアクセス中に UIM が例外を受け取ると、呼び出し側プログラムにエスケープ・メッセージが送信されます。

ダイアログ変数は、任意のサイズで宣言できます。データがテキスト域を満たすほどない場合は、テキスト域の残りの部分はブランクになります。データがテキスト域を超えるほど多い場合は、テキスト域を満たすのに必要なデータだけが使用されます。

テキスト域の出口プログラム (このタグの USREXIT 属性で指定する) を使用して、パネルが表示されるたびにこのダイアログ変数を更新できます。UIM は、682 ページの『テキスト・データ』で説明されているように、ダイアログ変数の値を使用して、テキスト域のデータを処理します。ダイアログ変数の値は、ダイアログ変数の CLASS タグ上の BASETYPE 属性によって決まります。

テキスト域のサイズは、機能キーがどれほどの行数を使用するかによって異なります。これは、複数の言語でコーディングする場合、または条件 (COND) タグで機能キーの条件を定義する場合に重要です。

USREXIT='CALL program-reference'

パネルが表示されるたびに、VAR 属性のダイアログ変数の値を更新するために呼び出される出口プログラム。

UIM は、この領域をスクロール可能と見なします。ユーザーがダイアログ・スクロール・コマンドに割り当てられたキーを押す場合で、テキスト域をスクロールする必要がある場合 (パネル上の他のスクロール可能域とは対照的に)、UIM はユーザー出口プログラムを呼び出します。UIM がパネルを再表示します。UIM によって渡されるユーザー出口プログラム構造体については、iSeries Information Center の『APIs』のトピックを参照してください。

ユーザーが過剰にスクロールした場合、汎用パネル出口プログラムを使用して診断する必要があります。ユーザーが過剰にスクロールした場合、汎用パネル出口は適切なメッセージと、それに続く決定済みの操作を取り消す特殊なメッセージを送信する必要があります。UIM によって渡される一般パネル出口プログラム構造体については、iSeries Information Center の『APIs』のトピックを参照してください。これが完了すると、スクロールし過ぎたことをユーザーに伝えるメッセージが 1 回だけ表示されます。テキスト域出口がメッセージを送信すると、メッセージは予想以上に長く画面に表示されることがあります。

CALL ダイアログ・コマンドの説明については、699 ページの『付録 B. UIM ダイアログ・コマンド』を参照してください。

オプションの属性

ROW=dialog-variable-name

カーソルの行。ダイアログ変数を定義するには、BASETYPE が BIN(31) のクラス定義 (CLASS) タグを使用します。

行および列のダイアログ変数は、UIM を呼び出してカーソルの位置をテキスト域内に設定する前にアプリケーションで設定できます。テキスト域の最初の行の先頭文字は、行 1 列 1 です。どちらかのダイアログ変数にゼロの値を指定すると、UIM にはデフォルトのカーソル位置を使用するよう通知されます。

UIM がアプリケーションに戻ると、これらのダイアログ変数は、ユーザーがパネルを終了したときにテキスト域内でカーソルが置かれていた行と列に設定されます。ユーザーがパネルを終了したときにカーソルがテキスト域内になかった場合は、行および列のダイアログ変数は両方ともゼロになります。

COL=dialog-variable-name

カーソルの列であることを除けば ROW と同じです。

カーソル位置

テキスト域のあるパネル内のカーソル位置は、多くの要因に左右されます。カーソルがどこに置かれるかについての優先順位を示したリストを次に示します。カーソルの位置は、最初に当てはまる項目にしたがって決定されます。

1. TEXT タグ上で ROW および COL 属性を使用する場合で、パネルが表示されるときにダイアログ変数の値が有効な場合、カーソルはその行と列に置かれます。
2. パネルにコマンド行がある場合、カーソルはコマンド行に置かれます。
3. パネルにメニュー・バーがある場合、カーソルは最初のメニュー・バー選択項目に置かれます。

TEXT タグ

4. テキスト域にタブ移動可能な強調表示クラスがある場合、カーソルは最初の強調表示クラスに置かれます。
5. 上記のいずれも当てはまらない場合、カーソルはパネルまたはポップアップ・ウィンドウの左上隅に置かれます。

テキスト・データ

データ内の任意の文字を強調表示クラスの選択文字にすることができます。テキスト域で使用できるのは出力データのみで、入力フィールドは使用できません。テキストは、クラス選択文字の後ろにある文字から、別のクラス選択文字または領域の最後 (いずれか先に登場する方) まで強調表示されます。

次の表は、UIM によって認識される強調表示の選択文字について説明しています。

16 進値	10 進値	カラー・ディスプレイ	モノクロ・ディスプレイ
X'01'	01		
X'02'	02	緑	通常
X'03'	03	青緑色*	通常
X'04'	04	赤	高輝度
X'05'	05	ピンク	通常
X'06'	06	黄色*	高輝度
X'07'	07	白	高輝度
X'11'	17	青の反転表示	通常の反転表示
X'12'	18	緑の反転表示	通常の反転表示
X'13'	19	青緑色の反転表示	通常の反転表示
X'14'	20	赤の反転表示	高輝度の反転表示
X'15'	21	ピンクの反転表示	通常反転表示
X'16'	22	黄色の反転表示	高輝度の反転表示
X'17'	23	白の反転表示	高輝度の反転表示
X'19'	25	青の下線	通常の下線
X'1A'	26	緑の下線	通常の下線
X'1B'	27	青緑色の下線 *	通常の下線
X'1C'	28	赤の下線	高輝度の下線
X'1D'	29	ピンクの下線	通常の下線
X'1E'	30	黄色の下線 *	高輝度の下線
X'1F'	31	白の下線	高輝度の下線
X'21'	33	青のタブ	通常のタブ
X'22'	34	緑のタブ	通常のタブ
X'23'	35	青緑色のタブ *	通常のタブ
X'24'	36	赤のタブ	高輝度のタブ
X'25'	37	ピンクのタブ	通常のタブ
X'26'	38	黄色のタブ *	高輝度のタブ
X'27'	39	白のタブ	高輝度のタブ
X'31'	49	青の反転表示タブ	正常の反転表示タブ
X'32'	50	緑の反転表示タブ	正常の反転表示タブ
X'33'	51	青緑色の反転表示タブ	正常の反転表示タブ
X'34'	52	赤の反転表示タブ	高輝度の反転表示タブ
X'35'	53	ピンクの反転表示タブ	正常の反転表示タブ
X'36'	54	黄色の反転表示タブ	高輝度の反転表示タブ
X'37'	55	白の反転表示タブ	高輝度の反転表示タブ
X'39'	57	青の下線タブ	通常の下線タブ
X'3A'	58	緑の下線タブ	通常の下線タブ
X'3B'	59	青緑色の下線タブ *	通常の下線タブ
X'3C'	60	赤の下線タブ	高輝度の下線タブ
X'3D'	61	ピンクの下線タブ	通常の下線タブ
X'3E'	62	黄色の下線タブ *	高輝度の下線タブ
X'3F'	63	白の下線タブ	高輝度の下線タブ

* 一部のディスプレイでは、この強調表示クラスは列区切り文字を表示し、ブランク文字は画面上で小さなボックスのように見えます。

RV3W081-0

図 146. テキスト域で使用可能な強調表示クラス

TEXT タグ

UIM は、それぞれのクラス選択文字をクラスの適切な表示属性に置き換えます。X'10' から X'3F' の範囲の他の文字値と X'FF' は X'1F' に変換され、反転表示ボックスとして画面に表示されます。文字 X'00' (ヌル)、X'0E' (2 バイトのシフトアウト)、X'0F' (2 バイトのシフトイン)、および X'40' から X'FE' (通常表示可能文字) は、未変更のまま画面に渡されます。

最適な外観を得るために、パネルの最初の列と最後の列にテキストを置くべきではありません。これによって、どの行においても最初の語を簡単に強調表示できます。

特定の色でタブ移動可能な句を作成する場合は、句を構成する語と語の間に空白を使用しないでください。タブ移動非対応の同じ色の強調表示クラスを使用する必要があります。これにより、すべてのワークステーションで最も効果的に表示を行うことができます。

テキスト域にタブ移動可能な強調表示クラスがあるパネルが表示されても、カーソルは最初の強調表示クラスに自動的に置かれません。これを行うには、このタグの ROW および COL 属性を使用してカーソルの位置を設定する必要があります。

1 つ以上のタブ移動可能な強調表示クラスをデータ内で隣同士に配置すると、最後の強調表示クラスだけがタブ移動可能となります。他のクラスは無視されます。タブ移動可能なフィールドの最大数は 240 です。ディスプレイ上に他の入力可能フィールドまたはタブ移動可能フィールド (コマンド行やメニュー・バー選択項目など) が存在する場合、最大数はこれより少なくなる可能性があります。

全画面表示パネルの最終行の最終列、またはポップアップ・ウィンドウの任意の行の最終列に強調表示クラスを配置しても、そのクラスは無視されます。

ポップアップ・ウィンドウの場合、強調表示クラスは次の行に続きません。次の行に続ける場合は、別の強調表示クラスを使用する必要があります。

最初にポップアップ・ウィンドウを追加しないで、装置より小さいパネルが表示されると、強調表示クラスは、画面上のパネルの右側にある空白・スペースに続きます。これらの強調表示クラスをオフにするには、各行の最後で外観が空白・スペースに似た他の強調表示クラスを使用します。

テキスト域データに対して文字セットおよびコード・ページ変換は行われません。アプリケーションは、装置に対して正しい文字セットおよびコード・ページでデータを提供する必要があります。

UIM が、テキスト域をフォーマットするために出口プログラムを呼び出すと、パネル・グループ (PNLGRP) タグの BIDI 属性、および表示装置のコード・ページ番号を識別する値が渡されます。

CLASS タグ上でテキスト域のダイアログ変数に NBRSHAPE および SYMSWAP 属性を使用しても、これらの属性は無視されます。

DBCS データを使用する場合、データが正常に表示されるようにアプリケーションが処理を行います。DBCS 文字の半分を表示しようとする場合、または IGC シフト文字 (X'0E'、X'0F') のいずれかが欠落している場合は、エラーが発生します。

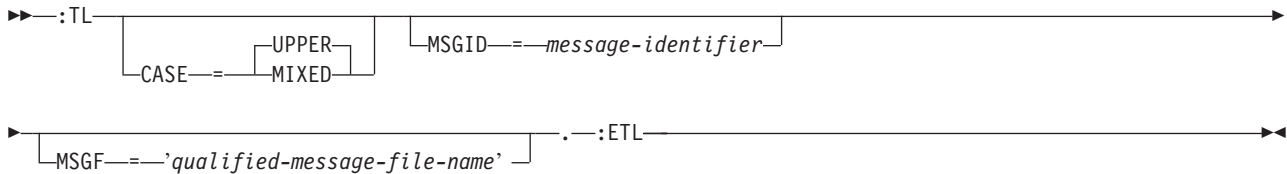
TI タグ

オプションのテキスト

displayed-value

パネルに表示される値、またはユーザーが入力する値。ユーザーが指定する外部値にどの内部値を使用するかを決定するために UIM が行う比較は、CLASS タグの CASE および BLANKS 属性で変更できます。リスト内で同じ外部値が 2 回以上出現する場合、入力時に最初の項目が変換に使用されます。値を指定しない場合は、ブランクの表示値が使用されます。

TL (変換リスト)



変換リスト (TL) タグを使用すると、変数プールがディスプレイに実際に入力される値による影響を受けないようにして、クラスの変換リストを指定することができます。BASETYPE が CHAR、IGC、DATE、TIME、NAME、OBJNAME の値、またはクラス定義 (CLASS) タグ上の数値で、変換リストを使用できます。各クラスで使用できる変換リストは 1 つだけです。変換は、妥当性検査が実行される前に行われます。

他のタグを TL および ETL タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、各タグについて詳しく説明しているページを示しています。

表 90. TL タグと ETL タグの間で許可されるタグ

タグ名	順序	ページ
TI (変換リスト項目)	1	685

オプションの属性

CASE=UPPER | MIXED

UPPER は、入力フィールド値の小文字 (a から z) が、変換リスト項目ごとに外部値と比較される前に、大文字 (A から Z) に変換されることを指示します。大文字への変換は、CLASS タグ上の CASE 属性の変換後に行われますが、一致する変換リスト項目が見つからない場合は、ダイアログ変数またはリスト項目に割り当てられた値に対して影響を与えることはありません。各変換リスト項目 (TI) タグで指定された外部ディスプレイ値も変更されません。TI タグ上のテキストは、大文字で指定しなければ一致しません。

MIXED は、入力値を入力したとおりに各 TI タグで指定された外部値と比較することを指示します。

MSGID=message-identifier

リストに指定されていない値をユーザーが入力したために変換が失敗した場合に送信されるメッセージのメッセージ ID。この属性を指定しない場合、UIM は入力された値を入力どおり有効と見なし、その値をクラスの妥当性検査に渡します。

MSGF='qualified-message-file-name'

変換が失敗する場合に、このエラーで送信されるメッセージが入っているメッセージ・ファイル。パネ

ル・グループ (PNLGRP) タグに DFTMSGF 属性が指定されていない場合で、このタグに MSGID 属性が指定されている場合は、この属性を指定する必要があります。

例: 変換リスト

この例は、変換リストがどのように定義されるかを示しています。

UIM ソース

```
:tl msgid=MMM0001 msgf='*LIBL/QCPFMSG'.
:ti value=1.Blue
:ti value=1.BLUE
:ti value=2.Red
:ti value=2.RED
:ti value=3.Green
:ti value=3.GREEN
:ti value=4.Turquoise
:ti value=4.TURQUOISE
:etl.
```

入力時に *Blue* が値 1 にマップされます (デフォルトの CASE=UPPER、および変換項目の大文字バージョンが指定されているため)。出力時に、1 が値 *Blue* (大/小文字混合) にマップされます。常に、一致する最初の変換リスト項目が使用されるためです。値に整数表記が使用されるため、この変換リストが適用される CLASS タグ上の BASETYPE は数値でなければなりません。

次に示すのは、有効な i5/OS ライブラリー名 (*LIBL、*CURLIB、またはブランク) を使用できるダイアログ変数のクラス定義および変換リストの例です。

UIM ソース

```
:class name=objlibc
      basetype='objname 10'.
:tl.
:ti value='*LIBL'.*LIBL
:ti value='*CURLIB'.*CURLIB
:ti value=""
:etl.
:eclass.
```

TOPINST (最上部指示)

```
▶▶:TOPINST [INST=—dialog-variable-name] . [instruction-text] ▶▶
```

領域の最上部指示 (TOPINST) タグは、最上部指示行を指定します。このタグはすべてに領域で有効ですが、表示パネルでのみ使用できます。これは、領域の導入タグの直後に出現します。

TOPINST タグに *instruction-text* が含まれている場合は、複数の最上部指示タグを使用して、複数の指示行を表示できます。複数の TOPINST タグをコーディングした場合、異なるタグのテキストの間にブランク行は表示されません。このタグで INST 属性変数が使用されている場合、領域ごとに使用可能な最上部指示タグは 1 つだけです。

メニューの場合、本体と指示行域の間に常にブランク行が残されます。特定の領域の本体に対して指示行フォーマットされる方法については、544 ページの『DATA (データ表示域)』の BODYSEP 属性を参照してください。

TOPINST タグ

領域に TOPINST タグを指定しない場合、指示行にテキストは使用されないため、領域内に行は割り振られません。

オプションの属性

INST=*dialog-variable-name*

表示する最上部指示テキストを含むダイアログ変数の名前。ダイアログ変数は、パネル・タグで指定する幅から 2 を引いた幅以下の幅で定義しなければなりません。この属性が使用される場合、このタグに *instruction-text* は指定できません。

ダイアログ変数は、クラス定義 (CLASS) タグで BASETYPE に CHAR、IGC、または BIN を指定して定義する必要があります。

ダイアログ変数のエラー状態は、テキストの強調表示の判別には使用されません。

IGC 用の特殊フォーマット。 省略形 IGC は、コマンドおよびキーワードで使用され、2 バイト文字セット (DBCS) 機能を表します。BASETYPE が IGC であるダイアログ変数を CLASS タグ上で指定すると、UIM は特殊フォーマットを行います。変数値がシフトアウト文字 (X'0E') で始まる場合は、UIM は値を左に 1 バイト移動して、他の行と垂直位置合わせを行います。

オプションのテキスト

instruction-text

領域の最上部指示として表示されるテキスト。このテキストは暗黙の段落です。ディスプレイのフォーマット時に、1 表示行に収まらないテキストは、必要に応じて 2 行目以降に 2 桁下げしてフォーマットされます。テキストは最大 255 文字で、テキスト反転 (RT) タグだけを含めることができます。このタグに INST 属性を指定しない限り、*instruction text* は必須です。

TT (真理値表)

▶▶:TT—NAME—=*truth-table-name*—CONDS—=*'condition-name-list'*—.—:ETT.—▶▶

真理値表 (TT) タグは、パネル定義の処理中にコンパイラーが使用する真理値表の定義を開始します。TT タグは、パネル・グループのプロローグ・セクション内で、条件定義 (COND) タグとメニュー・バー (MBAR) タグの間に出現する必要があります。真理値表を使用して、どの条件が相互に排他的であるかを UIM に通知する必要があります。

パネル・グループのコンパイル中に、パネルは真理値表の 1 行ごとに 1 回フォーマットされます。各条件の真理値を使用して、コンパイラーはパネル定義が有効かどうかを検査します。このタグの COND 属性を使用してパネル・エレメントの条件を定義する場合で、その条件が真理値表にない場合、コンパイラーは条件の値を真と見なします。

コンパイラーが真理値表を使用してパネル検査を行えるようにするには、真理値表の名前を、表示パネル (PANEL)、印刷パネル (PRTPNL)、または印刷ヘッド・パネル (PRTHEAD) タグの TT 属性に指定する必要があります。

コンパイル後、真理値表の誤りのために、領域内に表示しきれないほど多くのパネル・エレメントが条件定義によってアクティブにされる場合は、領域内に収まるエレメントのみが表示され、例外は報告されません。

他のタグを TT タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、各タグについて詳しく説明しているページを示しています。

表 91. TT タグと ETT タグの間で許可されるタグ

タグ名	順序	ページ
TTROW (真理値表行)	1	689

必須属性

NAME='truth-table-name'

真理値表の名前。命名規則について詳しくは、515 ページの『名前構文』を参照してください。この名前は、この真理値表をコンパイル時検査に使用するすべてのパネルによって参照されます。この名前は、パネル・グループ内で固有でなければなりません。

CONDS='condition-name-list'.

ブランクで分離された条件名のリスト。リスト内の各条件は、COND タグで定義済みでなければなりません。真理値表の行は、真理値表行 (TTROW) タグで定義します。TTROW タグの VALUES 属性内の各項目は、この属性で指定する条件と位置的に対応します。最大 50 の条件名を指定できます。

例: 真理値表

この例は、条件 A、B、および C がパネル X 上で相互に排他的な場合、これをコンパイラーに通知する方法を示しています。

UIM ソース

```
:tt name=tt1 conds='A B C'.
:ttrow values='1 0 0'.
:ttrow values='0 1 0'.
:ttrow values='0 0 1'.
:ett.
```

TTROW (真理値表行)

▶▶:TTROW—VALUES=—'—condition-value-list—'—◀◀

真理値表行 (TTROW) タグは、パネル定義の処理中にコンパイラーが使用する真理値表の行を定義します。TTROW タグは、パネル・グループのプロローグ・セクションの真理値表 (TT) タグと ETT タグの間に出現する必要があります。

パネル・グループのコンパイル中に、パネルは真理値表の 1 行ごとに 1 回フォーマットされ、各条件の真理値を使用してパネル定義が有効かどうか検査されます。真理値表では、パネルのフォーマットを制御する条件のうち、相互に排他的な条件を記述できます。パネル定義内で条件を使用する場合、真理値表にその条件がない場合、コンパイル時検査を行う目的でその条件は真と見なされます。これは、このタグの VALUES 属性にアスタリスク (*) を指定することと同じです。

必須属性

VALUES='condition-value-list'.

コンパイラーがパネル定義を検査するために使用する各条件の値。TTROW タグの VALUES 属性内の各項目は、この属性で指定する条件と位置的に対応します。条件に使用できる値は、次のとおりです。

TTROW タグ

- 0 条件は偽です。
- 1 条件は真です。
- * 真理値は指定されていません。ワーストケースの場合にコンパイラーは、値を真と見なします。

条件値のリスト内の項目は、ブランクで分離されます。リスト内の項目数は、TT タグの CONDS 属性の項目数と同じでなければなりません。

UL (順不同リスト)



順不同リスト (UL) タグは、項目の順不同リストを示します。UL タグには、対応する終了タグが必要です。これらのタグは、情報域とヘルプ領域でのみ許可されます。

順不同リストは、テキスト内の任意の場所に出現でき、他のリスト内でネストできます。

注: 順不同リスト内で不定形式行 (LINES)、図表 (FIG)、および例 (XMP) タグを使用するときには注意が必要です。なぜなら、1 行に収まらないテキストは次の行の列 1 に折り返されるからです。行および図表は現行左マージンから始まり、例は現行左マージンから 4 スペース字下げされます。ネストされたリストがフォーマットされると、現行左マージンは変わります。ヘルプ情報のさまざまな場所 (リスト内を含む) に LINES、FIG、または XMP タグが埋め込まれている場合、ヘルプ情報は表示されるたびに外観が異なる場合があるため、望ましくない結果が生じる可能性があります。

順不同リストはぶら下げインデント・リストとしてフォーマットされ、項目 ID が現行左マージンに表示されます。ネストされた順不同リストの項目 ID には、次の 3 つのレベルがあります。

- 中黒 (小文字の o)
- ハイフン (-)
- ダッシュ (--)

ネストされた順不同リストでは、これらのレベルは 3 レベルを超えて繰り返されます。

他のタグを UL タグ内にネストすることができます。それらのタグを以下の表にリストします。この表は、タグの出現順序を定義し、表示パネルでのみ、印刷パネルでのみ、あるいはその両方で使用できるタグを示し (それぞれ D、P、または B で表される)、各タグについて詳しく説明しているページを示しています。

複数のタグが同じ順序番号でリストされている場合、その番号のすべてのタグは任意の順序で混在させることができます。ただし、順序番号が大きいタグを順序番号が小さいタグより先に使用することはできません。例えば、順序番号が 3 のタグは、順序番号が 1 または 2 のタグより先に使用することはできません。

表 92. UL タグと EUL タグの間で許可されるタグ

タグ名	順序	使用法	ページ
FIG (図表)	1	B	577

表 92. UL タグと EUL タグの間で許可されるタグ (続き)

タグ名	順序	使用法	ページ
LINES (不定形式行)	1	B	597
XMP (例)	1	B	697
NT (注記)	1	B	644
P (段落)	1	B	648
PC (段落の連結)	1	B	658
LP (リストの一部)	1	B	633
OL (番号付きリスト)	1	B	645
SL (単純リスト)	1	B	678
UL (順不同リスト)	1	B	690
PARML (パラメーター・リスト)	1	B	656
DL (定義リスト)	1	B	575

オプションの属性

COMPACT

リストは、項目間に空白行を入れずにフォーマットされます。

必須タグ

:LL.item-text

リスト項目のテキスト。このテキストの先頭に項目 ID があります。テキストは、現行マージンから 4 スペース字下げされます。

例: 順不同リスト

この例では 2 つの順不同リストが使用されていて、一方のリストが他方のリストに埋め込まれています。2 番目の順不同リストでは、COMPACT 属性が使用されています。

UIM ソース

```
Some normal text...
:ul.
:li.First item (bullet)
:ul.
:li.First item (hyphen)
:li.Second item (hyphen)
:eul.
:li.Second item (bullet)
:eul.
```

結果

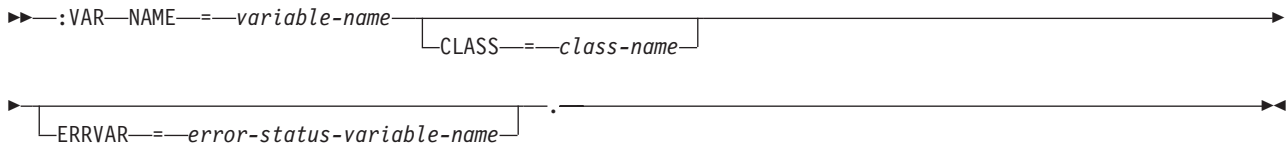
```
Some normal text...

• First item (bullet)

- First item (hyphen)
- Second item (hyphen)

• Second item (bullet)
```

VAR (変数定義)



変数定義 (VAR) タグは、パネル・グループ内でダイアログ変数を宣言します。VAR タグでは、タグをネストできません。また、このタグは、最後のクラス定義 (CLASS) タグと最初の変数レコード定義 (VARRCD) タグの間で使用する必要があります。

各ダイアログ変数の値は、このパネル・グループを使用して開かれているアプリケーションの変数プール内にあります。ダイアログ変数は、ユーザーにパネルが表示されるときに、パネルとの間で値をやり取りします。ダイアログ変数値の設定および検索は、UIM アプリケーション・プログラミング・インターフェース (API) を使用して行われます。

VAR タグは、すべての Z 変数と、アプリケーションで使用される他のダイアログ変数を定義します。開いているアプリケーションで使用できるのは、パネル・グループ・ソース内で VAR タグによって定義される変数だけです。

必須属性

NAME=variable-name

ダイアログ変数の名前。この名前はパネル・グループ内で固有でなければなりません。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

オプションの属性

CLASS=class-name

この変数が属するクラスの名前。クラスは、パネル・グループ内で事前に定義しておく必要があります。

すべての Z 変数のクラス定義は UIM によって決定されるため、この属性を Z 変数の VAR タグに指定することはできません。パネル・グループ内で宣言される他の変数にクラスを指定する必要があります。

ERRVAR=error-status-variable-name

CLASS タグ上で BASETYPE に CHAR 1 を指定して UIM によって自動的に定義されるダイアログ変数の名前。エラー状況変数を別の VAR タグで定義してはなりません。また、この名前はパネル・グループ内で固有でなければなりません。

このダイアログ変数には、このタグで定義される変数のエラー状況が入ります。この属性は、Put Dialog Variable (QUIPUTV) API および Get Dialog Variable (QUIGETV) API で使用されます。

QUIPUTV API の要求中に、エラー状況はエラー変数の現行値によって決定されます。値 1 (X'F1') は、このタグの NAME 属性で指定したダイアログ変数にエラーがあることを示します。1 (X'F1') 以外の値は、ダイアログ変数にエラーがないことを示します。エラー状況変数が、関連するダイアログ変数と同じ QUIPUTV API 要求中に更新される場合は、エラー状況変数の新しい値が使用されます。

NAME ダイアログ変数または *ERRVAR* ダイアログ変数の QUIGETV API 要求中に、このタグの NAME 属性で指定したダイアログ変数にエラーがある場合は、エラー状況変数の値が 1 (X'F1') に設定されます。エラーがない場合は、エラー状況変数の値はゼロ (X'F0') に設定されます。

注: *ERRVAR* ダイアログ変数を更新するだけでは、このタグの VAR 属性に指定したダイアログ変数のエラー状態は設定されません。エラー状態を変更するには、VAR ダイアログ変数を QUIPUTV API を使用して更新する必要があります。

エラー・ダイアログ変数は、通常ダイアログ変数を指定できるパネル・グループ・ソース内の任意の場所から参照できません。

UIM によって定義されるダイアログ変数

UIM は、開いているすべてのアプリケーションの変数プール内で自動的に使用可能にできるダイアログ変数のセットを定義します。変数プール内で使用できるようにするには、変数を変数定義 (VAR) タグを使用して定義する必要があります。これらの変数は、ユーザー・プログラムによって (QUIGETV API を参照)、またパネル定義内で参照され、一部は UIM 処理機能を制御する目的でユーザー・プログラムによって変更できます (QUIPUTV API を参照)。どのパネルでも、入力フィールドではダイアログ変数が許可されません。

UIM によって定義されるすべてのダイアログ変数の名前は、文字 Z で始まります。アプリケーションのダイアログ変数を UIM ダイアログ変数と同じ名前で作成すると、警告メッセージが表示されます。これは、同じ名前の UIM ダイアログ変数にアクセスできないことを告げるメッセージです。ただし、自分で定義したダイアログ変数を使用することは可能です。次に、UIM によって定義されるすべてのダイアログ変数について説明します。

ZCANCEL CANCEL 機能が要求されているかどうかを示し、以下のように設定されます。

- 0 CANCEL 機能は要求されていません。
- 1 CANCEL 機能は要求されています。

この Z 変数は、QUIPUTV API を使用して、呼び出し側 (アプリケーション) プログラムによって変更できます。このダイアログ変数を変更すると、ジョブの取り消しフラグが変更されます。取り消しフラグの詳細は、388 ページの『リスト・パネルをまとめて閉じる』を参照してください。

ZDBCS ジョブが現在 DBCS モードになっているかどうかを示し、以下のように設定されます。

- 0 ジョブは DBCS モードになっていません。
- 1 ジョブは DBCS モードになっています。

ZDPSIZ 装置で表示可能なディスプレイの最大サイズを示し、以下のように設定されます。

- *DS3 装置で表示可能なパネルの最大サイズは 24 x 80 です。
- *DS4 装置で表示可能なパネルの最大サイズは 27 x 132 です。この装置は、サイズが 24 x 80 のパネルを表示することも可能です。

ZEXIT EXIT 機能が要求されているかどうかを示し、以下のように設定されます。

- 0 EXIT 機能は要求されていません。
- 1 EXIT 機能は要求されています。

この Z 変数は、Put Dialog Variable API サービスを使用して、呼び出し側 (アプリケーション) プログラムによって変更できます。このダイアログ変数を変更すると、ジョブの出

VAR タグ

ロフラグが変更されます。ジョブ出口フラグの詳細については、387 ページの『EXIT の要求時に複数のパネルをまとめて閉じる』を参照してください。

ZJOB	現行ジョブの単純名 (ジョブ名の最初の修飾子)。
ZJOBNBR	現行ジョブのジョブ番号 (ジョブ名の最後の修飾子)。
ZLMTCPB	ユーザーが、初期プログラム、初期メニュー、現行ライブラリー、およびアテンション・キー処理プログラムの値をどこまで制御できるかを示し、以下のように設定されます。 *NO *PARTIAL *YES
ZMENU	UIM によって現在表示されているメニュー・オブジェクトの名前。このダイアログ変数は、*MENU オブジェクトの定義でのみ使用できます。
ZMNULIB	UIM によって現在表示されているメニュー・オブジェクトが入っているライブラリーの名前。このダイアログ変数は、*MENU オブジェクトの定義でのみ使用できます。
ZSYSNAM	SYSNAM ネットワーク属性から検索される現行システム名。
ZUSER	現行ジョブのユーザー・プロファイル名 (これはジョブ名の中央の修飾子です)。
Z36ENV	ジョブが現在 System/36 環境で実行されているかどうかを示し、以下のように設定されます。 0 ジョブは System/36 環境で実行されていません。 1 ジョブは System/36 環境で実行されています。

表 93. UIM によって定義される変数 (Z 変数) の属性

名前	基本タイプ	長さ	プログラムによる変更の可否	説明
ZCANCEL	CHAR	1	はい	CANCEL 機能が要求されているかどうかを示します
ZDBCS	CHAR	1	いいえ	ジョブ DBCS モードになっているかどうかを示します
ZDPSIZ	CHAR	4	いいえ	ディスプレイの最大サイズを示します
ZEXIT	CHAR	1	はい	EXIT 機能が要求されているかどうかを示します
ZJOB	OBJNAME	10	いいえ	現行ジョブのジョブ名 (最初の修飾子)
ZJOBNBR	CHAR	6	いいえ	現行ジョブのジョブ番号
ZLMTCPB	CHAR	10	いいえ	ユーザーの能力を制限します
ZMANMNU	CHAR	1	いいえ	必須メニュー・ジョブ・フラグ
ZMENU	OBJNAME	10	いいえ	メニュー・オブジェクトの名前
ZMNULIB	OBJNAME	10	いいえ	メニュー・オブジェクトが入っているライブラリーの名前
ZSYSNAM	CHAR	8	いいえ	現行システム名
ZUSER	OBJNAME	10	いいえ	ユーザー・プロファイル名
Z36ENV	CHAR	1	いいえ	System/36 環境で実行されているかどうかを示します

VARRCD (変数レコード定義)

```
▶▶:VARRCD—NAME—='variable-record-name'—VARS—='variable-list'—▶▶
▶▶┌──NOPUT—='variable-list'──┐┌──NOGET—='variable-list'──┐▶▶
```

変数レコード定義 (VARRCD) タグでは、タグをネストできません。また、このタグは、最後の変数定義 (VAR) タグと最初のリスト定義 (LISTDEF) タグの間で使用する必要があります。

呼び出し側プログラムと変数プールの間でデータの受け渡しを行う場合、変数バッファが使用されます。変数バッファには、1 つ以上のダイアログ変数の値が入ります。VARRCD タグは、変数バッファに入るダイアログ変数と、変数バッファ内での変数の順序を定義します。

Get Dialog Variable (QUIGETV) API 要求および Put Dialog Variable (QUIPUTV) API 要求で使用する変数レコード定義を定義するには、VARRCD タグを使用する必要があります。変数レコード定義で使用できるのは、パネル・グループ・ソース内で VAR タグによって定義される変数だけです。

必須属性

NAME=*variable-record-name*

変数レコード定義の名前。この名前はパネル・グループ内で固有でなければなりません。命名規則について詳しくは、515 ページの『名前構文』を参照してください。

VARS=*'variable-list'*

変数レコードを構成する最大 256 個のダイアログ変数のリスト。この変数リスト内のすべてのダイアログ変数は、VAR タグを使用して事前定義しておく必要があります。リスト内のダイアログ変数名は、1 つ以上のブランクで区切られます。

QUIPUTV および QUIGETV API の呼び出し時の呼び出し側プログラムの変数バッファの順序は、変数リスト内の変数名の順序でなければなりません。

変数は、呼び出し側プログラムの変数バッファ内に連続して現れます。UIM は、バッファ内の変数へのオフセットを計算する際に境界合わせ調整を行いません。

このタグの VARS 属性に読み取り専用 Z 変数がリストされている場合、このタグの NOPUT 属性にもこれらの変数を指定する必要があります。

オプションの属性

NOPUT=*'variable-list'*

QUIPUTV API の要求時に処理してはならない最大 256 個の変数のリスト。この変数リスト内のすべての変数は、このタグの VARS 属性で指定する変数リストに指定しておく必要があります。リスト内の変数名はブランクで区切られます。

この属性を使用することによって、QUIPUTV API 要求中に変数バッファ内の変数のサブセットだけを変数プールにコピーすることを指定できます。

この属性を指定しない場合は、QUIPUTV API 要求中にこのタグの VARS 属性で指定するすべての変数が、変数バッファから変数プールにコピーされます。

VARRCD タグ

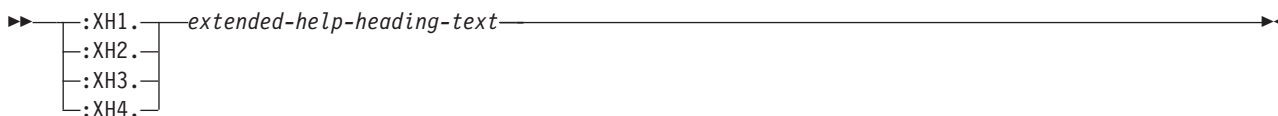
NOGET='variable-list'

QUIGETV API の要求時に処理してはならない最大 256 個の変数のリスト。この変数リスト内のすべての変数は、このタグの VARS 属性で指定する変数リストに指定しておく必要があります。リスト内の変数名はブランクで区切られます。

この属性を使用することによって、QUIGETV API 要求中に変数バッファ内の変数のサブセットだけを変数プールからコピーすることを指定できます。

この属性を指定しない場合は、QUIGETV API 要求中にこのタグの VARS 属性で指定するすべての変数が、変数プールから変数バッファにコピーされます。

XH1 から XH4 (拡張ヘルプの見出し)



拡張ヘルプの見出し (XH1-XH4) タグは、拡張パネル・ヘルプの一部として表示されるときに (索引検索トピックのヘルプを含む)、文脈ヘルプ上で使用される見出しを識別します。このタグはヘルプ・モジュール (HELP) タグと EHELP の境界内で表示される必要があります、情報 (INFO) タグでは許可されません。このタグに関連したテキストは、見出しテキストとして使用されます。

このタグにより、テキストが拡張ヘルプ・パネル用に表示されるか、文脈ヘルプ用に表示されるかに基づいて、見出しをヘルプ情報とともに表示することができます。XH n タグのテキストは常に拡張パネル・ヘルプで表示されます。また、テキストは文脈ヘルプでも表示されます。それは、ヘルプ情報を含むタグが XH n タグの前に付いているか、または文脈ヘルプがフルスクリーンで表示され、ヘルプ・パネルのタイトルが HELP タグで指定されていない場合です。

文脈ヘルプのどの場合にも (ただし、前述の 2 つの場合を除いて)、XH n タグが文脈ヘルプ用に使用される最初のタグである場合には、XH n タグのテキストは表示されません。

必須テキスト

extended-help-heading-text

このタグに関連したテキストは、ヘルプ項目が拡張パネル・ヘルプで使用されるときに見出しとして使用されます。テキストは、タグを終了するピリオドと同じ行または次の行に表示される必要があります。

フォーマットの規則

見出しでは、前後に 1 行のブランク行がフォーマット設定されています。見出しタグは見出しの前にブランク行を 1 行生成し、見出しの後のタグは見出しに後にスペースを生成します。

個々のフォーマット規則は以下のとおりです。

XH1 テキストを中央揃えし、下線を付けて、強調表示します (HP3)。このタグを設定すると、印刷されたヘルプ・モジュールで表示されるときにページ排出が行われます。

XH2 テキストを左揃えし、下線を付けて、強調表示します (HP3)。

XH3 テキストを左揃えして、強調表示します (HP2)。

XH4 テキストを左揃えして、下線を付けます (HP1)。

例: サンプル見出し

この例は、拡張ヘルプの見出しが強調表示され、位置調整される様子を示しています。

UIM ソース

```
:XH1.Main Subject
:p.Here's a paragraph.
:XH2.Topic
:p.Another paragraph.
:XH3.Subtopic
:p.Still another paragraph.
:XH4.A Four Heading
:p.Still another paragraph.
```

結果

Main Subject

Here's a paragraph.

Topic

Another paragraph.

Subtopic

Still another paragraph.

A Four Heading

Still another paragraph.

XMP (例)

▶▶:XMP.—:EXMP.◀◀

例 (XMP) タグは、コンピューターの入力または出力の例を識別します。XMP タグには、対応する終了タグが必要です。これらのタグは、情報域とヘルプ領域でのみ許可されます。

例は他の例の中以外であれば、テキスト内の任意の場所に挿入することができます。例の本体は、XMP タグと EXMP タグに挟まれたテキストで構成されます。

テキストの通常のフォーマットは例の内部では中断されます。つまり、行は連結されません。

注: リスト内の例を使用するときには注意が必要です。なぜなら、1 行に収まらない例テキストは次の行の列 1 に折り返されるからです。ネストされたリストがフォーマットされると、現行左マージンは変わります。さまざまな位置 (リスト内を含む) に組み込まれた例を含むヘルプ情報がある場合、外観は毎回同じとは限らないので、望ましくない結果になる可能性があります。例は、現行マージンから 4 スペース字下げされます。フルスクリーン・フォーマットまたは拡張ヘルプ・ウィンドウで表示されるヘルプ・モジュールは、4 バイト字下げされます。そのため、最初のフォーマット幅は、ヘルプ・モジュ

XMP タグ

ールで指定された幅より 10 バイト小さくなります。 XMP タグが情報域に表示される場合、最初のフォーマット幅は、パネル・タグで指定された幅より 6 バイト小さくなります。

例: 例のフォーマット設定

この例は、例の使用方法を示します。

UIM ソース

```
Some normal text...
:xmp.
This is an example
right here
:exp.
```

結果

```
Some normal text...
This is an example
right here
```

付録 B. UIM ダイアログ・コマンド

ダイアログ・コマンドは UIM によってのみ認識される特殊関数で、ユーザーが入力したアクションと画面管理機能を等価にします。ダイアログ・コマンドは、コマンド行上に入力できず、UIM のスコープ外では無効であるという点で CL コマンドとは異なります。

ダイアログ・コマンドごとに UIM が厳密にどのような機能を実行するのかは、表示されている画面のタイプ、および機能キー、キー・リスト項目 (KEYI)、メニュー項目 (MENU)、リスト・アクション (LISTACT)、またはプルダウン・フィールド選択項目 (PDFLDC) 言語タグで定義されたその他の属性に大きく依存します。言語タグについては、511 ページの『付録 A. UIM パネル・グループ定義言語』を参照してください。

ダイアログ・コマンドの中には、現在のコマンド行の内容、カーソルの位置、および画面のタイプによって定義されるものがあります。さらにダイアログ・コマンドは、アプリケーション、条件評価、および一般出口の呼び出しによって指定されるメッセージに影響を与えます。たとえば、ダイアログ・コマンドの中のあるものは条件評価を迂回させ、他のあるものは一般出口の呼び出しを迂回させます。

ダイアログ・コマンドは、以下のものに割り当てられます。

- メニュー項目
- 機能キー
- 処置リストのオプション
- ハイパーテキスト・リンク
- プルダウン・フィールド選択項目
- PANEL タグの ENTER および SELECT 属性

すべてのダイアログ・コマンドがこれらすべての状態で許可されるわけではありません。たとえば、ダイアログ・コマンド PAGEUP および PAGEDOWN は画面全体に適用され、個々のリスト項目には適用されないため、それらのダイアログ・コマンドは処置リストのオプションに割り当ててはできません。

ダイアログ・コマンドの有効な使用法の要約については、表 94 を参照してください。さまざまなダイアログ・コマンドの影響の要約については、700 ページの表 95 を参照してください。それぞれのダイアログ・コマンドの詳細記述がこの付録で後述されています。

表 94. ダイアログ・コマンドの有効な使用法の要約

ダイアログ・コマンド名	有効な使用法						
	機能キー	プルダウン・メニュー選択項目	メニュー項目	リスト・アクション	PANEL タグ上の ENTER	PANEL タグ上の SELECT	ハイパーテキスト・リンク・アクション
ACTIONS	○					○ ³	
CALL	○	○	○	○	○		
CANCEL	○		○				
CHGVIEW	○	○ ²					
CMD	○	○	○	○	○		
CMDLINE	○	○ ²					

ダイアログ・コマンド

表 94. ダイアログ・コマンドの有効な使用法の要約 (続き)

ダイアログ・コマンド名	有効な使用法						
	機能キー	プルダウン・メニュー選択項目	メニュー項目	リスト・アクション	PANEL タグ上の ENTER	PANEL タグ上の SELECT	ハイパーテキスト・リンク・アクション
DSPHELP		○ ²					○
ENTER	○						
EXIT	○	○ ²	○				
EXTHelp		○ ²					
HELP	○						
HELPHelp		○ ²					
HELPIIDX		○ ²					
HOME	○						
KEYSHelp		○ ²					
MENU	○	○ ²	○				
MOREKEYS	○						
MOVETOP	○						
MSG					○	○	
PAGEDOWN	○						
PAGEUP	○						
PRINT	○						
PROMPT ¹	○						
PULLDOWN						○ ³	
RETRIEVE	○	○ ²					
RETURN	○	○	○		○	○	

注:

¹ DATAI、DATAIX、または LISTCOL 言語タグの PROMPT 属性により定義されたカーソル移動に影響されるプロンプトを処理する場合、PROMPT は CALL または RETURN になります。

² ACTFOR=PANEL が PDFLDC タグに指定されている場合のみ有効。

³ MBAR 属性が PANEL タグに指定されている場合のみ有効。

表 95. ダイアログ・コマンドの影響の要約

ダイアログ・コマンド名	影響			
	KEYI または PDFLDC タグ上の VARUPD ¹	ユーザー・メッセージの表示停止 ²	条件の評価	一般出口の呼び出し
ACTIONS	いいえ	いいえ	いいえ	いいえ
CALL	はい/いいえ	はい	はい	はい
CANCEL	はい/いいえ	はい	N/A	はい
CHGVIEW	はい	はい	はい	はい
CMD	はい/いいえ	はい	はい	はい
CMDLINE	いいえ	いいえ	いいえ	いいえ

表 95. ダイアログ・コマンドの影響の要約 (続き)

ダイアログ・ コマンド名	影響			
	KEYI または PDFLDC タグ上の VARUPD ¹	ユーザー・ メッセージの 表示停止 ²	条件の評価	一般出口の 呼び出し
DSPHELP	いいえ	いいえ	いいえ	いいえ
ENTER	はい	適用外 ³	適用外 ³	適用外 ³
EXIT	はい/いいえ	はい	適用外	はい
EXTHELP	いいえ	いいえ	いいえ	いいえ
HELP	いいえ	いいえ	いいえ	いいえ
HELPHelp	いいえ	いいえ	いいえ	いいえ
HELPIDX	いいえ	いいえ	いいえ	いいえ
HOME	はい	はい	はい	はい
KEYSHELP	いいえ	いいえ	いいえ	いいえ
MENU	はい	はい	はい	はい
MOREKEYS	いいえ	いいえ	いいえ	いいえ
MOVETOP	はい	いいえ	いいえ	いいえ
MSG	N/A	はい	はい	はい
PAGEDOWN	はい ⁶	いいえ	いいえ	はい ⁶
PAGEUP	はい ⁶	いいえ	いいえ	はい ⁶
PRINT	いいえ	いいえ	いいえ	いいえ
PROMPT ⁵	はい/いいえ	はい	はい	はい
PULLDOWN	いいえ	いいえ	いいえ	いいえ
RETRIEVE	いいえ	いいえ	いいえ	いいえ
RETURN	はい/いいえ	はい	N/A	はい
デフォルト ⁴	N/A	はい	はい	はい

注:

- ¹ VARUPD 属性は、アクションが機能キーまたはプルダウン選択項目に割り当てられている場合のみ適用されます。 ACTFOR=LIST が PDFLDC タグに指定されている場合、VARUPD は YES です。
- ² UIM は、別の関数が開始する直前に、アプリケーションにより生成されるメッセージ (ユーザー・メッセージ) の表示を停止します。 HELP は、そのヘルプがコマンド用であり、ユーザーがヘルプ画面からプロンプトとコマンドの処理を直接開始する場合には、関数として使用することができます。
- ³ ENTER は、これらのアクションが実行される時には別の関数と常に等価になるため、これらの ENTER 値は適用されません (適用外)。
- ⁴ ENTER または SELECT のデフォルトのアクションとして、UIM はパネルを再度表示します。
- ⁵ DATAI、DATAIX、または LISTCOL 言語タグの PROMPT 属性により定義されたカーソル移動に影響されるプロンプトを処理する場合、PROMPT は CALL または RETURN になります。
- ⁶ PAGEUP および PAGEDOWN がメッセージ領域、情報域、またはメニュー域に適用される場合、これらの値は適用されません。

VARUPD 属性

関数の主な修飾子の 1 つに、KEYI (キー・リスト項目) および PDFLDC (プルダウン・フィールド選択項目) 言語タグの VARUPD 属性があります。この属性は、ダイアログ変数およびリスト項目が、ユーザーが入力した値で更新されるかどうかを定義します。この属性は、キーに関連した関数が、画面上の入力フィールドの値の妥当性検査中にエラーが検出された場合に処理されるかどうかも定義します。ほとんどのダイアログ・コマンドでは、VARUPD に対して、パネル定義によりオーバーライドできない特定の値を想定しています。以下に VARUPD の影響の要約を示します。

アクションのタイプ	VARUPD 属性の影響	
	VARUPD=YES	VARUPD=NO
QUIDSPP API を呼び出したアプリケーションに制御を戻す。	すべての妥当性検査が実行され、正しい場合にはダイアログ変数およびリスト項目が更新されます。1 つ以上の変数で妥当性検査に失敗すると、エラー・メッセージと共にパネルが再度表示されます。	妥当性検査は実行されず、ダイアログ変数およびリスト項目は更新されません。入力フィールドに対して最後の操作の後にユーザーにより行われた変更は、失われます。ユーザーにより入力される値は、QUIDSPP を呼び出し、再表示パラメーターに「はい」を指定することにより回復できることがあります。詳しくは、API のトピックを参照してください。
アプリケーションに戻らずに UIM の制御下で実行されるアクション	すべての妥当性検査が実行され、正しい場合にはダイアログ変数およびリスト項目が更新されます。1 つ以上の変数で妥当性検査に失敗すると、エラー・メッセージと共にパネルが再度表示されます。	妥当性検査は実行されず、ダイアログ変数およびリスト項目は更新されません。保管された値を廃止してしまうような更新が行われないうえに、入力フィールドの内容は UIM により保管され、パネルを再度表示する際に使用されます。たとえば、保管されたパネルに表示されたダイアログ変数を変更すると、保管された値が廃棄され、パネルが再度表示されるときに新しい値が表示されます。

ACTIONS (メニュー・バー・カーソル・アクション)

▶▶—ACTIONS—◀◀

パネルとメニュー・バーの間でカーソル位置を代替します。共通ユーザー・アクセス (CUA) ガイドラインでは、この関数を F10 に割り当てることを推奨しています。

ACTIONS 機能キーを押すと、UIM はカーソルが置かれていた場所の情報を保管し、カーソルをメニュー・バーの最初の選択項目に移動します。ACTIONS 機能キーを再度押した場合、カーソルは、メニュー・バーに移動する前の直前のカーソル位置に戻ります。ユーザーがカーソル移動キー (矢印キー) を使用してカーソルをメニュー・バー領域に移動した後に ACTIONS 機能キーを押した場合、UIM は最適なカーソル位置を判別します。

メッセージ

処理中に UIM によりメッセージは出されません。

CALL (プログラム呼び出し)

▶▶—CALL—*dialog-variable-name*—▶▶

アプリケーション・プログラムを呼び出し、アプリケーションがプログラムと画面を互いにリンクできるようにします。

呼び出し関数は、アドレスにより、名前により、または拡張プログラム・モデル (EPM) 呼び出しにより呼び出します。

最高のパフォーマンスを得たい場合には、アドレスによる呼び出しを使用してください。アドレスによる呼び出しが適切でない場合、名前による呼び出し、またはライブラリー修飾プログラム名を使った EPM 呼び出し (*LIBL は使用しない) を使用してください。

UIM と呼び出されるプログラム間のインターフェースの説明については、API トピックの「User Interface Management EXIT Program」という節を参照してください。CALL ダイアログ・コマンドは、以下において使用できます。

- メニュー項目アクション
- 機能キー・アクション
- 処置リストのオプション
- 処置リスト出口
- 一般パネル出口
- デフォルトの入力処置
- アプリケーション・フォーマット出口
- コンテキスト・プロンプト出口
- プルダウン選択項目アクション

CALL ダイアログ・コマンドは、メニュー作成 (CRTMNU) コマンドを使用して作成されたメニュー・オブジェクト上では使用できませんが、パネル・グループ作成 (CRTPNLGRP) コマンドを使用して作成されたパネル・グループ・オブジェクト内では使用できます。外部メニューのメニュー・オプションの結果としてプログラムの呼び出しを必要としている場合には、CMD ダイアログ・コマンドを使用して CALL CL コマンドを実行依頼してください。ユーザーには、CALL CL コマンドおよび呼び出されるプログラムに対する適切な権限がなければなりません。

必須パラメーター

dialog-variable-name

呼び出しの 3 つのタイプごとに、以下に説明されている呼び出し情報が含まれるダイアログ変数を指定します。

アドレスによる呼び出し

このタイプの呼び出しは i5/OS システム・プログラムで使用され、他のシステム・モジュールの呼び出しを許可します。指定されたダイアログ変数に `BASETYPE='PTR'` を定義してください。呼び出し関数が要求される際、ダイアログ変数にはプログラムへのシステム・ポインターが含まれていなければなりません。ポインターに適切な権限があることを確認してください。

名前による呼び出し

プログラム・オブジェクトの修飾名を使用したアプリケーション・プログラムの呼び出しを許

CALL コマンド

可します。指定されたダイアログ変数に `BASETYPE='CHAR 20'` を定義してください。呼び出し関数が要求される際、ダイアログ変数の最初の 10 文字にはプログラム・オブジェクトの名前が含まれていなければなりません。次の 10 文字にはプログラムが入っているライブラリーの名前が含まれていなければなりません。プログラムおよびライブラリーの名前は左寄せにする必要があります、引用符で囲まれたオブジェクト名が使用されている場合以外は大文字でなければなりません。

拡張プログラム・モデル (EPM) による呼び出し

このタイプの呼び出しでは、EPM 言語プログラムの入り口点の呼び出しを許可します。UIM は `QPXXCALL` インターフェースを使用して呼び出しを行います。指定されたダイアログ変数に `BASETYPE='CHAR 130'` を定義してください。呼び出し関数が要求される際、ダイアログ変数には以下の内容が含まれていなければなりません。

文字	内容
1-20	EPM プログラム・オブジェクトの名前およびオブジェクトが入っているライブラリーの名前
21-120	呼び出される外部入り口点の名前
121-130	入り口点が属する環境の名前

UIM はこの情報を `QPXXCALL` 言語インターフェース・プログラムに渡すだけなので、`QPXXCALL` により定義された要件にダイアログ変数の内容が従っていることを確認してください。UIM が `QPXXCALL` プログラムを呼び出す際、EPM 環境は暗黙的に開始され、明示的に削除されるまで活動状態を保ちます。

注: EPM プログラムの呼び出しには、大きなパフォーマンスのオーバーヘッドが伴います。可能であれば、アドレスによる呼び出しまたは名前による呼び出しを使用してください。EPM プログラムの呼び出しのサポートは、アプリケーション設計の事情で EPM プログラムが `CALL` ダイアログ・コマンドのターゲットとして使用される場合に提供されます。

VARUPD 値

`CALL` ダイアログ・コマンドには、事前定義された `VARUPD` 値はありません。`CALL` が機能キーに割り当てられる時、機能キーの `VARUPD` 値は、妥当性検査を実行するかどうかを決定し、その後ダイアログ変数を更新します。

メッセージ

処理中に以下のエラーが発生する可能性があります。

- プログラムが見つからない。
- 指定されたプログラムに対する権限がない。
- 指定されたプログラムに対するパラメーター数が無効である。
- プログラムが `STG(*FREE)` で保管されている。
- `QPXXCALL` 言語インターフェース・プログラムにより通知された例外。

これらのエラーが、一般パネル出口の呼び出し、リスト・アクション出口の呼び出し、およびアプリケーション・フォーマット済み領域の呼び出しで発生した場合、UIM は処理を停止しません。

他のすべての呼び出しタイプでは、UIM はそれが実行していたアクションを終了します。ほとんどの場合、これには、プログラム呼び出しに成功しなかったことを示すエラー・メッセージと共にパネルを再度表示させることが含まれます。

CANCEL



1 つの画面をバックアップし、メニュー項目アクションまたは機能キー・アクションで使用されます。

現行画面が QUIDSPP API により呼び出された場合、関数パラメーターには CANCEL が要求されたという標識が入ります。

現行画面が、メニュー高速機能 (たとえば、コマンド行で GO と入力する) を使用した直前の画面から呼び出されたメニューである場合、UIM は直前の画面を表示します。

オプション・パラメーター

NOSET/SET

取り消しフラグが設定されているかどうかを示します。NOSET は、取り消しフラグが設定されていないことを意味します。これはデフォルト値です。SET は、取り消しフラグが設定されていることを意味します。取り消しフラグの詳細は、388 ページの『リスト・パネルをまとめて閉じる』を参照してください。

VARUPD 値

CANCEL には事前定義された VARUPD 値はありません。CANCEL が機能キーに割り当てられる時、機能キーの VARUPD 値は、妥当性検査を実行するかどうかを決定し、その後ダイアログ変数を更新します。

メッセージ

処理中にメッセージは出されません。

CHGVIEW (ビュー変更)



リストの表示されるビューを変更します。機能キーまたはプルダウン・フィールド選択項目アクションのみに割り当てることができます。

ユーザーが見る必要がある情報の欄がいくつもあり、すべての欄が画面に収まらない場合、ユーザーがある欄のセットを表示した後に別の欄のセットを表示するようにビューを変更することができます。これは、すべての情報欄が表示されるまで続きます。

リスト域を表示するいくつかの異なる方法を LISTVIEW 言語タグで指定したり、どの欄が各ビューに属するのかを指定したりすることができます。リスト・ビューは 0 から開始して順次に番号が振られます。たとえばリスト域に 5 つの表示ビューがある場合、ビューは 0 から 4 まで番号が振られます。リスト域のパネル・ソース内の最初の LISTVIEW タグはビュー 0 を定義し、リスト域のパネル・ソース内の 2 番目の LISTVIEW タグはビュー 1 を定義する、というようにリスト域内のすべての LISTVIEW タグについて定義します。

CHGVIEW コマンド

CHGVIEW ダイアログ・コマンドで、ビューを切り替えることができます。この関数は循環方式で作動します。たとえば、現行ビューが 3 である場合、CHGVIEW でビュー 4 を表示します (ビュー 4 がある場合)。ビュー 4 がない場合にはビュー 0 に戻ります。LIST タグの VIEW 属性でコード化されているダイアログ変数を変更することにより、アプリケーションで指定してビューを選択することができます。

CHGVIEW ダイアログ・コマンドが実行されるときには必ず、UIM もこのダイアログ変数を変更します。

要求がどのリストに適用されるかはカーソル位置によります。規則は PAGEDOWN および PAGEUP ダイアログ・コマンドの場合の規則に非常によく似ています。

- 複数のビューが含まれるように定義されたリスト域にカーソルがある場合、要求はその領域に適用されます。
- 複数のビューが含まれるように定義されたリスト域にカーソルがない場合、要求は複数のビューが含まれる画面の最初のリスト域に適用されます。

パラメーター

なし

VARUPD 値

CHGVIEW ダイアログ・コマンドは常に VARUPD=YES の指定で作動し、以下の操作が行われます。

- 妥当性検査は常に実行されます。
- 有効な値が処理され、ダイアログ変数が更新されます。
- 妥当性検査により障害が発生するとビューの変更が行われなくなり、該当するエラー・メッセージと共にスクリーンが再度表示されます。

メッセージ

妥当性検査でエラーがない場合は、処理中にメッセージは出されません。

CMD (システム・コマンド)

▶—CMD—*command-text*—▶

CL または OCL コマンドをシステム・コマンドに実行依頼します。

パラメーター

command-text

CMD ダイアログ・コマンドに続くすべてのテキストはコマンドとして処理され、処理のためにシステムに実行依頼されます。コマンドがネイティブのものかそれとも System/36 環境かは UIM によってではなく、システムによって判別されます。

コマンド・ストリングには、UIM がコマンドをシステムに実行依頼する前にコマンド・ストリングで置換されるダイアログ変数の名前が含まれる場合があります。それぞれのダイアログ変数名は、最初にアンパーサンド (&)、最後にピリオド (.) を付ける必要があります。末尾のピリオドは必須ではありませんが、あいまいさを避けるため使用することが勧められています。

コマンドを実行依頼する前に、UIM はダイアログ変数の置換を実行します。コマンド・ストリングの一部として指定されたそれぞれのダイアログ変数は、その表示可能な値に置換されます。この値は常にダイアログ変数の現行値から算出されます。

VARUPD 値

CMD ダイアログ・コマンドには、事前定義された VARUPD 値はありません。CMD が機能キーまたはプルダウン・フィールド選択項目に割り当てられる時、KEYI または PDFLDC 言語タグの VARUPD 属性は、妥当性検査を実行するかどうかを決定し、その後ダイアログ変数を更新します。コマンドの置換はダイアログ変数の値に依存するため、VARUPD=NO を使用する場合には注意が必要です。

メッセージ

妥当性検査でエラーがない場合は、処理中にメッセージは出されません。

ヒント

これらのコマンドはジョブ・ログに記録されないため、コマンド構文エラーを発見して訂正することが難しい場合があります。すべてのダイアログ変数が置換された後のコマンドを表示する 1 つの方法は、以下のように、コマンドを SNDMSG コマンド上のメッセージの値にすることです。

```
SNDMSG MSG (ここにコマンドを入れる) TOUSR (*REQUESTER)
```

CMDLINE (コマンド行)

▶▶—CMDLINE—◀◀

画面の下部にコマンド行を含むポップアップ・ウィンドウを表示します。

パラメーター

なし

メッセージ

処理中にメッセージは出されません。

DSPHELP (ヘルプ表示)

▶▶—DSPHELP—*help-module-name*—◀◀

└panel-group┘

└product-library┘

指定された UIM ヘルプ・モジュールを表示します。

必須パラメーター

help-module-name

表示されるヘルプ・モジュールの名前を指定します。

オプション・パラメーター

qualified-panel-group-name

ヘルプ・モジュールを含むパネル・グループの名前を指定します。これを指定しない場合、次のいずれかになります。(1) ヘルプ・モジュールは、DSPHELP ダイアログ・コマンドが含まれる同じパネル・

DSPHELP コマンド

グループからのものでなければなりません。(2) ヘルプ・モジュールは、IMPORT タグを使用して別のパネル・グループからインポートする必要があります。

product-library

ヘルプ・モジュールが表示されるときに製品ライブラリーとなるライブラリーの名前を指定します。このパラメーターは、ハイパーテキスト・リンク・アクションのために DSPHELP ダイアログ・コマンドが使用される場合にのみ使用されます。

メッセージ

処理中にメッセージは出されません。

ENTER

▶▶—ENTER—◀◀

アクションの処理を許可します。Enter キーのみで許可されます。

詳しくは、『ENTER、HELP、および PROMPT ダイアログ・コマンドの使用における考慮事項』を参照してください。

パラメーター

なし

VARUPD 値

ENTER ダイアログ・コマンドは常に VARUPD=YES の指定で作動し、以下の操作が行われます。

- 妥当性検査は常に実行されます。
- 有効な値が処理され、ダイアログ変数が更新されます。
- 妥当性検査で障害が発生すると操作が停止します。該当するエラー・メッセージと共に画面が再度表示されます。

メッセージ

コマンドの実行依頼または取り消しの結果として生成されたメッセージは、すべてメッセージ行に表示されます。妥当性検査エラーが発生した場合、追加メッセージが送信される場合があります。

ENTER、HELP、および PROMPT ダイアログ・コマンドの使用における考慮事項

ENTER、HELP、および PROMPT に対して実行されるアクションは、画面のタイプおよびコマンド行の内容に大きく依存します。これらのダイアログ・コマンドは機能キーのみに割り当てることができ、他のダイアログ・コマンドとは異なる特性を備えています。これらのダイアログ・コマンドはどちらかというアクション修飾子のような働きをし、システム・コマンド・ストリングのように、より具体的なアクションを判別または変更するために使用されます。

たとえば、処置リスト・パネルには、さまざまなフィールドの内容を基にした、実行されるいくつかのアクションの記述があります。ENTER および PROMPT ダイアログ・コマンド、および頻度は少ないものの HELP ダイアログ・コマンドは初期に要求されますが、残りのパネルをどのように解釈するかを記述する役割しか持ちません。何を実行するかを記述するためにはこれらだけでは不十分です。

以下に、これらのダイアログ・コマンドの記述を読む際に覚えておくべき重要な点を記します。

- コマンド行が含まれるメニューには、メニュー項目またはコマンドからの選択を含めることができます。最初の非空白文字ストリングが引用符で囲まれておらず、数値のみが含まれている場合、その文字ストリングはメニュー項目として解釈されます。それ以外の場合には、その文字ストリングはコマンドと見なされます。
- コマンド行が含まれる処置リストには、パラメーター・ストリングまたはコマンドを含めることができます。処置リスト上のいずれかの項目が選択された場合、コマンド行にあるすべてのストリングはパラメーター・リストとして処理されます。それ以外の場合には、そのストリングはコマンドとして処理されます。

EXIT (表示終了)



ユーザーが画面のグループからバックアウトすることを許可します。

現行画面が QUIDSPP API を使用して表示された場合、UIM は関数により要求されたパラメーターを設定し、EXIT ダイアログ・コマンドが要求されたことを示します。次に何を行うかはアプリケーションが決定します。

現行画面が一連の画面から呼び出されたメニューである場合、UIM は、MENU コマンドが最初に要求された画面を表示します。EXIT ダイアログ・コマンドは、一連のメニューから完全にバックアウトします。

オプション・パラメーター

NOSETSET

ジョブの終了フラグが設定されているかどうかを示します。NOSET は、終了フラグが設定されていないことを意味します。これはデフォルトです。SET は、終了フラグが設定されていることを意味します。終了フラグの詳細は、387 ページの『EXIT の要求時に複数のパネルをまとめて閉じる』を参照してください。

VARUPD 値

EXIT ダイアログ・コマンドには、事前定義された VARUPD 値はありません。ただしユーザー・インターフェース・スタイルは、VARUPD=NO を通常は使用する必要があることを示唆しています。EXIT が機能キーまたはプルダウン選択項目に割り当てられる時、機能キーの VARUPD 値は、妥当性検査を実行するかどうかを決定し、その後ダイアログ変数を更新します。

メッセージ

処理中にメッセージは出されません。

EXTHELP (拡張ヘルプ)

▶▶—EXTHELP—◀◀

パネルの拡張ヘルプを表示します。

パラメーター

なし

HELP

▶▶—HELP—◀◀

ヘルプ情報を表示します。共通ユーザー・アクセス (CUA) ガイドラインでは、これを Help および F1 キーに割り当ててることを推奨しています。

取られる特定のアクション (および表示されるヘルプ・テキスト) は、画面タイプ、カーソル位置などにより異なります。UIM 規則については、380 ページの『文脈ヘルプの定義』を参照してください。特定の属性についての詳細は、708 ページの『ENTER、HELP、および PROMPT ダイアログ・コマンドの使用における考慮事項』を参照してください。

パラメーター

なし

VARUPD 値

常に VARUPD=NO です。妥当性検査は現行パネルの現行の内容に対して実行されず、ダイアログ変数の更新は行われません。

メッセージ

処理中に「使用可能なヘルプ情報はありません」というメッセージが出される場合があります。これはユーザーが、システムにより提供されたヘルプ・テキストを削除するかインストールしない選択をした場合に発生する可能性があります。

コマンド行で特定のコマンドに対するヘルプを要求した場合にも、追加のメッセージが出される場合があります。考えられるメッセージとしては、コマンドが存在しない、ユーザーにその権限がない可能性がある、といったものがあります。これらのメッセージが出されると、メッセージ領域に最初のメッセージとして表示されます。画面の他の状態を記述したメッセージも、前方にスクロールすることにより表示することができます。

HELPHELP

▶▶—HELPHELP—◀◀

文脈ヘルプ、拡張ヘルプ、機能キーのヘルプ、およびヘルプ・パネルのヘルプ索引について、ヘルプ機能の使用方法に関する情報を表示します。HELPHelp ダイアログ・コマンドは、メニュー・バーのプルダウン選択項目にのみコード化できます。

パラメーター

なし

メッセージ

処理中にメッセージは出されません。

HELPIIDX

▶▶—HELPIIDX—◀◀

パネル・グループ・タグで指定された SCHIDX 属性に対する索引検索機能を開始します。ユーザーは索引内のトピックを選択して表示または印刷できます。HELPIIDX ダイアログ・コマンドは、メニュー・バーのプルダウン選択項目に対してのみコード化できます。

パラメーター

なし

メッセージ

処理中にメッセージは出されません。

HOME (ホーム・メニュー表示)

▶▶—HOME—◀◀

ジョブのホーム・メニューを表示します。このメニューは、メニューのスタックに追加されます。これは、メニューのスタックの先頭に戻るという意味ではありません。HOME ダイアログ・コマンドは、機能キーのみで使用することができます。

注: HOME はメニュー域を持つパネルのみで許可されます。

パラメーター

なし

VARUPD 値

HOME ダイアログ・コマンドは常に VARUPD=YES の指定で作動し、以下の操作が行われます。

- 妥当性検査は常に実行されます。
- 有効な値が処理され、ダイアログ変数が更新されます。
- 妥当性検査により障害が発生すると操作が行われなくなり、該当するエラー・メッセージと共にスクリーンが再度表示されます。

HOME コマンド

メッセージ

処理中にメッセージは出されません。

KEYSHELP

▶▶—KEYSHELP—▶▶

機能キーのヘルプを表示されたパネルに表示します。共通ユーザー・アクセス (CUA) ガイドラインでは、これを F1 キーに割り当てることを推奨しています。

メッセージ

処理中にメッセージは出されません。

MENU

▶▶—MENU—*qualified-menu-name*—

RTNPNT
NORTNPNT

—▶▶

メニュー項目を選択したり機能キーを押した結果として、後続のメニューを表示します。

必須パラメーター

qualified-menu-name

表示されるメニューの修飾名を指定します。

オプション・パラメーター

RTNPNT|NORTNPNT

このパラメーターは、ダイアログ・コマンドが要求された場合に、現行メニューが戻り点と見なされるかどうかを示します。このパラメーターは、GO CL コマンドの戻り点 (RTNPNT) パラメーターと同様に機能します。RTNPNT は、終了キーを押した時に現行メニューが戻されることを示します。これはデフォルトです。戻り点なし (NORTNPNT) は、終了キーを押した時に現行メニューに戻らないことを示します。

パネル・グループ作成 (CRTPNLGRP) コマンドを使用してパネル・グループ・オブジェクトにコード化されている場合には、このパラメーターは影響を与えません。このパラメーターは、メニュー作成 (CRTMNU) コマンドを使用して作成されたメニュー・オブジェクト内で使用される場合のみ、意味を持ちます。

メッセージ

処理中にメッセージは出されません。

MOREKEYS (機能キー続き表示)

▶▶—MOREKEYS—◀◀

機能キーすべてがパネルに収まらない場合に活動状態の機能キーの追加リストを表示します。MOREKEYS ダイアログ・コマンドのアクションは、機能キー F24 のみに割り当てることができます。そのとき、キー項目 (KEYI) 言語タグの COND 属性を使用してそのアクションを条件付けることはできません。

アプリケーションが KEYI タグ上の ACTION=MOREKEYS をコード化する場合、機能キーの記述がすべて一度にパネルに収まらない場合に UIM が状態を管理します。

MOREKEYS ダイアログ・コマンドを使用して KEYL (キー・リスト) 言語タグを持つパネルが表示される時、UIM は、現在活動状態にあるすべての機能キーのリストを収集します。これは、KEYI 言語タグの PRIORITY 属性のそれぞれのキー項目に割り当てられている優先順位を基にしています。UIM は、パネルの機能キー域にできるだけ多くの機能キーを収めようとします。

その領域に機能キーのすべての記述が収まらない場合、UIM は、MOREKEYS ダイアログ・コマンドに割り当てられた機能キーを使用可能にし、機能キーの記述をパネルに配置します。機能キーのすべての記述が収まる場合、MOREKEYS ダイアログ・コマンドに割り当てられた機能キーは使用可能になりません。

MOREKEYS ダイアログ・コマンドに割り当てられた機能キーをユーザーが押すと、UIM は、キー・リストの終わった場所から開始し、機能キー域に収まるだけの活動状態の機能キーの記述を優先順位の順序でできるだけ多く収集します。MOREKEYS ダイアログ・コマンドに割り当てられた機能キーは使用可能になります。

このプロセスは、キー・リストの末尾に達するまで続けられます。MOREKEYS ダイアログ・コマンドが再度選択された場合、UIM はキー・リストの先頭に戻って開始され、MOREKEYS ダイアログ・コマンドの処理を再度開始します。

パラメーター

なし

メッセージ

処理中にメッセージは出されません。

MOVETOP (先頭へ移動)

▶▶—MOVETOP—◀◀

カーソルで選択された行をスクロール可能な情報域の先頭に移動します。

このダイアログ・コマンドに割り当てられている機能キーが押されると、UIM は、カーソルが含まれる行をスクロール可能な領域の先頭に移動します。UIM により挿入されたブランク行 (パラグラフ・セパレーターなど) にカーソルがある場合、その次の行がスクロール可能な領域の先頭に移動します。

PAGEDOWN

▶▶—PAGEDOWN—◀◀

画面またはパネル 1 つ分前方にスクロールします。機能キーのみに割り当てることができます。

未完了リストの処理、スクロール移動域の選択、および VARUPD=YES の指定を行う時、処理は、PAGEDOWN ダイアログ・コマンドの場合も PAGEUP ダイアログ・コマンドの場合と同様に行われます。

パラメーター

なし

メッセージ

UIM は、「すでに区域の最下部が表示されています」を送出する場合があります。

妥当性検査でエラーが発生した場合、追加メッセージが送信される場合があります。

PAGEUP

▶▶—PAGEUP—◀◀

画面またはパネル 1 つ分後方にスクロールします。機能キーのみに割り当てることができます。

UIM がデータのスクロール可能域全体を使用できる場合、またはスクロール可能域が完全である場合、スクロールはアプリケーションに妨害されずに実行できます。スクロール可能域が、選択済みアクションを含む処置リストである場合、この時点では何も実行されません。

不完全なリストが表示されて PAGEUP スクロールにより不完全なパネルが表示されるという特殊なケースもあります。このケースでは、UIM は指定されたプログラムを呼び出し、追加のリスト項目を追加するか、リストが完全であるというマークを付けるかのいずれかを行います。

複数のスクロール可能域を持つ画面またはパネルでは、カーソルの位置によりスクロールされる部分が決定されます。これには以下のような規則があります。

- カーソルがスクロール可能域にある場合、その領域が選択されます。メッセージ行はスクロール可能域と見なされます。

スクロール可能域の境界は、メニュー、情報、リスト、またはデータ域の先頭と末尾の行により定義されます。ただし、その領域にあるすべての情報が実際にスクロールされるわけではありません (指示行など)。

- カーソルがスクロール可能域の外部にある場合、パネル上の最初のスクロール可能域 (先頭から末尾まで) がスクロールの対象として選択されます。

パラメーター

なし

PAGEUP コマンド

VARUPD 値

PAGEUP は、メッセージ行のスクロール時を除いて VARUPD=YES で作動し、以下の操作が行われます。

- 妥当性検査は常に実行されます。
- 有効な値が処理され、ダイアログ変数が更新されます。
- 妥当性検査により障害が発生するとスクロールが行われなくなり、該当するエラー・メッセージと共にパネルが再度表示されます。

メッセージ行のスクロール時には、PAGEUP ダイアログ・コマンドは VARUPD=NO が有効であるかのように作動します。フィールドは正確でなくてもよく、ダイアログ変数は更新されません。

メッセージ

UIM は、「すでに区域の最上部が表示されています」を送出する場合があります。

妥当性検査でエラーが発生した場合、追加メッセージが送信される場合があります。

PRINT (画面印刷)

▶▶—PRINT—◀◀

現行画面を印刷します。5250 PRINT キーのみに割り当てることができます。

パラメーター

なし

VARUPD 値

適用されません。

メッセージ

妥当性検査中にエラーがないかぎり、処理中にメッセージは出されません。

PROMPT

▶▶—PROMPT—◀◀

コマンド、処置リストのオプション、および入力フィールドに対するプロンプトを出します。機能キーのみに割り当てることができます。その特定の機能は以下に依存します。

- 管理対象の画面のタイプ
- カーソルの位置
- コマンドまたはパラメーター行の内容

パラメーター

なし

VARUPD 値

詳細は、『入力フィールドのプロンプト』および『処置リスト・オプションまたはコマンドのプロンプト』を参照してください。

メッセージ

コマンドの実行依頼または取り消しの結果として生成されたメッセージは、すべてメッセージ行に表示されます。加えて、UIM は、以下のメッセージを送出する場合があります。

- メニュー項目に対するプロンプト試行は無効でした。
- カーソルは、プロンプト関数が許可されていない場所にあります。

妥当性検査によるエラーが発生した場合、追加メッセージが送信される場合があります。

入力フィールドのプロンプト

プロンプト関数は、関連したパネル定義タグの PROMPT 属性で指定されたプロンプト定義に基づいて、CALL または RETURN 関数になります。入力フィールドに対するプロンプトは、F4 キーに割り当ててください。

入力フィールドにプロンプトを出すとき、VARUPD が機能キーの定義で指定されている場合にはその処理が行われます。

機能キーに VARUPD=YES が指定されている場合、以下の操作が行われます。

- 妥当性検査は常に実行されます。
- 有効な値が処理され、ダイアログ変数が更新されます。
- 妥当性検査により障害が発生するとプロンプトが出されなくなり、該当するエラー・メッセージと共にパネルが再度表示されます。

機能キーに VARUPD=NO が指定されている場合、アプリケーション・プログラムは、プロンプトが定義されたタグの DSPVALUE 属性を使用することにより、検査されていないキー付きデータにアクセスすることができます。

入力フィールドのプロンプトについての詳細は、511 ページの『付録 A. UIM パネル・グループ定義言語』に記載されている以下のタグの PROMPT 属性の説明を参照してください。

- データ項目 (DATAI)
- データ項目エクステンダー (DATAIX)
- リスト列 (LISTCOL)

処置リスト・オプションまたはコマンドのプロンプト

処置リストのオプションまたはコマンド行にプロンプトを出す場合、PROMPT ダイアログ・コマンドは常に VARUPD=YES で作動し、以下の操作が行われます。

- 妥当性検査は常に実行されます。
- 有効な値が処理され、ダイアログ変数が更新されます。
- 妥当性検査により障害が発生するとプロンプトが出されなくなり、該当するエラー・メッセージと共にパネルが再度表示されます。

プロンプト関数の結果として、コマンド・ストリングをシステムに実行依頼するか、アプリケーション・プログラムを呼び出すかのいずれかとなります。これは、次の規則に従って実行されます。

PROMPT コマンド

- コマンド行にメニュー項目が含まれる場合、エラーが送信されます。UIM は、メニュー項目に対するプロンプトをサポートしません。
- カーソルがコマンド行上またはオプション欄にあり、パネルに、オプションが指定された処置リスト・パネルが含まれている場合、UIM は、選択された各オプションに関連した PROMPT アクションを選択します。PROMPT アクションは、システムにコマンドを実行依頼するか、アプリケーション・プログラムを呼び出すかのいずれかとなります。

定義されたプロンプト・アクションを持たないオプションを選択した場合、処置リストの処理が停止し、エラー・メッセージと共にパネルが再度表示されます。エラー・メッセージは、ユーザーは選択したオプションに対してプロンプトを出すことができないことを示します。この場合、選択したオプションの番号が、エラーを強調したメッセージと共に表示されます。

コマンド行の内容はリスト・アクション用のパラメーターとなります。これらのパラメーターは、UIM が更新するためのダイアログ変数を定義することによりアクセスできます。更新は、処置リストが処理される時のコマンド行の内容で行われます。詳しくは、511 ページの『付録 A. UIM パネル・グループ定義言語』の LIST タグを参照してください。

個々のオプションごとに別個のプロンプトを受け取ります。UIM は、あるコマンドから別のコマンドへプロンプトを出している間に追加された情報は適用しません。

プロンプト・アクションでアプリケーション・プログラムを呼び出すことになる場合、オプションは、UIM によるプロンプトなしで処理されます。UIM がアプリケーション・プログラムを呼び出すと、ユーザーがプロンプト関数を要求したという標識が出されます。アプリケーション・プログラムには、その固有のバージョンのプロンプトを提供する責任があります。

- コマンド行にコマンドが含まれている場合、プロンプトおよび処理に対してそのコマンドが実行依頼されます。その後コマンドは、通常のプロンプター機能により実行または取り消されます。ジョブが実行されている環境に基づいて、システムはコマンドを System/36 環境または REXX に渡すことができます。

コマンド行がブランク (それがコマンドを含んでいるという想定の子クラス) であった場合、メジャー・コマンド・グループ・メニューが表示されます。

PROMPT ダイアログ・コマンドについての追加情報は、708 ページの『ENTER、HELP、および PROMPT ダイアログ・コマンドの使用における考慮事項』を参照してください。

PULLDOWN (プルダウン・メニュー表示)

▶▶—PULLDOWN—◀◀

メニュー・バーの最初のプルダウン・メニューを表示します。

パラメーター

なし

RETRIEVE (コマンド・ストリング取得)

▶▶—RETRIEVE—◀◀

直前に入力したコマンドを表示します。

UIM は、コマンド行に入力された一連のコマンドを保持しています。RETRIEVE ダイアログ・コマンドが出されると、直前のコマンドがコマンド行に表示されます。他の関数が実行される前に RETRIEVE ダイアログ・コマンドが再度出された場合、次に新しいコマンドが表示されます。

メニュー項目の選択およびパラメーター・ストリングは取得できません。さらに、CMD ダイアログ・コマンドにより実行依頼されたコマンドも取得できません。

パラメーター

なし

VARUPD 値

RETRIEVE ダイアログ・コマンドは、常に VARUPD=NO で作動します。妥当性検査は現行パネルの内容に対して実行されず、ダイアログ変数は更新されません。

メッセージ

処理中にメッセージは出されません。

RETURN (アプリケーションへの制御戻り)

▶▶—RETURN—*n*—◀◀

UIM に、該当する標識と共に制御をアプリケーションに戻させることにより、機能キー、メニュー項目、およびプルダウン選択項目を処理する方法をアプリケーションに提供します。

注: アプリケーションに対してリフレッシュ/再表示機能を提供するために、RETURN ダイアログ・コマンドを使用してください。

LIST タグの ACTOR 属性を使用することにより、処置リストに対する RETURN ダイアログ・コマンドの効果を得ることができます。この属性は、UIM が処置リストを処理する必要があるかどうかを決定します。

必須パラメーター

n このパラメーターは、1 から 32767 までの数値を指定します。この値は、RETURN 関数が出現した時に、アプリケーションの QUIDSPP API の、関数が要求するパラメーターに戻されます。

VARUPD 値

RETURN ダイアログ・コマンドには、事前定義された VARUPD 値はありません。RETURN が機能キーに割り当てられる時、機能キーの VARUPD 値は、妥当性検査を実行するかどうかを決定し、その後ダイアログ変数を更新します。

RETURN コマンド

メッセージ

妥当性検査中にエラーがないかぎり、処理中にメッセージは出されません。

付録 C. ディスプレイ・ファイルのフィードバック域のレイアウト

この付録には、プロダクト・センシティブ・プログラミング・インターフェースおよび関連したガイダンス情報が含まれています。

ここに示されている表は、オープンされたディスプレイ・ファイルに関するオープンおよび入出力のフィードバック域について記述したものです。それらのフィードバック域の各項目に関して、次の情報が示されています。

- オフセット (フィードバック域の始めから、各項目の位置までのバイト数)
- データ・タイプ
- 長さ (バイト数)
- 内容 (項目の説明、および項目の有効値)

ディスプレイ・ファイル以外のファイルのフィードバック域のレイアウトに関する情報は、iSeries Information Center で「ファイルおよびファイル・システム」のコレクションを参照してください。

使用する高水準言語によるサポートによって、フィードバック情報へのアクセス方法とデータ・タイプの表現方法が異なります。詳しくは、使用されている高水準言語の資料をご覧ください。

オープン・フィードバック域

オープン・フィードバック域は、オープン・データ・パス (ODP) のうち、ファイルがオープンされた後、そのファイルについての情報が含まれている部分です。それには、ディスプレイ・ファイルのファイル固有の情報、およびそのファイルに対して定義されている各装置に関する情報も入っています。この情報は、オープン処理中に設定され、他の操作が実行されると更新されることもあります。

表 96. オープン・フィードバック域

オフセット	データ・タイプ	長さ	内容
0	文字	2	オープン・データ・パス (ODP) のタイプ: DS ディスプレイ・ファイル、テープ・ファイル、ICF ファイル、保管ファイル、プリンター・ファイル (スプールされない)、またはディスクレット・ファイル (スプールされない)。 DB データベース・メンバー。 SP スプールされるプリンター・ファイルまたはディスクレット・ファイル、あるいはインライン・データ・ファイル。
2	文字	10	オープンされたファイルの名前。 ODP タイプが DS (ディスプレイ・ファイル用) の場合、これはディスプレイ・ファイルの名前です。
12	文字	10	ファイルが入っているライブラリーの名前。インライン・データ・ファイルの場合、この値は *N です。
22	文字	10	ディスプレイには適用されません。
32	文字	10	ディスプレイには適用されません。
42	2 進数	2	ディスプレイには適用されません。
44	2 進数	2	最大レコード長。
46	文字	2	ディスプレイには適用されません。
48	文字	10	ディスプレイには適用されません。
58	2 進数	4	予約済み。

表 96. オープン・フィールドバック域 (続き)

オフセット	データ・タイプ	長さ	内容
62	2 進数	4	予約済み。
66	2 進数	2	ファイル・タイプ: 1 ディスプレイ 2 プリンター 4 ディスケット 5 テープ 9 保管 10 DDM 11 ICF 20 インライン・データ 21 データベース
68	文字	3	予約済み。
71	2 進数	2	1 表示画面上の行数。
73	2 進数	2	1 表示画面上の桁数。
75	2 進数	4	ディスプレイには適用されません。
79	文字	2	ディスプレイには適用されません。
81	文字	1	ディスプレイには適用されません。
82	文字	1	ディスプレイには適用されません。
83	文字	10	予約済み。
93	文字	10	予約済み。
103	2 進数	2	ディスプレイには適用されません。
105	2 進数	2	ブロック化レコード入出力を使用している場合に、1 つのブロック内で読み取りまたは書き込みを行えるレコードの最大数。
107	2 進数	2	ディスプレイには適用されません。
109	2 進数	2	ブロック化レコードの入出力レコードの増加分。ブロックにある次のレコードをアドレス指定するために、ブロックの各レコードの始めに加えなければならないバイト数。
111	2 進数	4	予約済み。
115	文字	1	その他のフラグ。 ビット 1: 予約済み。 ビット 2: 共用を有効にしたファイル 0 ファイルは共用を有効にしてオープンされていません。 1 ファイルは共用を有効にしてオープンされています (SHARE(*YES))。 ビット 3~5: ディスプレイには適用されません。 ビット 6: フィールド・レベル記述 0 ファイルにはフィールド・レベル記述が入っていません。 1 ファイルにフィールド・レベル記述が入っています。

表 96. オープン・フィールドバック域 (続き)

オフセット	データ・タイプ	長さ	内容
			ビット 7: DBCS 可能ファイル 0 ファイルは DBCS 可能ではありません。 1 ファイルは DBCS 可能です。
116	文字	10	ビット 8: ディスプレイには適用されません。 ディスプレイ・ファイルの場合、これは要求元プログラム装置である表示装置記述の名前です。 このフィールドが提供されるのは、オープンまたは獲得操作によって *REQUESTER の装置名がディスプレイ・ファイルに付加されたときだけです。その他の場合は、このフィールドには *N が含まれます。
126	2 進数	2	ファイル・オープン・カウント。そのファイルが共用を有効にしてオープンされていない場合は、このフィールドの値は 1 です。ファイルが共用を有効にしていれば、このフィールドには、現在このファイルに接続されているプログラムの数が入ります。
128	2 進数	2	予約済み。
130	2 進数	2	ディスプレイには適用されません。
132	文字	1	その他のフラグ。 ビット 1~4: ディスプレイには適用されません。 ビット 5: 別個の標識域 0 標識はプログラムの入出力バッファーにあります。 1 標識はプログラムの入出力バッファーにはありません。このファイルの作成時に、DDS キーワード、INDARA が使用されました。 ビット 6: ユーザー・バッファー 0 システムは、プログラム用の入出力バッファーを作成します。 1 ユーザー・プログラムが入出力バッファーを提供します。 ビット 7: 予約済み。
133	文字	2	ビット 8: ディスプレイには適用されません。 オープン識別コード。この値は、ファイルに対するフルオープン操作 (SHARE (*NO))、または SHARE(*YES) の指定されたファイルに対する最初のオープン操作で固有となります。このコードによって、ファイルを、関連データ待ち行列の項目に一致させることができます。
135	2 進数	2	このフィールドの値は、最大レコード様式長さで、データとファイル特有の情報 (オプション標識、応答標識、ソース順序番号、およびプログラム・システム間データなど) を含んだ長さです。値が 0 の場合は、オフセット 44 のフィールドを使用してください。
137	2 進数	2	ディスプレイには適用されません。
139	文字	1	その他のフラグ。 ビット 1~3: ディスプレイには適用されません。

表 96. オープン・フィールドバック域 (続き)

オフセット	データ・ タイプ	長さ	内容
			ビット 4: CCSID 文字置換
			0 CCSID データ変換の際に置換文字は使用されません。
			1 CCSID データ変換の際に置換文字を使用できます。
			ビット 5~8: 予約済み。
140	文字	6	予約済み。
146	2 進数	2	この ODP に定義された装置の数。ディスプレイの場合、これはディスプレイ・ファイル作成 (CRTDSPF) コマンドの DEV パラメーターで定義された装置の数によって決まります。
148	文字		装置名定義リスト。この配列については、『装置定義リスト』をご覧ください。

装置定義リスト

オープン・フィールドバック域の装置定義リストは、配列構造です。配列内の各項目には、ファイルに接続されている各装置に関する情報が入っています。この配列にある項目の数は、オープン・フィールドバック域のオフセット 146 の数値によって決まります。装置定義リストは、オープン・フィールドバック域のオフセット 148 から始まります。下記に示されているオフセットは、オープン・フィールドバック域の始めからの値ではなく、装置定義リストの始めからの値です。

表 97. 装置定義リスト

オフセット	データ・ タイプ	長さ	内容
0	文字	10	プログラム装置名。ディスプレイ・ファイルでは、この値は、装置記述の名前です。
10	文字	50	予約済み。
60	文字	10	装置記述名。ディスプレイ・ファイルでは、この値は、装置記述の名前です。

表 97. 装置定義リスト (続き)

オフセット	データ・ タイプ	長さ	内容
70	文字	1	装置クラス。ディスプレイの場合、装置クラスは hex 01 です。 装置タイプ。
hex 07			5251 ディスプレイ装置
hex 0B			5291 ディスプレイ装置
hex 0D			5292 ディスプレイ装置
hex 12			5555-B01 ディスプレイ装置
hex 13			3270 ディスプレイ装置
hex 15			グラフィック使用可能装置
hex 16			金融端末
hex 17			3180 ディスプレイ装置
hex 19			3277 DHCP 装置
hex 26			3179-2 ディスプレイ装置
hex 27			3196-A ディスプレイ装置
hex 28			3196-B ディスプレイ装置
hex 33			3197-C1 ディスプレイ装置
hex 34			3197-C2 ディスプレイ装置
hex 35			3197-D1 ディスプレイ装置
hex 36			3197-D2 ディスプレイ装置
hex 37			3197-W1 ディスプレイ装置
hex 38			3197-W2 ディスプレイ装置
hex 39			5555-E01 ディスプレイ装置
hex 3E			3476-EA ディスプレイ装置
hex 3F			3477-FG ディスプレイ装置
hex 40			3278 DHCP 装置
hex 41			3279 DHCP 装置
hex 42			ICF 金融装置
hex 43			小売業通信装置
hex 44			3477-FA ディスプレイ装置
hex 45			3477-FC ディスプレイ装置
hex 46			3477-FD ディスプレイ装置
hex 47			3477-FW ディスプレイ装置
hex 48			3477-FE ディスプレイ装置
hex 4D			ネットワーク仮想端末装置ディスプレイ装置
hex 51			5555-C01 ディスプレイ装置
hex 52			5555-F01 ディスプレイ装置
hex 56			3476-EC ディスプレイ装置

表 97. 装置定義リスト (続き)

オフセット	データ・ タイプ	長さ	内容
71	文字	1	<p>hex 58 5555-G01 ディスプレイ装置</p> <p>hex 59 5555-G02 ディスプレイ装置</p> <p>hex 5D 3486-BA ディスプレイ装置</p> <p>hex 5F 3487-HA ディスプレイ装置</p> <p>hex 60 3487-HG ディスプレイ装置</p> <p>hex 61 3487-HW ディスプレイ装置</p> <p>hex 62 3487-HC ディスプレイ装置</p>
72	2 進数	2	表示画面上の行数。
74	2 進数	2	表示画面での各行の桁数。
76	文字	2	<p>ビット・フラグ。</p> <p>ビット 1: 明滅機能</p> <p>0 ディスプレイには明滅する能力がありません。</p> <p>1 ディスプレイには明滅する能力があります。</p> <p>ビット 2: デバイスのロケーション。</p> <p>0 ローカル・デバイス。</p> <p>1 リモート・デバイス。</p> <p>ビット 3: 獲得状況。オープン時にデバイスが暗黙的に獲得される場合でも、このビットは設定されます。</p> <p>0 デバイスは獲得されていません。</p> <p>1 デバイスは獲得されています。</p> <p>ビット 4: 送信勧誘の状況。</p> <p>0 デバイスは送信勧誘されていません。</p> <p>1 デバイスは送信勧誘されています。</p> <p>ビット 5: データの可用性状況 (デバイスが送信勧誘されている場合のみ)。</p> <p>0 データは使用不可です。</p> <p>1 データは使用可能です。</p> <p>ビット 6: ディスプレイには適用されません。</p> <p>ビット 7: 要求元のプログラム装置。</p> <p>0 要求元のプログラム装置ではありません。</p> <p>1 要求元のプログラム装置。</p>

表 97. 装置定義リスト (続き)

オフセット	データ・ タイプ	長さ	内容
			ビット 8: DBCS 装置。
			0 装置には 2 バイト・データを処理する能力がありません。
			1 装置には 2 バイト・データを処理する能力があります。
			ビット 9~10: 予約済み。
			ビット 11: DBCS キーボード。
			0 キーボードには 2 バイト・データを入力する能力がありません。
			1 キーボードには 2 バイト・データを入力する能力があります。
			ビット 12~16: 予約済み。
78	文字	1	ディスプレイには適用されません。
79	文字	1	ディスプレイには適用されません。
80	文字	50	予約済み。

入出力フィードバック域

入出力操作の結果は、i5/OS メッセージおよび入出力フィードバック情報を使用することによってプログラムに通信されます。入出力フィードバック域は、プログラムがブロック化レコード入出力を使用しているでない限り、各入出力操作のたびに更新されます。その場合、フィードバック域はレコードのブロックが読み取られたとき、または書き出されたときにのみ更新されます。ある情報は、ブロックの最後のレコードを反映しています。また、入出力操作回数のように、レコードのブロックへの操作回数 (レコードの数ではなく) を反映している情報もあります。プログラムでブロック化レコード入出力を使用しているかどうかを判別するには、使用されている高水準言語についての資料を参照してください。

入出力フィードバック域には、共通域とファイル依存域の 2 つがあります。

共通入出力フィードバック域

表 98. 共通入出力フィードバック域

オフセット	データ・ タイプ	長さ	内容
0	2 進数	2	ファイル依存フィードバック域へのオフセット。
2	2 進数	4	書き込み操作回数。書き込み操作が正常に完了したときにのみ更新されません。ブロック化レコード入出力操作では、この回数はブロックの数であって、レコードの数ではありません。
6	2 進数	4	読み取り操作回数。読み取り操作が正常に完了したときにのみ更新されません。ブロック化レコード入出力操作では、この回数はブロックの数であって、レコードの数ではありません。
10	2 進数	4	書き込み・読み取り操作回数。書き込み・読み取り操作が正常に完了したときにのみ更新されます。

表 98. 共通入出力フィールドバック域 (続き)

オフセット	データ・ タイプ	長さ	内容
14	2 進数	4	その他の操作回数。書き込み、読み取り、または書き込み・読み取り操作以外の正常終了した操作の数。操作が正常に完了したときにのみ更新されま す。この数には、更新、削除、データの強制終了、ボリュームの強制終了、 データ終了の変更、レコード・ロックの解放、およびデバイスの獲得/解放 の各操作が含まれます。
18	文字	1	予約済み。
19	文字	1	現行の操作。
			hex 01 読み取り、ブロック読み取り、または送信勧誘されたデバイスから の読み取り
			hex 02 直接読み取り
			hex 03 キーによる読み取り
			hex 05 書き込みまたはブロックの書き込み
			hex 06 書き込み・読み取り
			hex 07 更新
			hex 08 削除
			hex 09 データの強制終了
			hex 0A ボリュームの強制終了
			hex 0D レコード・ロックの解放
			hex 0E データ終了の変更
			hex 0F 削除の PUT
			hex 11 デバイスの解放
			hex 12 デバイスの獲得
20	文字	10	処理されたレコード様式の名前。以下のいずれかです。 <ul style="list-style-type: none"> • 入出力要求で指定された名前。 • デフォルトまたは形式選択処理によって決定された名前。

ディスプレイ・ファイルの場合、デフォルト名はファイル内の唯一のレコー
ド様式の名前、または入力可能フィールドを含むディスプレイに書き込まれ
るレコードの前のレコード様式の名前です。ディスプレイ・ファイルでは、
そのディスプレイに対して同時に複数の様式が指定されていることがあるの
で、この様式は最後のカーソル位置が入力されたときの様式を表さないこと
があります。

表 98. 共通入出力フィールドバック域 (続き)

オフセット	データ・ タイプ	長さ	内容
30	文字	2	装置クラス: バイト 1: hex 00 データベース hex 01 ディスプレイ hex 02 プリンター hex 04 ディスケット hex 05 テープ hex 09 保管 hex 0B ICF バイト 2 (バイト 1 がディスプレイを示す 16 進数 01 の場合): hex 07 5251 ディスプレイ装置 hex 0B 5291 ディスプレイ装置 hex 0D 5292 ディスプレイ装置 hex 12 5555-B01 ディスプレイ装置 hex 13 3270 ディスプレイ装置 hex 15 グラフィック使用可能装置 hex 16 金融端末 hex 17 3180 ディスプレイ装置 hex 19 3277 DHCF 装置 hex 26 3179-2 ディスプレイ装置 hex 27 3196-A ディスプレイ装置 hex 28 3196-B ディスプレイ装置 hex 33 3197-C1 ディスプレイ装置 hex 34 3197-C2 ディスプレイ装置 hex 35 3197-D1 ディスプレイ装置 hex 36 3197-D2 ディスプレイ装置 hex 37 3197-W1 ディスプレイ装置 hex 38 3197-W2 ディスプレイ装置 hex 39 5555-E01 ディスプレイ装置 hex 3E 3476-EA ディスプレイ装置 hex 3F 3477-FG ディスプレイ装置 hex 40 3278 DHCF 装置 hex 41 3279 DHCF 装置 hex 42 ICF 金融装置 hex 43 小売業通信装置

表 98. 共通入出力フィールドバック域 (続き)

オフセット	タイプ	長さ	内容
30	文字	2	装置クラス: バイト 2 (バイト 1 がディスプレイを示す 16 進数 01 の場合): (続き)
			hex 44 3477-FA ディスプレイ装置
			hex 45 3477-FC ディスプレイ装置
			hex 46 3477-FD ディスプレイ装置
			hex 47 3477-FW ディスプレイ装置
			hex 48 3477-FE ディスプレイ装置
			hex 4D ネットワーク仮想端末装置ディスプレイ装置
			hex 51 5555-C01 ディスプレイ装置
			hex 52 5555-F01 ディスプレイ装置
			hex 56 3476-EC ディスプレイ装置
			hex 58 5555-G01 ディスプレイ装置
			hex 59 5555-G02 ディスプレイ装置
			hex 5D 3486-BA ディスプレイ装置
			hex 5E 3486-BG ディスプレイ装置
			hex 5F 3487-HE ディスプレイ装置
			hex 60 3487-HD ディスプレイ装置
			hex 61 3487-HW ディスプレイ装置
			hex 62 3487-HC ディスプレイ装置
32	文字	10	装置名。ディスプレイの場合、操作が直前に終了した装置の名前。
42	2 進数	4	最後の入出力操作で処理されたレコードの長さ。
46	文字	80	予約済み。
126	2 進数	2	ブロックされたレコードに対する読み取り要求で取得したレコード数、またはブロックされたレコードに対する書き込み、データの強制終了、またはボリュームの強制終了の要求で送信されたレコード数。
128	2 進数	2	出力の場合、このフィールドの値はレコード様式の長さで、先頭文字用紙制御、オプション標識、ソース順序番号、およびプログラム・システム間データを含んだ長さです。値が 0 の場合は、オフセット 42 のフィールドを使用してください。 入力の場合、このフィールドの値はレコード様式の長さで、応答標識およびソース順序番号を含んだ長さです。値が 0 の場合は、オフセット 42 のフィールドを使用してください。
130	文字	2	予約済み。
132	2 進数	4	ディスプレイには適用されません。
136	文字	8	予約済み。

ディスプレイ・ファイルに関する入出力フィードバック域

表 99. ディスプレイ・ファイルに関する入出力フィードバック域

オフセット	データ・ タイプ	長さ	内容
0	文字	2	フラグ・ビット。
		ビット 1:	読み取りの取り消し標識。
		0	読み取りの取り消し操作は、読み取り要求を取り消しませんでした。
		1	読み取りの取り消し操作は、読み取り要求を取り消しました。
		ビット 2:	データの戻り標識。
		0	読み取りの取り消し操作は、入力バッファの内容を変更しませんでした。
		1	読み取りの取り消し操作は、待機なしの読み取り操作からのデータを入力バッファに入れました。
		ビット 3:	コマンド・キー標識
		0	この標識を設定する条件は生じませんでした。
		1	Print、Help、Home、Roll Up、Roll Down、または Clear キーが押されました。このキーでは DDS キーワードが使用可能になっていますが、応答標識は指定されていません。
		ビット 4~16:	予約済み。

表 99. ディスプレイ・ファイルに関する入出力フィールドバック域 (続き)

オフセット	データ・ タイプ	長さ	内容																																																																		
2	文字	1	<p>アテンション標識バイト (AID)。このフィールドは、どの機能キーが押されたかを示します。このフィールドには hex F1 が常に含まれて、ディスプレイ装置で Enter キーが押されたことを模倣します。ディスプレイ・ファイルの場合、このフィールドにはデバイスから戻された 1 バイトの 16 進値が含まれます。</p> <table border="1"> <thead> <tr> <th>16 進コード</th> <th>機能キー</th> </tr> </thead> <tbody> <tr><td>hex 31</td><td>1</td></tr> <tr><td>hex 32</td><td>2</td></tr> <tr><td>hex 33</td><td>3</td></tr> <tr><td>hex 34</td><td>4</td></tr> <tr><td>hex 35</td><td>5</td></tr> <tr><td>hex 36</td><td>6</td></tr> <tr><td>hex 37</td><td>7</td></tr> <tr><td>hex 38</td><td>8</td></tr> <tr><td>hex 39</td><td>9</td></tr> <tr><td>hex 3A</td><td>10</td></tr> <tr><td>hex 3B</td><td>11</td></tr> <tr><td>hex 3C</td><td>12</td></tr> <tr><td>hex B1</td><td>13</td></tr> <tr><td>hex B2</td><td>14</td></tr> <tr><td>hex B3</td><td>15</td></tr> <tr><td>hex B4</td><td>16</td></tr> <tr><td>hex B5</td><td>17</td></tr> <tr><td>hex B6</td><td>18</td></tr> <tr><td>hex B7</td><td>19</td></tr> <tr><td>hex B8</td><td>20</td></tr> <tr><td>hex B9</td><td>21</td></tr> <tr><td>hex BA</td><td>22</td></tr> <tr><td>hex BB</td><td>23</td></tr> <tr><td>hex BC</td><td>24</td></tr> <tr><td>hex BD</td><td>Clear</td></tr> <tr><td>hex F1</td><td>Enter/Rec Adv</td></tr> <tr><td>hex F3</td><td>Help (操作エラー・モードではない)</td></tr> <tr><td>hex F4</td><td>Roll Down</td></tr> <tr><td>hex F5</td><td>Roll Up</td></tr> <tr><td>hex F6</td><td>Print</td></tr> <tr><td>hex F8</td><td>Record Backspace</td></tr> <tr><td>hex 3F</td><td>Automatic Enter (選択ライト・ペン用)</td></tr> </tbody> </table>	16 進コード	機能キー	hex 31	1	hex 32	2	hex 33	3	hex 34	4	hex 35	5	hex 36	6	hex 37	7	hex 38	8	hex 39	9	hex 3A	10	hex 3B	11	hex 3C	12	hex B1	13	hex B2	14	hex B3	15	hex B4	16	hex B5	17	hex B6	18	hex B7	19	hex B8	20	hex B9	21	hex BA	22	hex BB	23	hex BC	24	hex BD	Clear	hex F1	Enter/Rec Adv	hex F3	Help (操作エラー・モードではない)	hex F4	Roll Down	hex F5	Roll Up	hex F6	Print	hex F8	Record Backspace	hex 3F	Automatic Enter (選択ライト・ペン用)
16 進コード	機能キー																																																																				
hex 31	1																																																																				
hex 32	2																																																																				
hex 33	3																																																																				
hex 34	4																																																																				
hex 35	5																																																																				
hex 36	6																																																																				
hex 37	7																																																																				
hex 38	8																																																																				
hex 39	9																																																																				
hex 3A	10																																																																				
hex 3B	11																																																																				
hex 3C	12																																																																				
hex B1	13																																																																				
hex B2	14																																																																				
hex B3	15																																																																				
hex B4	16																																																																				
hex B5	17																																																																				
hex B6	18																																																																				
hex B7	19																																																																				
hex B8	20																																																																				
hex B9	21																																																																				
hex BA	22																																																																				
hex BB	23																																																																				
hex BC	24																																																																				
hex BD	Clear																																																																				
hex F1	Enter/Rec Adv																																																																				
hex F3	Help (操作エラー・モードではない)																																																																				
hex F4	Roll Down																																																																				
hex F5	Roll Up																																																																				
hex F6	Print																																																																				
hex F8	Record Backspace																																																																				
hex 3F	Automatic Enter (選択ライト・ペン用)																																																																				
3	文字	2	<p>カーソル位置 (行および桁)。プログラムにデータを戻すサブファイル操作ではない入力操作の際に更新されます。例として、hex 0102 は行 1、桁 2 を意味します。行 10、桁 33 は hex 0A21 となります。</p>																																																																		
5	2 進数	4	<p>実際のデータ長。入出力操作で処理されたレコード様式の長さ。</p>																																																																		

表 99. ディスプレイ・ファイルに関する入出力フィールドバック域 (続き)

オフセット	データ・タイプ	長さ	内容
9	2 進数	2	サブファイル・レコードの相対レコード番号。サブファイル・レコード操作の際に更新されます。入力操作では、データがプログラムに戻される場合にのみ更新されます。複数のサブファイルがディスプレイ上にある場合、このオフセットには最後に更新されたサブファイルの相対レコード番号が含まれます。
11	2 進数	2	SFLDSP が指定されたサブファイル制御レコードに対して最後の書き込み操作が行われた場合、最上部のサブファイル表示域に現在表示されている最小のサブファイル相対レコード番号を示します。ロールアップおよびロールダウン操作の際に更新されます。出力操作が、この値を最初に設定したサブファイル制御様式ではない様式に対して行われた場合、これはゼロにリセットされます。メッセージ・サブファイルに対しては設定されません。
13	2 進数	2	サブファイル内のレコードの合計数。いずれかのサブファイル・レコードに対する PUT 相対操作の際に更新されます。 SFLINZ キーワード・オプションがオンに設定されたサブファイル制御レコードに対する書き込み操作または書き込み/読み取り操作の際に、この数値はゼロに設定されます。 SFLINZ キーワード・オプションがオンに設定されたサブファイル制御レコードに対して書き込み操作または書き込み/読み取り操作が行われていないと想定した場合、レコードがディスプレイ上の複数のサブファイルに書き込まれるとき、このオフセットにはすべてのサブファイルに関するレコードの合計数が含まれます。
15	文字	2	アクティブな DDS WINDOW キーワードに対する相対的なカーソル位置 (行および桁)。プログラムにデータに戻すサブファイル操作ではない入力操作の際に更新されます。カーソル位置は、ウィンドウ内の使用可能な位置に基づいています。例として、hex 0C13 は行 12、桁 19 を意味します。
17	文字	17	予約済み。
34	文字	2	メジャー戻りコード。 00 操作が正常に完了しました。 02 入力操作が正常に完了しましたが、ジョブはキャンセル (制御) されています。 03 入力操作が正常に完了しましたが、受け取ったデータはありません。 04 出力例外。 08 デバイスはすでに獲得されます。 11 送信勧誘されたデバイスからの読み取りは成功しませんでした。 34 入力例外。 80 永続的なシステム・エラーまたはファイル・エラー。 81 永続的なセッション・エラーまたはデバイス・エラー。 82 獲得操作またはオープン操作が失敗しました。 83 リカバリー可能なセッションまたはデバイスのエラー。
36	文字	2	マイナー戻りコード。ディスプレイ・ファイルの値については、739 ページの『付録 D. ディスプレイ・ファイルの戻りコード』を参照してください。
38	文字	8	ディスプレイには適用されません。
46	文字	1	ディスプレイには適用されません。
47	文字	1	予約済み。
48	文字	1	ディスプレイには適用されません。
49	文字	10	ディスプレイには適用されません。
59	文字	4	予約済み。
63	文字	8	ディスプレイには適用されません。
71	文字	9	予約済み。

取得属性

取得属性操作を実行すると、特定のディスプレイ装置に関する情報を取得できます。

表 100. 取得属性

オフセット	データ・タイプ	長さ	内容
0	文字	10	プログラム装置名。
10	文字	10	装置記述名。この項目に関連した装置記述の名前。
20	文字	10	ユーザー ID。

表 100. 取得属性 (続き)

オフセット	データ・ タイプ	長さ	内容
30	文字	1	装置クラス。ディスプレイの場合、装置クラスは D です。

表 100. 取得属性 (続き)

オフセット	データ・ タイプ	長さ	内容 装置タイプ:
31	文字	6	
			3179 3179 ディスプレイ装置
			317902 3179-2 ディスプレイ装置
			3180 3180 ディスプレイ装置
			3196A 3196-A1/A2 ディスプレイ装置
			3196B 3196-B1/B2 ディスプレイ装置
			3197C1 3197-C1 ディスプレイ装置
			3197C2 3197-C2 ディスプレイ装置
			3197D1 3197-D1 ディスプレイ装置
			3197D2 3197-D2 ディスプレイ装置
			3197W1 3197-W1 ディスプレイ装置
			3197W2 3197-W2 ディスプレイ装置
			3270 3270 ディスプレイ装置
			3476EA 3476-EA ディスプレイ装置
			3476-EC 3476-EC ディスプレイ装置
			3477FA 3477-FA ディスプレイ装置
			3477FC 3477-FC ディスプレイ装置
			3477FD 3477-FD ディスプレイ装置
			3477FE 3477-FE ディスプレイ装置
			3477FG 3477-FG ディスプレイ装置
			3477FW 3477-FW ディスプレイ装置
			525111 5251 ディスプレイ装置
			5291 5291 ディスプレイ装置
			5292 5292 ディスプレイ装置
			529202 5292-2 ディスプレイ装置
			5555B1 5555-B01 ディスプレイ装置
			5555C1 5555-C1 ディスプレイ装置
			5555E1 5555-E01 ディスプレイ装置
			5555F1 5555-F1 ディスプレイ装置
			5555-G1 5555-G01 ディスプレイ装置
			5555-G2 5555-G02 ディスプレイ装置
			3486BA 3486-BA ディスプレイ装置
			3486BG 3486-BG ディスプレイ装置
			3487HC 3487-HC ディスプレイ装置
			3487HD 3487-HD ディスプレイ装置
			3487HE 3487-HE ディスプレイ装置
			3487HW 3487-HW ディスプレイ装置
			DHCF77 3277 DHCF 装置
			DHCF78 3278 DHCF 装置
			DHCF79 3279 DHCF 装置

表 100. 取得属性 (続き)

オフセット	データ・ タイプ	長さ	内容
37	文字	1	要求元のプログラム装置。このフラグは、この項目が *REQUESTER 装置を定義しているかどうかを示します。 N *REQUESTER 装置ではありません (通信のソース装置)。 Y *REQUESTER 装置です (通信のターゲット装置)。
38	文字	1	獲得状況。オープン時に装置が暗黙的に獲得される場合でも設定されます。 N デバイスは獲得されていません。 Y デバイスは獲得されています。
39	文字	1	送信勧誘の状況。 Y デバイスは送信勧誘されています。
40	文字	1	N デバイスは送信勧誘されていません。 データの使用可能性。 Y 送信勧誘されているデータは使用可能です。 N 送信勧誘されているデータは使用不可です。
41	2 進数	2	ディスプレイ上の行数。
43	2 進数	2	ディスプレイ上の列数。
45	文字	1	ディスプレイの明滅の許可。 Y デ스플레이には明滅する能力があります。 N デ스플레이には明滅する能力がありません。
46	文字	1	オンライン/オフラインの状況。 O デ스플레이はオンラインです。 F デ스플레이はオフラインです。
47	文字	1	ディスプレイのロケーション。 L ローカル・ディスプレイ。 R リモート・ディスプレイ。
48	文字	1	ディスプレイ・タイプ。 A 英数字またはカタカナ。 I DBCS。
49	文字	1	ディスプレイのキーボード・タイプ。 A 英数字またはカタカナ・キーボード。 I DBCS キーボード。
50	文字	1	ディスプレイには適用されません。
51	文字	1	ディスプレイには適用されません。
52	文字	1	ディスプレイには適用されません。
53	文字	8	ディスプレイには適用されません。
61	文字	8	ディスプレイには適用されません。
69	文字	8	ディスプレイには適用されません。
77	文字	8	ディスプレイには適用されません。
85	文字	8	ディスプレイには適用されません。
93	文字	8	ディスプレイには適用されません。
101	文字	1	制御装置の情報。 N デ스플레이は、非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置に接続されていません。 1 デ스플레이は、非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置 (タイプ 1) ⁵ に接続されています。 2 デ스플레이は、非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置 (タイプ 2) ⁵ に接続されています。 3 デ스플레이は、非プログラマブル・ワークステーション用の拡張インターフェースをサポートする制御装置 (タイプ 3) ⁵ に接続されています。

表 100. 取得属性 (続き)

オフセット	データ・ タイプ	長さ	内容
102	文字	1	ディスプレイの色表示機能。 Y カラー・ディスプレイ N モノクロ・ディスプレイ
103	文字	1	ディスプレイによる罫線サポート。 Y ディスプレイは罫線をサポートします。 N ディスプレイは罫線をサポートしません。
104	文字	1	ディスプレイには適用されません。
105	文字	8	ディスプレイには適用されません。
113	文字	31	予約済み。

5.

タイプ 1

ウィンドウ、選択フィールド、スクロール・バー、および継続カーソル進行などをサポートする、V2R2 で使用可能な制御装置。

タイプ 2

V2R3 で使用可能な制御装置。これらは V2R2 の機能すべてに加えて、メニュー・バー、継続入力フィールド、編集マスク、および簡単なホット・スポットをサポートします。

タイプ 3

V3R1 で使用可能な制御装置。これらは V2R2 および V2R3 の機能をすべてサポートします。さらに、ウィンドウの下部の枠でのテキストをサポートします。

付録 D. ディスプレイ・ファイルの戻りコード

このセクションには、ディスプレイ・ファイルに関する、すべてのメジャーおよびマイナー戻りコードの説明が記載されています。これら戻りコードは、ディスプレイ・ファイルの入出力フィードバック域にセットされます。戻りコードは、それぞれの操作の結果を報告するものです。操作を発行したアプリケーション・プログラムは、適切な戻りコードを入手することができます。プログラムは、戻りコードをチェックして、それに応じて適切な処置を取ります。それらの戻りコードにアクセスする方法については、高水準言語の資料を参照してください。

戻りコードは、4桁の値で、最初の2桁はメジャー・コードで、後の2桁はマイナー・コードです。一部の戻りコードについては、ジョブ・ログまたはオペレーター・メッセージ待ち行列 (QSYSOPR) にもメッセージが送られます。追加情報を得るために、そのメッセージを参照することができます。

メジャー・コード 00

メジャー・コード 00 - 操作が正常に完了。

説明: プログラムによって発行された操作は、正常に完了しました。

処置: 次の操作を続けます。

コード	説明 / 処置
0000	<p>説明: プログラムによって実行された入力操作の場合、0000 は、入力操作が正常に行われ、なんらかのデータが受信されたことを示します。</p> <p>プログラムによって実行された出力操作の場合、0000 は、最後の出力操作が正常に完了したことを示します。</p> <p>処置: プログラムを継続できます。エラーではないが、プログラムにとって重要であるかもしれない異常状態を知らせるために、以下にリストされたメッセージのいずれかが出されることがあります。</p> <p>メッセージ:</p> <ul style="list-style-type: none">CPF4018 (状況)CPF4019 (診断)CPF4054 (診断)CPF4082 (診断)CPF4410 (診断)CPF5003 (状況)CPF5508 (診断)

メジャー・コード 02

メジャー・コード 02- 入力操作が正常に完了したが、ジョブが終了中 (制御された)。

説明: プログラムによって発行された入力操作は、正常に完了しました。しかし、ジョブが終了中 (制御された) です。

処置: プログラムを正しく完了するために、表示処理をできるだけ早く完了する必要があります。システムは最終的に、終了 (制御された) ジョブを終了 (即時) ジョブに変更し、ジョブに関するすべての処理を強制的に停止します。

コード	説明 / 処置
0200	<p>説明: 入力操作が正常に行われた際に、終了 (制御された) ジョブ要求が保留中であるという指示を受け取りました。さらに、0200 はなんらかのデータを受信したことを示しています。</p> <p>処置: プログラムを継続できます。ただし、推奨される処置は、表示処理を完了してプログラムを終了することです。これは、システムが最終的にジョブをキャンセルし、ジョブに関するすべての処理が停止されるためです。</p>

メジャー・コード 03

メジャー・コード 03- 入力操作は正常に完了したが、データを受信しなかった。

説明: 入力操作は正常に完了しましたが、データを受信しませんでした。

処置: 追加情報のマイナー戻りコードを検査し、次の操作を継続します。

コード	説明 / 処置
0300	<p>説明: 入力操作が正常に行われた際に、データを受信されませんでした。この原因となる条件の例として、サブファイル・レコード様式に対する <code>get-relative</code> 操作または <code>get-next-changed</code> 操作で使用可能なデータがないことが挙げられます。</p> <p>処置: 任意の適切な処理を継続します。たとえば、プログラムがサブファイル・レコード様式に対して <code>get-next-changed</code> 操作を発行した場合、0300 は、これ以上ユーザーによって変更されるサブファイル・レコードがないこと、およびこれ以上処理するユーザー入力データがないことを示します。</p> <p>メッセージ:</p> <ul style="list-style-type: none">CPF5017 (通知)CPF5020 (通知)CPF5037 (通知)
0309	<p>説明: プログラムが終了中 (制御された) です。データは受信されませんでした。この戻りコードは、送信勧誘された装置からの読み取り操作にのみ適用されます。</p> <p>処置: プログラムは処理を継続できます。ただし、推奨される処置は、表示処理を完了してプログラムを終了することです。これは、システムが最終的にジョブをキャンセルし、ジョブに関するすべての処理が停止されるためです。</p> <p>メッセージ:</p> <ul style="list-style-type: none">CPF4741 (通知)

0310 **説明:** ディスプレイ・ファイル用の WAITRCD 値によって指定された時間間隔が終了しました。

この戻りコードは、送信勧誘された装置からの読み取り操作にのみ適用されます。

注: この操作の完了に関連した装置はないため、I/O フィードバック域にある装置名には *N が入ります。

処置: 指定された時間間隔の終了後に、対象とする機能を実行する操作を発行します。たとえば、時間間隔を使用してデータを待機する時間の長さを制御する場合、送信勧誘された装置からの読み取り操作を別途発行してデータを受信することができます。

メッセージ:

CPF4742 (状況)

CPF4743 (状況)

メジャー・コード 04

メジャー・コード 04- 出力例外が発生。

説明: ディスプレイによってすでに送信されたデータを受信するはずであったのに、プログラムが出力を送信しようとしたため、出力例外が発生しました。この出力操作に関連したデータはディスプレイに送信されませんでした。プログラムはその出力を後で送信できます。

処置: 入力操作を発行してデータを受信します。

コード **説明 / 処置**

0412 **説明:** ディスプレイによってすでに送信されたデータを受信するはずであったのに、プログラムがデータを送信しようとしたため、出力例外が発生しました。プログラム・データは送信されませんでした。ディスプレイからのデータを受信した後に送信する必要があります。

処置: 入力操作を発行してデータを受信します。

メッセージ:

CPF4737 (通知)

メジャー・コード 08-11

メジャー・コード 08 および 11- 各種プログラム・エラーが発生。

説明: プログラムによる直前の操作が正常に完了しませんでした。操作は、間違った時点で発行されたため失敗した可能性があります。

処置: 適切なりカバーリ処置については、個別の戻りコードの説明を参照してください。

コード **説明 / 処置**

0800 **説明:** 実行されたばかりの獲得操作は正常に完了しませんでした。すでに獲得されていた装置を獲得しようとする試行がなされました。

処置: 元の獲得操作によって要求された表示装置が必要とされていたものである場合、プログラムは、それがすでに使用可能であるため、その装置の使用を開始できます。異なる装置が獲得された場合、異なる装置名の別の獲得処理を発行します。

メッセージ:

CPD4077 (診断)

CPF50A0 (状況)

1100

説明: 送信勧誘された装置がなかったため、送信勧誘された装置からの読み取り操作は正常に実行されませんでした。

処置: 送信勧誘機能の後に、送信勧誘された装置からの読み取り操作を発行します。

メッセージ:

CPF4740 (通知)

メジャー・コード 34

メジャー・コード 34- 入力例外が発生。

説明: プログラムによって試みられた入力操作が正常に完了しませんでした。受信したデータが、入力操作で指定されたレコード様式に対して長すぎます。

処置: 適切なりカバーリ処置については、個別の戻りコードの説明を参照してください。

コード

説明 / 処置

3431

説明: プログラムによって発行された入力操作は、ディスプレイから受信したデータの長さがユーザー定義のデータ・ストリームで指定された受信データの長さを超えるため、正常に完了しませんでした。受信されたデータは切り捨てられます。

この戻りコードは、ユーザー定義のデータ・ストリーム (USRDFN DDS キーワード) を指定するレコード様式を使用した入力操作にのみ適用されます。

処置: 装置ファイルをクローズして、プログラムを終了します。続いて、プログラムを変更し、入力レコードの長さが少なくとも受信するデータ・レコードと同じ長さになるようにします。

メッセージ:

CPF5062 (通知)

メジャー・コード 80

メジャー・コード 80- 永続システムまたはファイル・エラー (回復不能)。

説明: 回復不能なファイルまたはシステム・エラーが発生しました。エラーの原因が訂正されるまで、回復の見込みはありません。

処置: 以下に、それぞれの 80xx 戻りコードに対してプログラム側で取ることができる一般的な処置を示します。他の特定の処置については、それぞれの戻りコードの説明で示されています。

- ディスプレイなしで処理を続ける。
- 装置ファイルをクローズしてから、再びオープンする。それでも操作が失敗する場合は、一定の回数だけ操作を再試行してください。(回数はプログラム側で指定しなければなりません。)
- 終了する。

コード 説明 / 処置

8081 **説明:** システム・エラー条件が検出されたため、操作は正常に実行されませんでした。

処置: 表示装置をオフに構成変更してから、再びオンに構成変更する必要があるかもしれません。プログラムでは、次のいずれかを実行することができます。

- 表示装置なしで処理を続ける。
- 装置ファイルをクローズしてから、再びオープンする。
- 終了する。

メッセージ:

CPF4182 (エスケープ) CPF5416 (エスケープ)

CPF4510 (エスケープ) CPF5418 (エスケープ)

CPF5192 (エスケープ) CPF5423 (エスケープ)

CPF5257 (エスケープ) CPF5429 (エスケープ)

CPF5403 (エスケープ) CPF5431 (エスケープ)

CPF5404 (エスケープ) CPF5433 (エスケープ)

CPF5405 (エスケープ) CPF5434 (エスケープ)

CPF5408 (エスケープ) CPF5441 (エスケープ)

CPF5409 (エスケープ) CPF5447 (エスケープ)

CPF5410 (エスケープ) CPF5455 (エスケープ)

CPF5411 (エスケープ) CPF5456 (エスケープ)

CPF5414 (エスケープ) CPF5507 (エスケープ)

CPF5415 (エスケープ)

8082 **説明:** 表示装置が使用できないため、操作は正常に実行されませんでした。これは、ディスプレイのエラー回復メッセージに対して取り消し応答がされたか、あるいはディスプレイが通信装置の保留 (HLDCMNDEV) コマンドによって保留されているために発生することがあります。装置に対して操作を発行してはなりません。

処置: 表示装置がオンに構成変更された状態にリセットされるまで、ディスプレイとの通信は再開できません。装置が保留されている場合は、通信装置の解放 (RLSCMNDEV) コ

マンドを使用して装置をリセットしてください。装置がエラー状態の場合は、装置をオフにしてからもう一度オンにしてください。いったん装置がリセットされると、表示装置ファイルを再オープンして、通常の操作を開始できます。プログラムでは、次のいずれかを実行することができます。

- 表示装置なしで処理を続ける。
- 装置ファイルをクローズしてから、再びオープンする。
- 終了する。

メッセージ:

CPF4354 (エスケープ)

CPF5269 (エスケープ)

80A6

説明: クローズまたは解放操作の際に、システム・ネットワーク体系 (SNA) のアンバインド操作が正常に行われませんでした。これは、装置構成エラーの結果として起こる場合があります。装置が使用不能になっている可能性があります。装置に対して操作を発行してはなりません。

処置: 失敗の原因については、付随するエラー・メッセージの装置応答コードを参照してください。エラーをリセットするために、装置をオフに構成変更し、再びオンにします。エラーを訂正し、プログラムを再度試行してください。

メッセージ:

CPF4527 (エスケープ)

80B3

説明: ディスプレイ・ファイルが使用できないため、オープン操作は正常に実行されませんでした。ファイルは、必要なリソースが使用可能になるまで再オープンできません。

処置: プログラムでは、そのディスプレイ・ファイルが使用可能になるのを待ってから、別のオープン操作を発行することができます。あるいは、他の処理を継続するか、またはプログラムを終了することができます。

表示ファイル変更 (CHGDSPF) コマンドまたは表示装置ファイル一時変更 (OVRDSPF) コマンドの WAITFILE パラメーターの値を大きくすることによって、ファイルが使用可能になるまで待つ時間を長くすることを検討してください。

メッセージ:

CPF4128 (エスケープ)

80C0

説明: 表示装置で回復不能なエラーが発生しました。

処置: 表示装置をオフに構成変更してから、再びオンに構成変更する必要があるかもしれません。プログラムでは、次のいずれかを実行することができます。

- ディスプレイ装置なしで処理を続ける。
- 装置ファイルをクローズしてから、再びオープンする。
- 終了する。

メッセージ:

CPF5103 (エスケープ)

CPF5420 (エスケープ)

CPF5192 (エスケープ)

CPF5421 (エスケープ)

CPF5412 (エスケープ)

CPF5430 (エスケープ)

CPF5413 (エスケープ)

CPF5437 (エスケープ)

CPF5419 (エスケープ) CPF5439 (エスケープ)

80EB

説明: プログラム、ディスプレイ・ファイル、またはオーバーライド・コマンドで、無効なオープン・オプションまたは無効なオプションの組み合わせが指定されたので、オープン操作は正常に実行されませんでした。

処置: ディスプレイ・ファイルをクローズし、問題を訂正してから、ファイルを再びオープンしてください。個々のメッセージを参照して、どのオプションが正しくないのかを判定してください。

メッセージ:

CPF4062 (エスケープ) CPF4345 (エスケープ)

CPF4129 (エスケープ) CPF5151 (エスケープ)

CPF4148 (エスケープ) CPF5510 (エスケープ)

CPF4156 (エスケープ) CPF5511 (エスケープ)

CPF4163 (エスケープ) CPF5512 (エスケープ)

CPF4169 (エスケープ) CPF5513 (エスケープ)

CPF4191 (エスケープ) CPF5552 (エスケープ)

CPF4238 (エスケープ)

80ED

説明: プログラムのコンパイル以降に、ファイルのレコード様式記述が変更されているので、オープン操作は正常に実行されませんでした。

処置: ファイルをクローズし、プログラムを終了してください。変更がアプリケーション・プログラムに影響するどうかを判別します。影響があれば、プログラムを再コンパイルします。プログラムに影響がない場合は、ファイルを LVLCHK(*NO) に変更またはオーバーライドしなければなりません。LVLCHK(*NO) が指定されると、システムはレコード様式記述を比較しません。

メッセージ:

CPF4131 (エスケープ)

80F8

説明: ファイルにエラーのマークが付いているので、ファイルに対する操作は正常に実行されませんでした。

処置: ファイルをクローズしてください。ジョブ・ログにあるメッセージを参照して、どのようなエラーが発生したかを調べてください。それらのエラーに対して、適切な処置を取ってください。

メッセージ:

CPF4132 (エスケープ) CPF5129 (エスケープ)

CPF4213 (エスケープ) CPF5144 (エスケープ)

CPF4550 (エスケープ) CPF5427 (エスケープ)

メジャー・コード 81

メジャー・コード 81- 永続装置エラー (回復不能)。

説明: 入出力操作において回復不能な装置関連のエラーが発生しました。問題の原因が見つかって解決されるまで、この表示装置の使用を続けようとする試みはうまくいきませんが、他の表示装置に送られた、ファイルに関連した操作は作動するものと予期されます。

処置: 以下に、それぞれの 81xx 戻りコードに対してプログラムで取ることができる一般的な処置を示します。他の特定の処置については、それぞれの戻りコードの説明で示されています。

- 表示装置なしで処理を続ける。
- 装置を解放するかファイルをクローズし、問題を訂正してから、装置を再び獲得するかファイルをオープンする。それでも操作が失敗する場合は、一定の回数だけ操作を再試行してください。(回数はプログラム側で指定しなければなりません。)
- 終了する。

いくつかの戻りコードは、装置をオフに構成変更し、再びオンに構成変更することによって、エラー条件を訂正しなければならないことを指示します。

コード 説明 / 処置

8181 **説明:** 装置への入出力操作中に、システム・エラー条件が検出されました。

処置: エラーのある装置を解放するか、ファイルをクローズします。エラーを消去するために、装置をオフに構成変更し、再びオンにすることが必要な場合があります。関連するメッセージから、障害の原因を調べてください。それ以外の修正処置を指示するシステム・オペレーター・メッセージがないかどうかを調べてください。ファイルを再びオープンするか、装置を獲得して継続ください。

メッセージ:

CPF4553 (エスケープ)
CPF4725 (エスケープ)
CPF5105 (エスケープ)
CPF5189 (エスケープ)
CPF5254 (エスケープ)

8191 **説明:** 永続的回線エラーが発生したため、操作は正常に実行されず、システム・オペレーターが回線エラー・メッセージに答えて、回復オプションを選択しました。(システム・オペレーターに尋ねれば、発生した回線エラーのタイプを知ることができます。) 装置には、使用不能のマークが付けられます。

処置: エラーのある装置を解放するか、ファイルをクローズします。エラーを消去するために、装置をオフに構成変更し、再びオンにします。ファイルを再びオープンするか、装置を獲得して継続ください。

メッセージ:

CPF4526 (エスケープ)
CPF4542 (エスケープ)
CPF4551 (エスケープ)
CPF5128 (エスケープ)
CPF5143 (エスケープ)

CPF5198 (エスケープ)

8197

説明: 回復不能なエラー条件が装置で検出されました。こうしたエラーの例としては、ユーザーが表示装置をオフにしています。

処置: エラーのある装置を解放するか、ファイルをクローズします。エラーをクリアするには、表示装置をオフに構成変更してから、再びオンに構成変更する必要があるかもしれません。検出された特定のエラーの原因に関する詳細については、付随するエラー・メッセージを参照してください。ファイルをオープンするか、再び装置を獲得して継続します。

メッセージ:

CPF4149 (エスケープ) CPF5106 (エスケープ)

CPF4197 (エスケープ) CPF5140 (エスケープ)

CPF4524 (エスケープ) CPF5143 (エスケープ)

CPF4533 (エスケープ) CPF5199 (エスケープ)

CPF4538 (エスケープ) CPF5265 (エスケープ)

CPF5047 (エスケープ)

81C2

説明: ディスプレイについてシステム・ネットワーク体系 (SNA) セッションが活動状態でないため、プログラムによって発行された操作は正常に実行されませんでした。

処置: 装置を解放するか、ファイルをクローズしてください。エラーを消去するために、装置をオフに構成変更し、再びオンにします。ファイルをオープンするか、再び装置を獲得して継続します。

メッセージ:

CPF5170 (エスケープ)

CPF5422 (エスケープ)

メジャー・コード 82

メジャー・コード 82- オープンまたは獲得操作が失敗。

説明: ディスプレイ・ファイルのオープンまたは表示装置の獲得が正常に実行されませんでした。エラーは、回復可能な場合と永続的な場合とがありますが、特定の表示装置に限定されています。エラーの原因が訂正されるまで、回復の見込みはありません。

処置: 以下に、それぞれの 82xx 戻りコードに対して取ることができる一般的な処置を示します。他の特定の処置については、それぞれの戻りコードの説明で示されています。次のいずれかが可能です。

- 印刷装置なしで処理を続ける。
- 装置を解放するかファイルをクローズし、問題を訂正してから、再び装置を獲得するかファイルをオープンする。その時点で装置が (他の出力のため) 使用中であったというような、一時的な状況でエラーが発生した場合は、後続の操作は正常に実行されることがあります。

それでも操作が失敗する場合は、一定の回数だけ操作を再試行してください。(回数はプログラム側で指定しなければなりません。)

- 試行された操作が獲得操作であった場合、装置を解放して獲得操作を再び発行します。
- 試行された操作がオープン操作であった場合、ファイルをクローズしてオープン操作を再び発行します。

- 終了する。

いくつかの戻りコードでは、ファイルの値を変更して、エラー条件を訂正しなければならないことが指示されています。ファイルのパラメータ値を変更するには、表示ファイル変更 (CHGDSPF) コマンド、または表示装置ファイル一時変更 (OVRDSPF) コマンドを使用します。

コード

説明 / 処置

8281

説明: オープン操作または獲得操作でシステム・エラー条件が検出されました。ファイルが以前にエラー状態であったか、またはシステム・エラーのためにファイルがオープンできなかった可能性があります。

処置: 関連するメッセージから、障害の原因を調べてください。それ以外の修正処置を指示するシステム・オペレーター・メッセージがないかどうかを調べてください。

エラーをクリアするには、表示装置をオフに構成変更してから、再びオンに構成変更する必要があるかもしれません。プログラムでは、次のいずれかを実行することができます。

- 表示装置なしで処理を続ける。
- 装置を解放するかファイルをクローズし、問題を訂正してから、再び装置を獲得するかファイルをオープンする。
- 終了する。

メッセージ:

CPF4168 (エスケープ)	CPF5410 (エスケープ)
CPF4182 (エスケープ)	CPF5411 (エスケープ)
CPF4221 (エスケープ)	CPF5424 (エスケープ)
CPF5105 (エスケープ)	CPF5447 (エスケープ)
CPF5254 (エスケープ)	CPF5455 (エスケープ)
CPF5257 (エスケープ)	CPF919E (エスケープ)

8282

説明: 表示装置が使用できないため、オープン操作または獲得操作は正常に実行されませんでした。これは、ディスプレイのエラー回復メッセージに対して取り消し応答がされたか、あるいはディスプレイが通信装置の保留 (HLDCMNDEV) コマンドによって保留されているために発生することがあります。装置に対して操作を発行してはなりません。装置が、オンに構成変更された状態にリセットされるまで、ディスプレイとの通信は再開できません。

処置: 表示装置ファイルをクローズしてください。装置が保留されている場合は、通信装置の解放 (RLSCMNDEV) コマンドを使用して装置をリセットしてください。装置がエラー状態の場合は、装置をオフにしてからもう一度オンにしてください。いったん装置をリセットして、表示装置ファイルを再オープンし、通常の操作を開始します。

メッセージ:

CPF4171 (エスケープ)

CPF4354 (エスケープ)

CPF5548 (エスケープ)

8291

説明: オープン操作または獲得操作で、永続的な回線エラーが発生しました。装置には、使用不能のマークが付けられます。

処置: エラーのある装置を解放するか、ファイルをクローズします。エラーを消去するために、装置をオフに構成変更し、再びオンにします。ファイルを再オープンして、継続してください。

メッセージ:

CPF4146 (エスケープ)

CPF4179 (エスケープ)

CPF4193 (エスケープ)

CPF4291 (エスケープ)

CPF5198 (エスケープ)

CPF5260 (エスケープ)

8297

説明: 回復不能なエラー条件が表示装置で検出されたため、オープン操作または獲得操作が異常終了しました。こうしたエラーの例としては、ユーザーが表示装置をオフにしています。

処置: エラーのある装置を解放するか、ファイルをクローズします。エラーをクリアするには、表示装置をオフに構成変更してから、再びオンに構成変更する必要があるかもしれません。検出された特定のエラーの原因に関する詳細については、付随するエラー・メッセージを参照してください。ファイルをオープンするか、再び装置を獲得して継続します。

メッセージ:

CPF4192 (エスケープ)

CPF5047 (エスケープ)

CPF5106 (エスケープ)

CPF5140 (エスケープ)

CPF5143 (エスケープ)

CPF5199 (エスケープ)

- 82A6** **説明:** システム・ネットワーク体系 (SNA) バインド・コマンドが正常に実行されなかったため、オープン操作または獲得操作は失敗しました。
- 処置:** プログラムが通信している表示装置が、適切に構成されていることを確認してください。検出された特定のエラーに関する詳細については、付随するエラー・メッセージの装置応答コードを参照してください。
- メッセージ:**
- CPF4124 (エスケープ)
 - CPF4190 (エスケープ)
 - CPF5103 (エスケープ)
 - CPF5517 (エスケープ)
- 82A8** **説明:** ディスプレイ・ファイルに許可された装置の最大数に達したため、獲得操作は正常に実行されませんでした。
- 処置:** 異なる装置を解放して、獲得操作を再び実行することによって、プログラムを回復することができます。より多くの装置が必要な場合には、ファイルをクローズし、ディスプレイ・ファイルの MAXDEV 値を増やしてください。
- メッセージ:**
- CPF4745 (エスケープ)
 - CPF5041 (状況)
 - CPD4757 (診断)
- 82A9** **説明:** 要求中のプログラム装置が使用できないため、獲得操作は正常に実行されませんでした。要求中のプログラム装置は、プログラムが対話式ジョブで実行していないため、使用できない可能性があります。
- 処置:** プログラムでは、ディスプレイなしで続けるか、別の表示装置の使用を試みるか、または終了することができます。
- プログラムが要求中のプログラム装置を使用しなければならない場合、それが対話式ジョブで実行していることを確認してください。
- メッセージ:**
- CPF4366 (エスケープ)
 - CPF5381 (エスケープ)
- 82AA** **説明:** 表示装置記述が見つからなかったため、オープン操作または獲得操作は正常に実行されませんでした。
- 処置:** プログラムでは、ディスプレイなしで続けるか、別の表示装置の使用を試みるか、または終了することができます。
- 表示装置の名前が、CRTDSPF、CHGDSPF、CRTDEVDSF、または OVRDSPF コマンドの DEV パラメーターに、正しく指定されていることを確認してください。
- メッセージ:**
- CPF4103 (エスケープ)
 - CPF4747 (エスケープ)
- 82AB** **説明:** 表示装置がオンに構成変更されていないため、オープン操作または獲得操作は正常に実行されませんでした。

処置: プログラムでは、ディスプレイなしで続けるか、別の表示装置の獲得を試みるか、または終了することができます。表示装置をオンに構成変更し、オープン操作または獲得操作を再び試行します。

プログラムで同じディスプレイを継続したい場合、装置を解放するかファイルをクローズし、問題を訂正してから、再び装置を獲得するかファイルをオープンします。

メッセージ:

CPF4285 (エスケープ)

CPF5333 (エスケープ)

82B3

説明: 獲得中の表示装置が別の処理で使用中的であるため、オープン操作または獲得操作は正常に実行されませんでした。

処置: 表示装置が使用可能になるのを待ってから、獲得操作を再び発行します。あるいは、ディスプレイなしで他の処理を継続するか、またはプログラムを終了することができます。

構成状況の処理 (WRKCFGSTS) コマンドを使用して、表示装置を使用中のジョブを判別することができます。

CHGDSPF または OVRDSPF コマンドの WAITFILE パラメーターを増やして、装置が使用可能になるまで待つ時間を長くすることを検討してください。

メッセージ:

CPF4109 (エスケープ)

CPF4130 (エスケープ)

CPF4282 (エスケープ)

CPF5217 (エスケープ)

CPF5332 (エスケープ)

82EE

説明: ディスプレイ・ファイル用にサポートされていない装置に対して、オープン操作または獲得操作が試みられました。

プログラムは無効な表示装置を獲得しようとしています。あるいは、要求中のプログラム装置を獲得しようとしています。ジョブ用に要求中のプログラム装置は、表示装置ではなく通信装置です。

処置: プログラムでは、ディスプレイなしで続行させるか、異なる表示装置を獲得するか、あるいはファイルをクローズして終了させるかのいずれかを実行できます。

表示装置の名前が、CHGDSPF、CRTDEV DSP、CRTDSPF、または OVRDSPF コマンドに正しく指定されていることを確認してください。

プログラムが、要求中のプログラム装置を獲得しようとしていた場合、プログラムが対話式ジョブで実行しており、要求中のプログラム装置が表示装置であることを確認してください。

メッセージ:

CPF4105 (エスケープ)

CPF4223 (エスケープ)

CPF4760 (エスケープ)

CPF5038 (エスケープ)

- 82EF** **説明:** ユーザーに許可されていない装置か、または保守モードの装置に対してオープン操作または獲得操作が試みられました。
- 処置:** プログラムでは、ディスプレイなしで続けるか、別の表示装置の獲得を試みるか、または終了することができます。
- 操作がオープン操作であった場合、ファイルをクローズし、問題を訂正してから、オープン操作を再び発行してください。操作が獲得操作であった場合、問題を訂正してから、獲得操作を再び発行します。
- 権限エラーの場合は、セキュリティー担当者または装置の所有者から、装置に対する権限を取得してください。装置が保守モードの場合は、システム保守ツール (SST) 機能が装置を現在使用しています。装置が使用可能になるまで待つて、操作を再び発行してください。
- メッセージ:**
- CPF4104 (エスケープ)
 - CPF4186 (エスケープ)
 - CPF5278 (エスケープ)
 - CPF5279 (エスケープ)
- 82F8** **説明:** プログラムは、入出力操作または獲得操作中の直前のエラーのために、誤ってマークされた装置に対する獲得操作を試みました。
- 処置:** 装置を解放するかファイルをクローズし、前の問題を訂正してから、再び装置を獲得するかファイルをオープンしてください。
- メッセージ:**
- CPF5293 (エスケープ)

メジャー・コード 83

メジャー・コード 83- 装置エラーが発生 (回復可能)。

説明: 入出力操作中にエラーが発生しましたが、表示装置はまだ使用できます。プログラム内での回復が可能かもしれません。

処置: 以下に、それぞれの 83xx 戻りコードに対して取ることができる一般的な処置を示します。他の特定の処置については、それぞれの戻りコードの説明で示されています。

- 表示装置なしで処理を続ける。
- 問題を訂正し、表示装置で処理を続ける。操作が正常に回復されない場合は、一定の回数だけ操作を再試行してください。(回数はプログラム側で指定しなければなりません。)
- 終了する。

いくつかの戻りコードでは、ディスプレイ・ファイルの値を変更して、エラー条件を訂正しなければならないことが指示されています。ファイルのパラメーター値を変更するには、表示ファイル変更 (CHGDSPF) コマンド、または表示装置ファイル一時変更 (OVRDSPF) コマンドを使用します。

- | コード | 説明 / 処置 |
|-------------|--|
| 830B | 説明: プログラムは、装置の獲得前または装置の解放後に、入出力操作を発行しようとした。 |

CPF5051 (通知)

831F

説明: 操作について、無効な長さが指定されました。

出力操作でプログラムは、表示装置で許可されている最大レコード長を超える長さのデータ・レコードを送信しようとしていました。データは切り捨てられています。

処置: 出力長を小さくして、出力操作を再び発行してください。非フィールド・レベルのディスプレイ・ファイルのレコード長は、ディスプレイ・サイズを超えてはなりません。どのディスプレイ・ファイルのレコード長も、32 763 文字を超えてはなりません。

メッセージ:

CPF4010 (診断)

CPF4078 (診断)

8322

説明: 試みられた操作は現在の状態では無効です。プログラムが送信状態ではない時に書き込み操作が試みられたか、サブファイルがアクティブではない時にサブファイル操作が試みられました。

処置: プログラムでは、無効な操作を迂回して続行するか、またはファイルをクローズして終了することができます。プログラムでの操作の順序を訂正してから、ジョブを再び実行してください。

メッセージ:

CPF5013 (通知)

CPF5060 (通知)

832D

説明: 送信勧誘操作が未処理のときにプログラムが無効な操作を試みました。送信勧誘操作または即時ゲット操作を発行すると、最初の送信勧誘が読み取り操作または送信勧誘された装置からの読み取り操作によって完了するまで、別の送信勧誘操作を同じ表示装置で発行することはできません。

処置: 入力操作を発行して、別の送信勧誘操作を発行する前に送信勧誘されたデータを受信します。あるいは、ファイルをクローズして終了します。プログラムのコーディング・エラーがこのエラーの原因となっている場合、プログラムにおける操作の順序を訂正してください。

メッセージ:

CPF5052 (通知)

8343

説明: サブファイルがいっぱいになった後、別のレコードをサブファイルに追加しようとしていました。

処置: プログラムでサブファイルをクリアするか、レコードをこれ以上サブファイルに追加せずに続けることができます。あるいは、ファイルをクローズして終了します。

DDS ステートメント内でサブファイルの長さを増やします。プログラムのコーディング・エラーがこのエラーの原因となっている場合、プログラムを訂正してください。

メッセージ:

CPF5043 (通知)

83E0

説明: プログラムが、表示装置ファイルで定義されていないレコード様式、またはレコード様式名が省略されたレコード様式を使用する操作を発行しました。

処置: プログラムにおけるレコード様式の名前を調べて、それが正しいことを確認してください。次に、ファイルの DDS で、レコード様式が適切に定義されていることを確認してください。

メッセージ:

CPF5022 (通知)

CPF5023 (通知)

CPF5053 (通知)

CPF5054 (通知)

83E1 **説明:** 入出力操作中にエラーが発生しました。要求中のプログラム装置がプログラムを自動的に切断するよう設定されています。同じユーザーが同じ表示装置にサインオンしたため、プログラムは再び開始されました。エラーが発生した際に存在していたデータがディスプレイから消去されました。

処置: プログラムの通常の開始点に分岐して、ディスプレイを再書き込みします。エラーが発生した際にアクティブになっていたディスプレイ・ファイルでクローズまたはオープン操作を実行する必要はありません。

メッセージ:

CPF509F (通知)

83E8 **説明:** 未処理の即時読み取り操作がないときに、制御言語プログラムは受け取り終了 (ENDRCV) コマンドを発行しました。

処置: プログラムは、出力操作を発行して送信を続行するか、入力操作を発行して受信を開始するか、またはファイルをクローズして終了することができます。障害が発生した操作を再試行する前に、プログラムのエラーを訂正してください。

メッセージ:

CPF4910 (通知)

83F6 **説明:** プログラムが無効なデータをディスプレイに送信しました。データ・タイプが、使用されているフィールドにとって正しくない可能性があります。

処置: プログラムにおけるレコード様式の名前を調べて、それが正しいことを確認してください。プログラム内のデータ定義ステートメントが、ファイルの DDS で定義された出力レコードと一致することを確認してください。障害が発生した操作を再試行する前に、プログラムのエラーを訂正してください。

メッセージ:

CPF5063 (通知)

CPF5216 (エスケープ)

CPF5301 (エスケープ)

83F8 **説明:** プログラムは、入出力操作または獲得操作中の直前のエラーのために、誤ってマークされた装置に対する入出力操作を試みました。

処置: 装置を解放するかファイルをクローズし、前の問題を訂正してから、再び装置を獲得するかファイルをオープンしてください。

メッセージ:

CPF5293 (エスケープ)

付録 E. 編集コード

i5/OS 編集コード

以下の表は、i5/OS 編集コードが提供する機能について要約しています。

表 101. i5/OS 編集コードの要約表

編集コード	コンマ ¹ の表示	小数点 ¹ の表示	負の値の際に表示される符号	QDECFMT システム値のブランク値	QDECFMT システム値の I 値	QDECFMT システム値の J 値	先行ゼロの抑制
1	あり	あり	符号なし	.00 または 0	.00 または 0	0.00 または 0	あり
2	あり	あり	符号なし	ブランク	ブランク	ブランク	あり
3		あり	符号なし	.00 または 0	.00 または 0	0.00 または 0	あり
4		あり	符号なし	ブランク	ブランク	ブランク	あり
A	あり	あり	CR	.00 または 0	.00 または 0	0.00 または 0	あり
B	あり	あり	CR	ブランク	ブランク	ブランク	あり
C		あり	CR	.00 または 0	.00 または 0	0.00 または 0	あり
D		あり	CR	ブランク	ブランク	ブランク	あり
J	あり	あり	- (負符号)	.00 または 0	.00 または 0	0.00 または 0	あり
K	あり	あり	- (負符号)	ブランク	ブランク	ブランク	あり
L		あり	- (負符号)	.00 または 0	.00 または 0	0.00 または 0	あり
M		あり	- (負符号)	ブランク	ブランク	ブランク	あり
N	あり	あり	- (負符号)	.00 または 0	.00 または 0	0.00 または 0	あり
O	あり	あり	- (負符号)	ブランク	ブランク	ブランク	あり
P		あり	- (負符号)	.00 または 0	.00 または 0	0.00 または 0	あり
Q		あり	- (負符号)	ブランク	ブランク	ブランク	あり
Y ²							あり
Z ³							あり

表 101. i5/OS 編集コードの要約表 (続き)

編集コード	コンマ ¹ の表示	小数点 ¹ の表示	負の値の際に表示される符号	QDECFMT システム値のブランク値	QDECFMT システム値の I 値	QDECFMT システム値の J 値	先行ゼロの抑制
注:							
¹	QDECFMT システム値は、小数点文字 (米国ではピリオドを使用)、3 桁のグループを区切るのに使用する文字 (米国ではコンマを使用)、およびゼロ抑制のタイプ (コンマとピリオドの配置に応じて異なる) を決定します。QDECFMT システム値について詳しくは、「AS/400e シリーズ 実行管理の手引き」を参照してください。						
²	Y 編集コードは、3 桁から 6 桁の長さの日付フィールドの左端のゼロを抑制し、7 単位分の位置の長さのフィールドの左端から 2 個のゼロを抑制します。また、Y 編集コードは、以下のパターンに従って月、日、および年の間にスラッシュ (/) を挿入します。 nn/n nn/nn nn/nn/n nn/nn/nn nnn/nn/nn EDTCDE(Y) を指定して DATE キーワードを指定すると、区切り記号の文字としてジョブ属性 DATSEP が使用されます。デフォルトの DATSEP はスラッシュ (/) です。ファイルの作成時に DATFMT が JUL (ユリウス) の場合、日付は通常 nn/nnn としてフォーマットされ、EDTCDE(Y) は無効になります。						
³	Z 編集コードは、数字フィールドから符号 (正および負) を除去します。フィールドが書き込まれる前に、単位位置の符号は 16 進数の F に変更されます。						

注: 編集コード X を指定できますが、ディスプレイ・ファイルに対しては無効であり、無視されます。システムは優先される符号 F を使用して操作しますが、この操作は編集コード X を使用することと同等です。編集コード X を使用すると、ブランクのキーボード・シフト (位置 35) がデフォルトの数値専用 (属性 Y) になりますが、ディスプレイ・ファイルに対する他の影響はなく、他のすべての処理においてこのコードは無視されます。フィールドが表示される長さは、編集コード X ではなくキーボード・シフトによって決まります (デフォルトの数値専用 Y 属性により、10 進数のフィールドに位置 1 が追加されることがあります)。

i5/OS 編集コードを使用した編集の例

以下の表には、有効な編集コードと、未編集のソース・データおよび編集済みの出力の例が示されています。ゼロ抑制と小数点文字は、システム値 QDECFMT によって決まります。日付区切り文字は、ジョブ属性 DATSEP によって決まります。この図では、QDECFMT は x であると想定され、DATSEP は / であると想定されます。

表 102. 有効な編集コード、ソース・データ、および編集済みの出力

編集コード	正数 - 小数点以下の桁数 2	正数 - 小数点以下の桁数なし	負の数値 - 小数点以下の桁数 3 ¹	負の数値 - 小数点以下の桁数なし ¹	ゼロ・バランス - 小数点以下の桁数 2 ¹	ゼロ・バランス - 小数点以下の桁数なし ¹
未編集	1234567	1234567	xxxx.125-	xxxx.125-	xxxxxx	xxxxxx
1	12,345.67	1,234,567	.125	125	.00	0
2	12,345.67	1,234,567	.125	125		
3	12345.67	1234567	.125	125	.00	0
4	12345.67	1234567	.125	125		
A	12,345.67	1,234,567	.125CR	125CR	.00	0
B	12,345.67	1,234,567	.125CR	125CR		
C	12345.67	1234567	.125CR	125CR	.00	0
D	12345.67	1234567	.125CR	125CR		
J	12,345.67	1,234,567	.125-	125-	.00	0
K	12,345.67	1,234,567	.125-	125-		
L	12345.67	1234567	.125-	125-	.00	0
M	12345.67	1234567	.125-	125-		
N	12,345.67	1,234,567	-.125	-125	.00	0
O	12,345.67	1,234,567	-.125	-125		
P	12345.67	1234567	-.125	-125	.00	0
Q	12345.67	1234567	-.125	-125		
Y ²	123/45/67	123/45/67	0/01/25	0/01/25	0/00/00	0/00/00
Z ³	1234567	1234567	125	125		

注:

¹ x はブランクを表します。

² Y 編集コードは、3 桁から 6 桁の長さの日付フィールドの左端のゼロを抑制し、7 単位分の位置の長さのフィールドの左端から 2 個のゼロを抑制します。詳しくは、757 ページの表 101 の 2 番目の脚注を参照してください。

³ Z 編集コードは、符号 (正または負) を除去し、先行ゼロを抑制します。

ユーザー定義編集コード

5 つの編集コードを定義して、i5/OS 編集コードで使用できる機能よりも多くの編集機能を提供し、それが無い場合には編集語を使用する必要がある共通した編集機能进行处理できます。

編集記述作成 (CRTEDTD) コマンドを使用して独自の編集コードを定義できます。

表 103. IBM 提供の編集記述

説明	QEDIT5 編集コード ¹	QEDIT6 編集コード ¹	QEDIT7 編集コード ¹	QEDIT8 編集コード ¹	QEDIT9 編集コード ¹
整数マスク	xxx,xxx,xxx,xxx,xx0	xxx,xxx,xxx,xxx,xx0	xxx,xxx,xxx,xxx,x0x	xxx,xxx,xxx,xxx,x0x	0xx-xx-xx
小数点	. (ピリオド)	. (ピリオド)	. (ピリオド)	. (ピリオド)	. (ピリオド)
小数マスク	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx	なし
充てん文字	x	x	x	x	x
浮動通貨記号	なし	なし	なし	\$	なし
ゼロ・バランス	充てん文字に置き換え	充てん文字に置き換え	通常の編集規則	通常の編集規則	通常の編集規則

表 103. IBM 提供の編集記述 (続き)

説明	QEDIT5 編集コード ¹	QEDIT6 編集コード ¹	QEDIT7 編集コード ¹	QEDIT8 編集コード ¹	QEDIT9 編集コード ¹
負の状況	CR	- (負符号)	- (負符号)	- (負符号)	なし
正の状況	DR	なし	なし	なし	なし
左の定数	なし	なし	\$	なし	なし
右の定数	なし	x*	なし	なし	なし
:					
¹ x はブランクを表します。					

編集記述には、以下の項目が含まれています。

整数マスク フィールドの整数部分の編集について記述します。ブランク、ゼロ、およびアンパーサンド (&) 以外のすべての文字は定数として扱われます。

- ブランクは、ゼロ抑制が終了している場合はブランクが数字に置き換えられることを意味します。終了していない場合はブランクは充てん文字に置き換えられます。
- 左端のゼロは、ゼロが数字に置き換えられ、ゼロ抑制を終了することを意味します。その他のゼロはすべて定数として扱われます。
- アンパーサンドは、& がブランクに置き換えられることを意味します。

小数点 小数点として使用する文字を定義します。デフォルトでは、ピリオド (.) が使用されません。

小数マスク フィールドの小数部分の編集について記述します。アンパーサンドは、整数マスクの場合と同じです。ゼロはすべて定数として扱われ、ブランクはすべて数字に置き換えられます (充てん文字は使用されません)。

充てん文字 ゼロ抑制の結果の各位置に使用される文字を定義します。デフォルトでは、ブランクが使用されます。

浮動通貨記号 フィールドの編集に使用する浮動通貨記号を定義します。

ゼロ・バランス

ゼロ値を編集する方法を指定します。充てん文字か、または整数マスクと小数マスクを使用して編集できます。

負の状況 フィールドが負の場合に、フィールドの編集結果の後に続く文字を定義します。

正の状況 フィールドが正またはゼロの場合に、フィールドの編集結果の後に続く文字を定義します。

左の定数 フィールドの編集結果の左端の部分になる定数を定義します。

右の定数 フィールドの編集結果の右端の部分になる定数を定義します。

ユーザー定義編集コードの使用法

以下の規則は、編集記述の使用時に適用されます。これらの規則は、編集されるフィールドの長さや小数点以下の桁数の影響を受けます。

- 編集されるフィールドは、整数マスクと小数マスクによって位置合わせされます。
- 整数全体のマスクは常に使用されるわけではありません。整数マスクは、左端の有効数字または左端の (または置き換え) ゼロの直前で左側が切り捨てられます。
- 小数点は整数マスクの直後に、小数マスクは小数点の直後に続きます。小数点を使用しない場合は、小数マスクは整数マスクの直後になります。

- 小数全体のマスクは常に使用されるわけではありません。小数マスクは、右端の有効数字の直後で右側が切り捨てられます。
- 編集結果の幅は、以下の合計に等しくなります。
 - 左の定数の長さ
 - 浮動通貨記号の長さ
 - 切り捨てられる整数マスクの長さ
 - 小数点の長さ (小数点を使用される場合は常に 1)
 - 切り捨てられる小数マスクの長さ
 - 負または正の状況の値の長さ (長い方)

注: 指定した負の状況のテキストの長さが、正の状況のテキストと異なる場合、CRTEDTD コマンドは短い方の値にブランクを埋め込みます。したがって、短い値と長い値の長さが等しくなります。

- 右の定数の長さ
- 整数マスクまたは小数マスクに、編集されるフィールドにとって十分な数字の置換文字が含まれていない場合は、そのフィールドは編集されず、無視されます。

ユーザー定義編集コードの例

以下の編集記述作成 (CRTEDTD) コマンドは、数字フィールドを編集し、値がクレジットかデビットのどちらであるかを示す編集記述の作成方法を示します (x はブランクを示します)。

```
CRTEDTD  EDTD(5) INTMASK('xxx,xxx,xx')
          FRACMASK('xxxx') NEGSTS('DEBITx')
          POSSTS('CREDIT') LFTCNS('$')
          RGTCNS('xB**')
```

上記の編集記述を使用するフィールドには値 001234 が含まれており、2 つの小数点以下の桁数があります。編集されたフィールドは以下のようになります。

```
$xx12.34CREDITx**
```

編集記述を作成する際には、編集コードに関連した数値 (5、6、7、8、または 9) のみを指定することに注意してください。指定した数値に QEDIT がシステムにより自動的に添付されます。(上記の例では、EDTD(5) が指定されています。この数値は QEDIT5 になります。)

付録 F. System/36 と互換性のある画面データ管理機能

この付録では、i5/OS 画面データを使用して System/36 と互換性のある機能を提供する方法を説明します。System/36 の互換性についての追加情報は、iSeries Information Center の DDS のトピックに記載されています。

表示装置を使用する System/36 アプリケーションをマイグレーションするために、i5/OS 画面データ管理機能には、オペレーティング・システムが System/36 ワークステーション・データ管理機能のように作動することを可能にする機能があります。System/36 との互換性のレベルは、以下に依存しています。

- ユーザー画面管理 (USRDSMGT) キーワード

このキーワードは、\$SFGR ユーティリティまたは「System/36 ディスプレイ・ファイル作成」(CRTS36DSPF) CL コマンドが SFGR ソースを DDS ソースに変換する時に、自動的に DDS ソースに書き込まれます。このキーワードを指定すると、画面データ管理機能の代わりに、System/36 ワークステーション・データ管理機能が使用されることを示します。

たとえば、USRDSMGT キーワードは、System/36 の場合と同様にカーソルが配置されるように指定します。USRDSMGT キーワードが指定されていない場合、カーソルは画面データ管理規則に従って配置されます。

- プログラムを書くために使用されるプログラミング言語

アプリケーションのレベルで System/36 の互換性を確保するには、プログラムは、System/36 互換の RPG II または System/36 互換の COBOL で作成する必要があります。

たとえば、プログラムが System/36 互換の言語で書かれている場合、高水準言語で設定されるファイル状況コードおよびデータ管理機能により設定される戻りコードは、System/36 と同じです。プログラムが System/36 互換の言語で書かれていない場合、ファイル状況コードおよび戻りコードは i5/OS 値です。

- アプリケーションが実行される環境

環境レベルで System/36 との互換性を確保するには、プログラムを System/36 環境で実行する必要があります。

たとえば、System/36 環境で稼働中のアプリケーションが MSGID DDS キーワードを使用してユーザー・メッセージ・ファイルにあるメッセージを表示する場合、使用されるメッセージ・ファイルは、MEMBER OCL ステートメントの USER1 パラメーターで指定されたメッセージ・ファイルになります。この同じアプリケーションが System/36 環境で稼働していない場合には、使用されるメッセージ・ファイルは、ジョブのライブラリー・リストにある USR1 という名前のメッセージ・ファイルになります。

この付録の残りの部分には、S/36 の互換性に関心がある場合に知っておいた方がよいと思われる、画面データ管理機能のトピックごとのセクションが収められています。該当する場合、各トピックには以下の情報が含まれています。

- System/36 がその機能をサポートする方法。System/36 の機能についての詳細は、次の資料に記載されています。
 - *System/36 Creating Displays: Screen Design Aid and System Support Program*
 - *System/36 Concepts and Programmer's Guide*

- デフォルトの画面データ管理機能がその機能をサポートする方法
- System/36 に互換性のある機能を i5/OS 画面データ管理機能で取得する方法。互換性のある機能を得るために、上に列挙されている 3 つのレベルの互換性項目の中のどれが必要であるかも含まれます。

画面上の行の消去

System/36 アプリケーションがジョブの最初のレコードを書き込むと、System/36 ワークステーション・データ管理機能は、アプリケーション・データを表示する前に画面を消去します。プログラムが System/36 互換言語で書かれているか、またはディスプレイ・ファイルに USRDSPMGT キーワードが含まれている場合、System/36 環境ジョブでの最初の書き込み操作において、System/36 の機能が i5/OS システムに提供されます。「AS/400 システム/36 環境 プログラミング」の資料には、System/36 環境ジョブの開始および終了について記載されています。

i5/OS 画面データ管理機能は通常、ディスプレイ・ファイルがオープンされる時に画面を消去します。System/36 ワークステーション・データ管理機能は画面を消去しません。System/36 互換言語で書かれたプログラム、または USRDSPMGT キーワードを指定するディスプレイ・ファイルをオープンするプログラムの場合には、オペレーティング・システムにより System/36 の機能が提供されます。

アプリケーションは、行の消去 (CLRL) キーワードを使用することにより、画面のすべてまたは一部を消去することができます。CLRL キーワードは、System/36 環境が SFGR ソースを DDS に変換する時に、自動的に DDS ソースに書き込まれます。消去する行数について生成される値は、SFGR S 仕様の 19 および 20 桁目にある値に基づいています。

いくつかの i5/OS 機能と System/36 互換の画面消去機能は異なっているため、USRDSPMGT キーワードを指定するディスプレイ・ファイルでは以下の DDS キーワードを使用することはできません。

- ERASE
- PUTRETAIN
- KEEP
- ASSUME

さらに、System/36 互換言語で書かれたプログラム、および USRDSPMGT キーワードを指定するディスプレイ・ファイルでは、OVERLAY キーワードは無視されます。

ディスプレイ・ファイル・レコード用の入力データ

System/36 では、入力フィールドを持つ 1 つのレコードだけを表示できます。たとえば、System/36 上のアプリケーションがレコード REC1 を書き込み、さらにその後にレコード REC2 を書き込む場合、書き込まれた最終レコード (REC2) のみが入力可能フィールドを持ちます。REC1 が書き込まれると、REC1 により定義された入力フィールドが入力可能になります。REC2 が書き込まれた後は REC1 フィールドは入力可能ではなくなり、REC2 により定義された入力フィールドが入力可能になります。

デフォルトの i5/OS 画面データ管理機能サポートにより、入力フィールドを持つ多くの異なるレコードを同時に表示することができます。たとえば、レコード REC1 (行 1 から 12 を占める) およびレコード REC2 (行 21 から 24 を占める) を同時に表示することができます。アプリケーション・プログラムは、いずれのレコードからもデータを読み取ることができます。特定のレコードにより定義された入力可能フィールドのデータのみが戻されます。前の例において、アプリケーション・プログラムが、画面上のすべての入力可能フィールドからすべてのデータを取得する必要がある場合、アプリケーションは 2 回の読み取り操作を実行する必要があります。そのうちの 1 回は REC1 に対して、もう 1 回は REC2 に対して実行します。

System/36 互換言語で書かれたプログラム、または USRDSPMGT キーワードを指定するディスプレイ・ファイルでは、i5/OS 画面データ管理機能は、入力フィールドを持つ 1 つのレコードのみが許可されます。

ワークステーション制御装置からの入力データ

System/36 ワークステーション制御装置は、すべての入力可能フィールドに対してデータを戻します。デフォルトの i5/OS 画面データ管理機能サポートにより、ワークステーション制御装置は、ディスプレイ装置のオペレーターが変更した入力可能フィールドだけについて、画面にあるデータを戻します。

i5/OS 画面データ管理機能は、System/36 の機能に、DSPATR DDS キーワードの「変更データ・タグ」(MDT) 値を提供します。DSPATR(MDT) キーワードは、System/36 環境が SFGR ソースを DDS に変換する時に、各入力可能フィールドに対して自動的に DDS ソースに書き込まれます。このキーワードはワークステーション制御装置の標識をオンに設定し、DSPATR(MDT) を持つフィールドがディスプレイ装置のオペレーターにより変更されたかのようにワークステーション制御装置を作動させます。このキーワードはすべての入力可能フィールドに対して指定されるため、プログラムが読み取り操作を発行する時にすべての入力データはシステムおよびプログラムに戻されます。

自己検査

System/36 において、あるフィールドがモジュラス 10 またはモジュラス 11 の自己検査 (SFGR D 仕様の 30 桁目) を指定する場合、ワークステーション制御装置は、有効なデータがフィールドに入力されたことを確認します。ディスプレイ装置のオペレーターがフィールドに誤ったモジュラス 10 またはモジュラス 11 の番号を入力した場合、ワークステーション制御装置は明滅する 4 桁のエラー・コードを表示し、ユーザーはデータを再度フィールドに入力する必要があります。

自己検査用の i5/OS 画面データ管理機能サポート (CHECK(M10) および CHECK(M11) DDS キーワードを使用する) は、Enter キーまたは別の機能キーを押した時に、フィールドに有効なモジュラス 10 またはモジュラス 11 の番号が含まれているかを検査します。この検査は、データをフィールドに入力する際にエラー・メッセージを出す代わりに使用されます。CHECK(M10F) または CHECK(M11F) キーワードが指定された場合、ワークステーション制御装置は、有効なデータがフィールドに入力されていることを、前の段落で説明されているようにして検査します。

ディスプレイ・ファイルに CHECK(M10) または CHECK(M11) DDS キーワード、さらに USRDSPMGT キーワードが含まれている場合、画面データ管理機能は、System/36 同様、データを入力する際にエラー・メッセージを出します。DDS キーワード CHECK(M10F) および CHECK(M11F) は、USRDSPMGT キーワードとの使用は許可されません。

リターン入力

System/36 ワークステーションのデータ管理機能には、画面上のいずれかのフィールドがディスプレイ装置のオペレーターにより変更されたかどうかをアプリケーション・プログラムに示すためのサポートがあります。このサポートを使用するには、SFGR ソースで、SFGR S 仕様内のリターン入力 (22 桁目) に N を指定する必要があります。N が指定されていて、ユーザーが画面上のどのフィールドも変更していない場合、すべてのブランクは、読み取り操作に対して入力データとして受け取られます。

i5/OS 画面データ管理機能は、System/36 のサポートを提供します。CHANGE キーワードが、USRDSPMGT キーワードを指定するディスプレイ・ファイル内の応答標識を使用せずにレコード・レベルで指定されており、読み取り操作を行うプログラムが System/36 互換言語で書かれている場合、ディスプレイ装置のオペレーターが画面上のデータを全く変更していない場合には、プログラムはブランクのレコー

ドを受け取ります。リターン入力に N が指定されている場合、SFGR ソースが System/36 環境により DDS に変換される時、CHANGE キーワードが DDS ソースに自動的に書き込まれます。

入力フィールドの消去

System/36 では、プログラムがレコードを画面に書き込み、入力フィールド消去機能 (SFGR S 仕様の 31 および 32 桁目) がレコードに対して使用可能になっている場合、画面上の入力フィールドは消去されますが、指定されたレコードは画面に送信されません。たとえば、プログラムがレコード REC1 を画面に書き込んだ後、レコード REC2 を画面に書き込む場合 (この状態では REC2 の入力フィールド消去機能が使用可能になっている)、REC1 により定義されたすべての入力フィールドは消去され、REC2 は画面に送信されません。

i5/OS 画面データ管理機能では、プログラムが System/36 互換言語で書かれ、USRDSPMGT キーワードが指定されており、さらにレコードに対して ERASEINP キーワードが使用可能になっている場合、System/36 の入力フィールド消去の処理方法に従います。

System/36 では、プログラムがレコードを画面に書き込み、そのレコードに対して入力フィールド消去機能および一時変更 PUT 機能の両方が使用可能になっている場合、画面上の入力フィールドは消去され、指定されたレコード用の一時変更 PUT データが画面に送信されます。

i5/OS 画面データ管理機能では、プログラムが System/36 互換言語で書かれ、USRDSPMGT キーワードが指定されており、さらにレコードに対して ERASEINP および PUTOVR キーワードが使用可能になっている場合、System/36 の入力フィールド消去の処理方法に従います。

表示属性

System/36 では、フィールドの一部には開始表示属性がありません。フィールドは、以下の 1 つ以上が当てはまる場合のみ開始表示属性を持ちます。

- フィールドが入力可能である。
- 出力データが標識 (SFGR D 仕様の 23 から 24 桁目) を基にしている。
- フィールドが表示属性 (高輝度、明滅、反転表示、下線、列分離子など) を指定している。
- フィールドが、レコードにより消去されない行に表示される。

System/36 とのアプリケーション互換性を提供するために、i5/OS 画面データ管理機能は、USRDSPMGT キーワードがディスプレイ・ファイルに指定されている場合にフィールドに開始属性を含めるかどうかを、System/36 の規則に従って判断します。さらに、フィールドが開始属性を持たず、OVRATR キーワードがフィールドに指定されている場合、OVRATR キーワードは無視されます (フィールド属性は送信されません)。

System/36 では、フィールドの一部には終了表示属性がありません。フィールドは、以下の 1 つ以上が当てはまる場合のみ終了属性を持ちます。

- フィールドが入力可能である。
- 出力データが標識 (SFGR D 仕様の 23 から 24 桁目) を基にしている。
- フィールドが表示属性 (高輝度、明滅、反転表示、下線、列分離子など) を指定している。

i5/OS 画面データ管理機能は、USRDSPMGT キーワードがディスプレイ・ファイルに指定されている場合にフィールドに終了属性を含めるかどうかを、System/36 の規則に従って判断します。

カーソルの位置決め

System/36 は、レコードが表示される時にどこにカーソルを配置するかを決定するため、以下の規則を使用します。

1. カーソルは、カーソル位置決め属性に指定された標識を持ち、この標識がオンに設定され、さらに以下のいずれかの条件を満たす (SFGR ソース・ステートメントにより定義された) 最初の入力フィールドに配置されます。
 - このフィールドは保護属性が指定されていない (SFGR D 仕様の 37 および 38 桁目)。
 - このフィールドは保護属性の標識が指定されているが、標識がオフである。
 - このフィールドはオプションなし保護属性が指定されている (SFGR D 仕様の 37 または 38 桁目が Y または N)。
2. カーソルが規則 1 により配置されない場合、オプションなしカーソル位置決め属性を持つ (SFGR D 仕様の 32 または 33 桁目が Y) フィールドに対して検査が行われます。オプションなしカーソル位置決め属性を持つフィールドがある場合、カーソルはこのフィールドに配置されます。
3. カーソルが規則 1 または 2 により配置されない場合、カーソルは、オプションなし保護属性を持たない (SFGR D 仕様の 37 または 38 桁目がブランク) 最初の入力フィールド (SFGR ソース・ステートメントで定義済み) に配置されます。
4. カーソルが上記のいずれの規則によっても配置されない場合、カーソルは画面の左上隅に配置されません。

i5/OS 画面データ管理機能は、USRDSPMGT キーワードが指定されているディスプレイ・ファイルでは、System/36 の規則に従ってカーソルを配置します。

メッセージの表示

System/36 では、レコードが表示される時にメッセージ・テキストが自動的にフィールドに挿入されるようなフィールドを定義できます。System/36 ワークステーションのデータ管理機能は、フィールド用に定義されたメッセージ・テキストを検索し、そのメッセージ・テキストを出力データとしてフィールドに提供します。メッセージ・テキストを表示するには、System/36 アプリケーションで以下を指定する必要があります。

- 定数タイプとして M (SFGR D 仕様の 56 桁目)。
- メッセージ識別コード (MIC) (SFGR D 仕様の 57 から 60 桁目、またはプログラムにより提供された出力データ)。
- メッセージ・メンバー ID (SFGR D 仕様の 61 および 62 桁目、またはプログラムにより提供された出力データ)。

i5/OS アプリケーションはさまざまな異なる方法でメッセージを表示することができます。メッセージの表示については、247 ページの『独自のメッセージの作成および表示』を参照してください。

System/36 とのアプリケーション互換性を提供するために、MSGID キーワードをメッセージの表示に使用することができます。SFGR D 仕様の定数タイプに M が指定されている場合、MSGID キーワードは System/36 環境により自動的に生成されます。

メッセージの送信のための System/36 の規則に従うメッセージを表示するために、以下のいずれかの形式が使用されます。

```
MSGID(message-identifier message-file)
MSGID(USR message-identification-code message-file)
```

Message-identifier は、メッセージ接頭語およびメッセージ識別コード (MIC) の 2 つの部分で構成されます。アプリケーションが **MSGID** キーワードの最初の形式を使用する場合、3 文字のメッセージ接頭語は、7 文字の *message-identifier* の最初の 3 文字でなければなりません。アプリケーションが **MSGID** キーワードの 2 番目の形式を使用する場合、接頭部を指定する必要はありません。接頭部は、**MSGID** キーワードですでに **USR** として指定されています。

message-identification-code パラメーターは、表示されるメッセージの 4 文字のメッセージ ID を指定します。

ディスプレイ・ファイルの *message-identifier* の指定についての詳細は、iSeries Information Center の **DDS** のトピックを参照してください。

message-file パラメーターは、表示されるメッセージが入っているメッセージ・ファイルを示します。System/36 との互換性のために、このパラメーターは次のいずれかの形式で指定できます。

message-file パラメーターを指定するための最初の形式は、2 文字のフィールド名です。フィールド名は **MSGID** フィールドと同じレコードに存在していなければならず、フィールドは、使用法 **H** (隠蔽)、**P** (プログラムからシステムへ)、**B** (両方)、または **O** (出力のみ) の文字フィールドとして定義されている必要があります。フィールド名により識別されるフィールドは、メッセージを表示するために System/36 環境が使用するメッセージ・ファイルを示します。フィールドに指定できる値は次のとおりです。

表 104. *MSGID* 用のメッセージ・ファイル

フィールド名の値	使用されるメッセージ・ファイル
U1	MEMBER OCL ステートメントの USER1 パラメーターに指定されたメッセージ・ファイルからのメッセージ・テキスト。
U2	MEMBER OCL ステートメントの USER2 パラメーターに指定されたメッセージ・ファイルからのメッセージ・ヘルプ・テキスト。
P1	MEMBER OCL ステートメントの PROGRAM1 パラメーターに指定されたメッセージ・ファイルからのメッセージ・テキスト。
P2	MEMBER OCL ステートメントの PROGRAM2 パラメーターに指定されたメッセージ・ファイルからのメッセージ・ヘルプ・テキスト。
M1	IBM 提供のメッセージ・ファイル ##MSG1 からのメッセージ・テキスト。
M2	IBM 提供のメッセージ・ファイル ##MSG1 からのメッセージ・ヘルプ・テキスト。

message-file パラメーターを指定するための 2 番目の形式は、特殊値です。以下の特殊値をメッセージ・ファイルに指定できます。

表 105. *MSGID* 用のメッセージ・ファイル

特殊値	使用されるメッセージ・ファイル
*USR1	MEMBER OCL ステートメントの USER1 パラメーターに指定されたメッセージ・ファイルからのメッセージ・テキスト。
*USR2	MEMBER OCL ステートメントの USER2 パラメーターに指定されたメッセージ・ファイルからのメッセージ・ヘルプ・テキスト。
*PGM1	MEMBER OCL ステートメントの PROGRAM1 パラメーターに指定されたメッセージ・ファイルからのメッセージ・テキスト。
*PGM2	MEMBER OCL ステートメントの PROGRAM2 パラメーターに指定されたメッセージ・ファイルからのメッセージ・ヘルプ・テキスト。

表 105. MSGID 用のメッセージ・ファイル (続き)

特殊値	使用されるメッセージ・ファイル
*SYS1	IBM 提供のメッセージ・ファイル ##MSG1 からのメッセージ・テキスト。
*SYS2	IBM 提供のメッセージ・ファイル ##MSG1 からのメッセージ・ヘルプ・テキスト。

MSGID キーワードの System/36 互換機能を使用する場合、注意しなければならない以下の考慮事項があります。

- フィールド名または特殊値が指定されている場合、i5/OS 画面データ管理機能により行われる処理は、アプリケーションが実行される環境により異なります。たとえば、アプリケーションが System/36 環境で実行される場合、*USR1 は、MEMBER OCL ステートメントの USER1 パラメーターにより識別されるメッセージ・ファイルを使用するということを示します。同じアプリケーションが System/36 環境で稼働していない場合には、*USR1 は、メッセージ・ファイル USR1 を使用するということを示します。
- メッセージが表示されることになり、メッセージにメッセージ・ヘルプのみが含まれる場合、メッセージ・ヘルプが表示されます。
- メッセージ・ヘルプが表示されることになっているものの、メッセージにメッセージ・ヘルプがない場合、メッセージのみが表示されます。
- 表示予定のメッセージ・テキストが出力フィールドの長さより長い場合、メッセージ・テキストは出力フィールドの長さに切り捨てられます。表示予定のメッセージ・テキストが出力フィールドの長さより短い場合、メッセージ・テキストにはブランクが埋め込まれます。
- メッセージ ID に無効な文字が含まれている場合 (有効な文字は 0 から 9 および A から F)、表示されるメッセージ・テキストは、後ろに 2 文字のメッセージ・ファイル ID が付いたメッセージ ID になります。
- メッセージ ID またはメッセージ・ファイルが見つからない場合、表示されるメッセージ・テキストは、後ろに 2 つの疑問符 (??) が付いたメッセージ識別コードになります。
- 以下の DDS キーワードは、MSGID キーワードが含まれるフィールドでは指定できません。

DFT

DFTVAL

FLTFIXDEC

FLTPCN

MSGCON

一時変更の PUT

System/36 では、一時変更 PUT オプションが使用可能になっていて (SFGR S 仕様の 33 および 34 桁目)、消去する行数 (SFGR S 仕様の 19 および 20 桁目) が指定されている場合、消去する行数は無視されます。System/36 ワークステーションのデータ管理機能は、一時変更 PUT オプションが使用可能になっている場合には画面を消去しません。i5/OS 画面データ管理機能も、一時変更 PUT オプション (PUTOVR キーワード) が使用可能になっているレコードの消去機能 (CLRL キーワード) の無視をサポートします。このサポートは、System/36 互換言語で書かれたプログラム、または USRDSPMGT キーワードを指定するディスプレイ・ファイルを使用するプログラムの場合に使用されます。

System/36 では、一時変更 PUT オプションが使用可能である場合、入力フィールド定義は画面に送信されません。入力フィールド定義には、入力フィールドの長さ、全桁入力必須属性、入力必須属性、変更データ・タグ属性、保護入力フィールド属性などの情報が含まれます。入力フィールドが現在画面上にある場

合、これらのフィールドは入力可能のままになります。デフォルトの i5/OS 画面データ管理機能サポートは、一時変更 PUT オプションが使用可能の場合に入力フィールド定義を送信します。i5/OS 画面データ管理機能は、一時変更 PUT 機能が使用可能であり、表示されるレコードが USRDSPMGT キーワードを指定するディスプレイ・ファイルにある場合に、入力フィールド定義を画面に送信しないという System/36 の機能を備えています。アプリケーションが、一時変更 PUT 機能が使用可能である場合に入力フィールド定義を画面に送信しないという System/36 の機能を使用する場合、入力フィールド定義により定義された属性は、一時変更 PUT が使用可能である場合には変更できません。たとえば、一時変更 PUT が使用可能である場合には、入力フィールド定義が画面に送信されないために、保護入力フィールド属性 (DSPATR(PR)) は無視されます。

System/36 では、現在表示されているレコードおよび現在画面にはないレコードに対して一時変更 PUT オプションを指定できます。たとえば、レコード REC1 が現在表示されており、アプリケーション・プログラムが、一時変更 PUT 機能をレコード REC2 に対して使用可能にして書き込み操作を実行する場合、レコード REC2 を表示する要求は一時変更 PUT 要求として処理されます。i5/OS 画面データ管理機能のデフォルトのサポートは、現在表示されているレコードに対してのみ、一時変更 PUT 機能をサポートします。たとえば、レコード REC1 が現在表示されており、アプリケーション・プログラムが、一時変更 PUT 機能をレコード REC2 に対して使用可能にして書き込み操作を実行する場合、一時変更 PUT 機能は無視され、レコード REC2 を表示する要求は PUTOVR キーワードが使用可能ではなかった場合のように処理されます。ディスプレイ・ファイルが USRDSPMGT キーワードを指定する場合、i5/OS 画面データ管理機能は、レコードが現在表示されていない場合でも一時変更 PUT 機能を許可する System/36 の機能を備えています。

一時変更操作中、i5/OS 画面データ管理機能は、アプリケーション・プログラムにより表示されるレコード様式に含まれるヘルプ仕様を無視する場合があります。これは、USRDSPMGT キーワードが使用されている場合に発生します。ヘルプ消去 (HLPCLR) キーワードもレコード様式に指定されている場合、オンライン・ヘルプは使用できません。

符号付き数字データの処理

System/36 は、書き込み操作において符号付き数字フィールドに対して誤ったデータ (0 から 9 以外のデータ) が指定されている場合には検査をしません。たとえば、符号付き数字フィールド内のデータが文字データである場合、文字データが表示されます。i5/OS 画面データ管理機能のデフォルトのサポートでは、フィールドが表示される時に、すべての誤った符号付き数字データをヌルに置換します。i5/OS 画面データ管理機能は、USRDSPMGT キーワードがディスプレイ・ファイルで指定されている場合には System/36 の機能をサポートします。

読み取り操作で符号付き数字フィールドに対して誤ったデータが戻される場合にも、System/36 は検査を行いません。たとえば、符号付き数字フィールドのために戻されたデータが文字データである場合、文字データがアプリケーション・プログラムに渡されます。i5/OS 画面データ管理機能のデフォルトのサポートでは、データがアプリケーション・プログラムに渡される前に、すべての誤った符号付き数字データをゼロに置換します。i5/OS 画面データ管理機能は、System/36 互換言語で書かれたプログラムの System/36 の機能をサポートします。

System/36 では、符号付き数字フィールドにゼロが入力されており、フィールドから出るためにフィールド・マイナス・キーが使用される場合、負のゼロの値がアプリケーション・プログラムに戻されます。i5/OS 画面データ管理機能のデフォルトのサポートでは、負のゼロの代わりにゼロをアプリケーション・プログラムに戻します。i5/OS 画面データ管理機能は、System/36 互換言語で書かれたプログラム、または USRDSPMGT キーワードを指定するディスプレイ・ファイルを使用するプログラムのための System/36 の機能をサポートします。

機能キー

System/36 では、機能キーが押されると、アプリケーション・プログラムは制御を受け取ることができません。このサポートを使用するには、アプリケーションは必要な機能キーを SFGR ソース内で使用可能に設定する必要があります。プログラムは、機能キーをそのプログラムで処理できることを示す必要があります。

デフォルトの i5/OS 画面データ管理機能サポートでは、System/36 互換言語で書かれていないプログラムが、どの機能キーをプログラムで処理できるかを示すことを許可しません。DDS の中でディスプレイ・ファイルに対して使用可能に設定されている機能キーは、i5/OS 画面データ管理機能がプログラムに渡して処理する機能キーです。

i5/OS 画面データ管理機能は、System/36 互換言語で書かれたプログラムのために、どの機能キーをプログラムで処理できるかをそのプログラムが示す System/36 の機能をサポートします。プログラムでの機能キー処理の使用可能化については、該当する System/36 互換言語の資料を参照してください。

ヘルプ・キーの考慮事項

HELP および HLPRTN DDS キーワードは、ヘルプ・キーが押された時にシステムがどのような処理を実行するかを示すために使用されます。HELP キーワードは、アプリケーションがヘルプ・キーを使用可能にしたいということを示します。SFGR ソースが System/36 環境により DDS に変換される時、HELP キーワードが DDS ソースに自動的に書き込まれます。

HLPRTN キーワードは、ディスプレイ装置のオペレーターがヘルプ・キーを押した時にアプリケーション・プログラムでヘルプ・キーを処理したいということを示します。ヘルプ・キーが SFGR キー・マスク (SFGR S 仕様の 64 から 79 桁目) で使用可能に設定されている場合、HLPRTN キーワードは、System/36 環境が SFGR ソースを DDS に変換する時に DDS ソースに書き込まれます。

HELP および HLPRTN キーワードが指定されておらず、ディスプレイ装置のオペレーターがヘルプ・キーを押した場合、ヘルプ・キーは許可されていないということを示すメッセージが i5/OS 画面データ管理機能により表示されます。

USRDSMGT および HELP キーワードが指定されていて、HLPRTN キーワードが指定されておらず、さらにディスプレイ装置のオペレーターがヘルプ・キーを押した場合、i5/OS 画面データ管理機能は以下の規則を使用してヘルプ・キーの処理方法を決定します。

1. メッセージが表示されているときにヘルプ・キーが押されると、メッセージに対するメッセージ・ヘルプが表示されます。
2. 規則 1 によってヘルプ・キーが処理されず、レコードに対して定義されたアプリケーション・ヘルプがある場合、レコードのアプリケーション・ヘルプが表示されます。
3. 規則 1 または 2 によってヘルプ・キーが処理されなかった場合、そのキーが許可されていないということを示すエラー・メッセージが表示されます。

USRDSMGT、HELP、および HLPRTN キーワードが指定されていて、ディスプレイ装置のオペレーターがヘルプ・キーを押した場合、i5/OS 画面データ管理機能は以下の規則を使用してヘルプ・キーの処理方法を決定します。

1. アプリケーション・プログラムがヘルプ・キーを処理できるということをそのプログラムが示した場合、ヘルプ・キーが押されたという標識がアプリケーション・プログラムに戻されます。
2. 規則 1 によってヘルプ・キーが処理されず、メッセージが表示されているときにヘルプ・キーが押されると、メッセージに対するメッセージ・ヘルプが表示されます。

- 規則 1 または 2 によってヘルプ・キーが処理されず、レコードに対して定義されたアプリケーション・ヘルプがある場合、レコードのアプリケーション・ヘルプが表示されます。
- 規則 1、2、または 3 によってヘルプ・キーが処理されなかった場合、そのキーが許可されていないことを示すエラー・メッセージが表示されます。

コマンド・キーを使用したアプリケーション・ヘルプの終了

System/36 では、アプリケーション・ヘルプが表示されていて、ディスプレイ装置のオペレーターがアプリケーション・ヘルプを終了させるためのコマンド・キーを押した場合、どのコマンド・キーが押されたかを示す標識がアプリケーション・プログラムに渡されます。このサポートを使用するには、アプリケーションは、ヘルプ・キーが押される画面上で希望するコマンド・キーを使用できるようにする必要があります。そのキーはアプリケーション・ヘルプ画面で使用可能に設定されている必要があります。アプリケーション・ヘルプが入っているディスプレイ・ファイルが `USRDSPMGT` キーワードを指定する場合、i5/OS 画面データ管理機能は、アプリケーション・ヘルプを終了させるためにどのコマンド・キーが押されたかを示す標識を戻します。コマンド・キーがアプリケーション・レコードとヘルプ・レコードの両方には指定されなかった場合、「機能キーは使用できません」というメッセージが表示されます。

System/36 では、アプリケーション・ヘルプが表示され、ディスプレイ装置のオペレーターがアプリケーション・ヘルプを終了させるためのコマンド・キーを押した場合、ヘルプ・キーが押された画面からのデータがアプリケーション・プログラムに戻されます。このサポートを使用するには、アプリケーションは、ヘルプ・キーが押される画面に `restore yes` (SFGR H 仕様の 47 および 48 桁目) を示す必要があります。アプリケーション・ヘルプが入っているディスプレイ・ファイルが `USRDSPMGT` キーワードを指定する場合、i5/OS 画面データ管理機能は、ヘルプ・キーが押された画面からデータを戻します。

送信勧誘取り消し操作

System/36 では、プログラムが送信勧誘取り消し操作を送出するものの、システムがすでにデータを受け取っている場合 (たとえば、送信勧誘取り消しが出される前にディスプレイ装置のオペレーターが `Enter` キーを押した場合)、データは失われ、アプリケーションには通知されません。

デフォルトの i5/OS 画面データ管理機能サポートでは、データが使用できる場合には送信勧誘取り消し操作を失敗させます。メッセージ `CPF4737` および戻りコード `0412` がアプリケーション・プログラムに送信されます。アプリケーションは戻りコードを検査して、ディスプレイ装置から戻されたデータを受け取るための読み取り操作を送出することができます。

System/36 互換言語で書かれたプログラム、または `USRDSPMGT` キーワードが指定されたディスプレイ・ファイルでは、i5/OS 画面データ管理機能は、System/36 の機能をサポートし、メッセージ `CPF4737` を送信したり戻りコードを設定することはありません。

ディスプレイ・ファイルに `USRDSPMGT` キーワードおよび `INVITE` キーワードが含まれている i5/OS 画面データ管理機能環境で稼働するアプリケーションの場合、以下のようになります。

- レコードが別のディスプレイ・ファイルから書き込まれる場合にファイルが読み取り未処理で活動状態である場合、新しいレコードの書き込みの前に現行表示画面の保管が行われます。
- 中断状態のファイルは、中断状態のファイルに書き込みが行われる時に復元されます。

System/36 では、この状態では保管は行われません。

i5/OS 画面データ管理機能環境で保管コマンドを迂回するには、以下のようになります。

1. 図 147に示されているように、ファイル FILE1 に RECORDD という名前のレコードを定義します。
2. レコード RECORDA を書き込む前にプログラム PGM1 にレコード RECORDD を書き込みます。

レコード RECORDD を書き込むことによりレコード RECORDB の読み取り未処理が取り消され、未処理の送信勧誘がなくなるため、レコード RECORDB の保管/復元は必要なくなります。

```

PGM1 (RPG PROGRAM)
      MOVE *ON      *IN24
      WRITERECORDB
      WRITERECORDD
      WRITERECORDA
      EXFMTRECORDC
      ENDSR

*****

FILE1
      USRDSPMGT
R RECORDB      OVERLAY
                CF(01)
                INVITE
      1 18'TESTB'
R RECORDC      OVERLAY
      1 18'TESTC'
R RECORDD      OVERLAY

*****

FILE2
      USRDSPMGT
R RECORDA      OVERLAY
      22 5'TEST_____A'
24             DSPATR(ND)

```

図 147. 保管コマンドの迂回

コマンドおよび機能キーの保持

System/36 では、プログラムがレコードを表示するために書き込み操作を実行する場合、アプリケーションは、ディスプレイ装置において現在活動状態のコマンドおよび機能キーが新しいレコードで使用されるコマンドおよび機能キーとなるという指示を出すことができます。たとえば、アプリケーションがレコード REC1 を書き込み (REC1 はコマンド・キー 1 から 12 とすべての機能キーを使用可能にする)、その後アプリケーションがレコード REC2 を書き込む (REC2 は REC1 のコマンド・キーと機能キーを保持する) 場合には、コマンド・キー 1 から 12 とすべての機能キーは、REC2 が表示されるときに使用可能になります。System/36 アプリケーションは、SFGR ソースのコマンド・キー使用可能オプション (SFGR S 仕様の 28 桁目) に R を指定することにより、コマンド・キーを保持するよう指定します。System/36 アプリケーションは、SFGR ソースの機能キー使用可能オプション (SFGR S 仕様の 27 桁目) に R を指定することにより、機能キーを保持するよう指定します。

i5/OS アプリケーション・プログラムは、レコードが表示されているときに有効なコマンド・キーおよび機能キーのリスト、または現在活動状態であるコマンド・キーおよび機能キーはレコードが表示されているときに活動状態を保つという指示のいずれかを指定することができます。

i5/OS 画面データ管理機能は、レコードに RETCMDKEY キーワードが指定されている場合に、アプリケーションがレコードのための現行のコマンド・キーを保持できるようにします。SFGR S 仕様のコマンド・キーに R が指定されている場合、RETCMDKEY キーワードは System/36 環境により自動的に生成されます。

さらに、i5/OS 画面データ管理機能は、レコードに RETKEY キーワードが指定されている場合に、アプリケーションがレコードのための現行の機能キーを保持できるようにします。SFGR S 仕様の機能キーに R が指定されている場合、RETKEY キーワードは System/36 環境により自動的に生成されます。

サポートされない System/36 の機能

このセクションでは、i5/OS 画面データ管理機能によりサポートされない System/36 ワークステーションのデータ管理機能について説明します。

あるプログラムが、画面に送信勧誘する書き込み操作を実行し、別のプログラムが、同じ画面に対して一時変更 PUT 機能を使用可能に設定して書き込み操作を実行する場合、すべての書き込み/読み取り操作で同じディスプレイ・ファイルのレコード様式を使用する必要があります。たとえば、プログラム A が、ファイル FILE1 にあるレコード REC1 を書き込むことによりプログラム間画面データ渡し (RUF) 要求をセットアップする場合、2 番目のプログラムも、一時変更 PUT 機能が使用可能に設定されているすべての書き込み操作に対して FILE1 を使用する必要があります。異なるディスプレイ・ファイルが使用される場合、一時変更 PUT 機能が i5/OS 画面データ管理機能により処理されるときには、REC1 にあるすべての入力フィールドは出力専用フィールドに変更されます。

アプリケーション・プログラムが RUF を使用している場合、一時変更 PUT 機能は異なるジョブ間でのみ使用できます (たとえば、複数要求端末 (MRT) プログラムに制御を与える単一要求端末 (SRT) プログラム)。RUF が進行しておらず、2 番目のジョブが一時変更 PUT 機能を使用可能にして書き込み操作を実行する場合、一時変更 PUT 機能が i5/OS 画面データ管理機能により処理されるときには、画面上にあるすべての入力フィールドは出力専用フィールドに変更されます。

System/36 では、アプリケーション・ヘルプはアプリケーション・ヘルプ画面上の入力フィールドからデータを戻すことができます。これは、i5/OS 画面データ管理機能ではサポートされません。

制限された DDS キーワード/機能

USRDSPMGT キーワードが DDS ソースで指定されている場合、以下の DDS 機能は使用できません。

- 画面消去を制御する以下の DDS キーワード:
 - ASSUME
 - ERASE
 - KEEP
 - PUTRETAIN
- ERRSFL DDS キーワード
- HLPCMDKEY DDS キーワード
- 以下のサブファイル DDS キーワード:
 - SFL
 - SFLCTL

System/36 互換言語はサブファイルをサポートしないため、USRDSPMGT が指定されている場合にはサブファイルはディスプレイ・ファイルで使用できません。

- 応答標識

System/36 互換言語は、読み取り操作によりディスプレイ・ファイルに標識を戻さないため、USRDSPMGT が使用される場合には応答標識を指定すべきではありません。

- IGC 変換

入力可能な形式は一度に 1 つしか画面で保守できないため、IGC 変換は System/36、または System/36 環境での i5/OS ではサポートされません。USRDSPMGT キーワードを使用する場合、IGCCNV キーワードはファイル定義のどの場所でも許可されません。

- 以下の DDS キーワード:

- CHECK(M10F)
- CHECK(M11F)

USRDSPMGT と共に DDS キーワードの CHECK(M10) および CHECK(M11) を指定すると、USRDSPMGT なしの CHECK(M10F) および CHECK(M11F) と同じ機能を持つためです。

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

- | 〒106-0032
- | 東京都港区六本木 3-2-31
- | IBM World Trade Asia Corporation
- | Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。 IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとしします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Corporation
Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

- | 本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム
- | 契約の契約条項、IBM プログラムのご使用条件、IBM 機械コードのご使用条件、またはそれと同等の条項
- | に基づいて、IBM より提供されます。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. _年を入れる_. All rights reserved.

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は表示されない場合があります。

プログラミング・インターフェース情報

本書「Application Display Programming」には、プログラムを作成するユーザーが IBM i5/OS のサービスを使用するためのプログラミング・インターフェースが記述されています。

商標

以下は、IBM Corporation の商標です。

- |
- | AFP
- | AIX
- | Common User Access
- | CUA
- | IBM
- | InfoWindow
- | iSeries
- | OS/2

- | PS/2
- | System/36
- | System/38
- | Systems Application Architecture

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

使用条件

これらの資料は、以下の条件に同意していただける場合に限りご使用いただけます。

個人使用: これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布 (頒布、送信を含む) または表示 (上映を含む) することはできません。

商業的使用: これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。

お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入関連法規を含む、すべての関連法規を遵守するものとします。

IBM は、これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態を提供され、第三者の権利の不侵害の保証、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任なしで提供されます。

参考文献

以下のリストは、本書に関連した資料および iSeries Information Center のトピックです。資料は、完全なタイトル、基本資料番号、および内容の説明とともにリストされています。各資料の説明には、その資料と本書との関連が含まれます。

システム使用

- iSeries Information Center の基本システム操作のトピック: この情報を使用して、サーバーと i5/OS オペレーティング・システムに関連した導入概念のいくつかを学習できます。また、このガイドを使用して、基本システム操作のタスクを実行することができます。たとえば、サーバーの開始および停止、ユーザー、ジョブ、およびデバイスに関する処理などを実行できます。

システム管理

- iSeries Information Center のシステム管理のトピック: このトピックでは、実行管理機能環境の作成と変更、システム値の処理、およびパフォーマンス・データを収集し、それを使用してシステム・パフォーマンスを改善する方法について説明します。

アプリケーション開発

- AS/400 適用業務開発ツール・セット 文字作成ユーティリティ (CGU), SC88-5196-00

このガイドでは、アプリケーション開発ツールの文字作成ユーティリティ (CGU) を使用して、i5/OS システム上で 2 バイト文字 (DBCS) を作成および保守する方法について説明します。

ユーザー定義文字を DBCS フォント・テーブルに追加する際に、このガイドを使用してください。

- AS/400 プログラム開発管理機能 (PDM), SC88-5197-00

このガイドおよび解説書は、アプリケーション開発ツールのプログラム開発管理機能 (PDM) を使用して、コピー、削除、および名前変更などの操作を簡単に行うために、ライブラリー、オブジェクト、メンバー、およびユーザー定義オプションのリストで作業する方法を説明します。

DDS および UIM ソースを入力する方法を学習するには、このガイドおよび解説書を使用してください。

- AS/400 適用業務開発ツールセット AS/400 用画面設計機能 (SDA), SD88-5046-00

このガイドおよび解説書は、アプリケーション開発ツールの画面設計機能 (SDA) を使用して、画面、メニュー、およびオンライン・ヘルプ情報を設計、作成、および保守する方法を説明します。

独自の画面を設計するとき、システムによって DDS を作成する場合にこのガイドおよび解説書を使用してください。

- AS/400 適用業務開発ツールセット AS/400 用原始ステートメント入力ユーティリティ 使用者の手引きと参照, SD88-5047-00

このガイドは、アプリケーション開発ツールの原始ステートメント入力ユーティリティ (SEU) を使用して、ソース・メンバーを作成および編集する方法についての情報を、アプリケーション・プログラマーに提供します。この資料には、SEU セッションの開始と終了、この全画面テキスト・エディターの多数の機能を使用する方法が説明されています。

DDS、CL、およびその他のコードを作成および編集するために使用可能な SEU の機能を理解するには、このガイドを使用してください。

- Common User Access Basic Interface Design Guide, SC26-4583.

このガイドは、アプリケーション内で、およびその他のアプリケーションの間で一貫性のあるユーザー・インターフェースを設計するための

助けになります。このガイドは、ユーザー、設計者または開発者 (あるいはその両方) が考慮する必要がある、ユーザー・インターフェースのスタイル・ガイドラインおよびインプリメンテーションの考慮事項を示します。

通信および接続

- *ICF Programming*, SC41-5442-00

このガイドは、i5/OS 通信および i5/OS システム間通信機能 (i5/OS-ICF) を使用するアプリケーション・プログラムを作成するために必要な情報を提供します。また、データ記述仕様 (DDS) キーワード、システム提供フォーマット、戻りコード、ファイル転送サポート、およびプログラム例に関する情報も含まれています。

リモート・システムでアプリケーションを使用して作業する際に、このガイドを使用してください。

- *IBM 5250 Information Display System Functions Reference Manual*, SA21-9247.

この解説書は、5250 情報画面の機能を使用する方法を説明します。

5250 画面データ・ストリームを使用する際に、この資料を使用してください。

- *IBM 5494 遠隔制御装置機能参照マニュアル*, SC88-6207.

この解説書は、IBM 5494 リモート制御装置が、システム・ネットワーク体系 (SNA)、同期データ・リンク制御 (SDLC)、X.25、X.21、またはトークンリング・プロトコルを使用して、i5/OS と通信する方法を説明します。さらに、5494 が、接続されたワークステーションを管理し、ネットワーク・データ・ストリームを 5250 ワークステーションおよびプリンター用のプロトコルに変換する方法も説明します。

プログラム・イネーブラー

- iSeries Information Center の **DDS** のトピック: このトピックでは、データベース・ファイル (論理および物理の両方)、およびユーザーのプログラムにとって外部的な特定の装置ファイル

(ディスプレイ、印刷装置、ICF 用) を記述するのに必要な入力項目およびキーワードについて説明します。

このトピックを使用して、ディスプレイ・ファイルで使用される特定の DDS キーワードに関する詳細情報を入手できます。

- iSeries Information Center の **CL** のトピック: このトピックでは、i5/OS コマンドおよびその他の IBM 提供のコマンドを使用してプログラムを作成するシステム・プログラマーおよびシステム管理者に情報を提供します。

このトピックでは、i5/OS 制御言語 (CL) とそのコマンドに関する説明があります。(i5/OS 以外のコマンドの説明は、各ライセンス・プログラムに関する資料にあります。) この資料には、システム用のすべての CL コマンドの解説、およびこれらのコマンドをコーディングするために必要な構文規則の説明もあります。

このトピックは、i5/OS のプログラミングに関係する事柄を広範囲にわたって論じています。その中には、制御言語 (CL) プログラミング、プログラム相互間の流れと通信の制御、CL プログラムにおけるオブジェクトの扱い方、および CL プログラムの作成方法などがあります。他のトピックとしては、事前定義・即時メッセージとメッセージ処理、ユーザー定義コマンドおよびメニューの定義および作成、デバッグ・モード、停止点、追跡、および表示機能などのアプリケーションのテストを行う方法が説明されています。

このガイドを使用して、アプリケーション・プログラムを作成するために CL を使用する方法を理解できます。

- iSeries Information Center の **ファイルおよびファイル・システム** のトピック: このトピックでは、アプリケーション・プログラムでのファイルの使用に関する情報を提供します。これには、システム上のデータ管理機能サポートの基本構造および概念についての情報が含まれます。また、オーバーライドおよびファイル転送 (アプリケーション・プログラムの実行時にファイルに一時的な変更を加えること)、データをある場所から別の場所にコピーするシステム・コマンドを使用したファイルのコピー、および 2 バイト・データを使用するシステムの調整につ

いての情報も含まれます。さらに、印刷を制御し、理解する方法についての情報も提供します。これには、プリンターの接続性、PC に接続されたプリンター、および高機能印刷 (AFP™) プリンターが含まれます。最後に、磁気テープ用のデータ管理機能サポートについても情報を提供します。

このトピックは、アプリケーション・プログラムでのファイル (プリンター・ファイルおよび磁気テープ・ファイルを含む) の使用に関する一般情報として使用します。

- AS/400 システム/36 環境 プログラミング, SC88-5099-00.

このガイドは、i5/OS システム上の System/36 環境でアプリケーション・プロセスの相違点を識別するための情報を提供します。このガイドにより、i5/OS システム上の System/36 環境で処理を行う際の、(System/36 から見た) 機能および操作の相違点を理解できます。これには、環境機能の概要、マイグレーション、プログラミング、通信、セキュリティー、および共存に関する考慮事項が含まれます。

プログラム・インターフェース

- iSeries Information Center のプログラミングのカテゴリの API のトピック: このトピックでは、高水準言語プログラムから i5/OS アプリケーション・プログラミング・インターフェース (API) を使用して、データのリストや記述および情報の検索の実行を希望する、経験を積んだアプリケーション・プログラマーおよびシステム・プログラマーに、情報および例を提供します。

このトピックは、API の詳細情報を見つけるために使用します。

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

アウトライン、パネル

編成 514

UIM メニュー・アウトライン 514

アウトライン、UIM メニュー

編成 514

アクション変数定義 375, 376

アクセス権の除去

F18=続きの見出し 424

アクセラレーター・キー

定義 170

例 171

アクティブ・レコード様式

判別 49

表 49

値

時刻 538

数値 538

日付 538

文字 538

BASETYPE 526

アプリケーション

ヘルプ 203

メニュー・バーとの相互作用 197

アプリケーション画面

アクティブ・レコード様式の判別 49

書き込まれるレコード様式の順序 53

システムが入力を読み取る方法 82

出力の書き込み 49

情報

参照: 情報画面

スペースを占有しないレコードについての理解 52

送信勧誘された入力 74

単一選択メニュー

参照: メニュー表示

入力

参照: 入力画面

入力の送信勧誘 73

入力の読み取り 76

無保護の入力および出力/入力フィールドの消去 66

リスト

参照: リスト画面

アプリケーション画面 (続き)

レコードの配置 49

レコードまたはフィールドの保持 66

レコード様式のオーバーレイおよび消去 53

レコード様式の変更 52

2 行間のデータのロール 59

アプリケーションのポップアップ・ウィンドウ

定義 393

アプリケーション表示

印刷 40

機能、レコード様式内での定義 23

サンプルの作成 3

システム・パフォーマンスの向上 295

ディスプレイ・ファイル内での定義 17

ビュー、レコード様式の 24

フィールド、レコード様式内での定義 23

アプリケーションへのサブファイル内のカーソル位置の返送 72

アプリケーションへのサブファイルのモードの返送 72

アプリケーションへの制御戻り

(RETURN) ダイアログ・コマンド 719

アプリケーション・ウィンドウ

定義 393

アプリケーション・ハンドル定義 365

アプリケーション・フォーマット済みデータ

選択文字 519

アプリケーション・フォーマット済み領域 (APPFMT) タグ

アプリケーション・フォーマット済みデータ 519

概要 518

例

アプリケーション・フォーマット済み領域 520

アプリケーション・プログラミング

確認処理 333

データ表示パネル 347

メニュー・バー・パネル 363

リスト・パネル 327

アプリケーション・プログラミング・インターフェース (API)

ウィンドウの追加および除去 393

サービス 307

アプリケーション・プログラム

オーバーライド

HLL でのディスプレイ・ファイル属性 235

HLL でのディスプレイ・ファイル名 237

HLL でのディスプレイ・ファイル名および属性 238

コンパイル時のオーバーライドの適用 238

それによって使用されるレコード様式 24

ディスプレイ・ファイルの使用 45

内部または外部でのデータの記述 19

表示データの受け渡し 283

開く 365

ファイル・リソースの割り振り、リモート・システム 45

アプリケーション・ヘルプ

終了、System/36 アプリケーション互換性 772

定義 403

複数のディスプレイ装置に関する注意 403

参照: オンライン・ヘルプ情報

アプリケーション・レコード

MNUBARDSP 209

アラーム

メッセージに関する鳴動 253

アンバーサンド (&) 記号 515

位置情報、表示 472

移動

ウィンドウ 137

カーソル

カーソル・キーの使用 209

タブ・キーの使用 208

CSRINPONLY キーワード 31

印刷

印刷パネル 398

印刷ヘッド・パネル 398

概念 396

概要 396

画面 40

情報域 588

トレーラー 398

ページ排出 399

リスト域 608

2 バイト文字セット (DBCS) の考慮事項 400

印刷時のコード・ページおよび文字セットについての考慮事項 369

- 印刷タイトル行
 - 例 673
- 印刷中のページ排出 399
- 印刷パネル (PRTPNL) タグ
 - 概要 673
 - 許可されるタグの表 674
- 印刷パネルの印刷 398
- 印刷ヘッド・パネル (PRTHEAD) タグ
 - 概要 668
 - 許可されるタグの表 669
 - タイトル行のレイアウトの表 671, 672
 - 例、印刷タイトル行 673
- 印刷ヘッド・パネルの印刷 398
- インポート (IMPORT) タグ
 - 概要 586
- ウィンドウ
 - 移動 137
 - カーソル位置 131
 - からのデータの読み取り 136
 - グリッド・ライン構造 228
 - サイズ 130
 - 作成 129
 - システム・パフォーマンス考慮事項 298
 - 除去 138
 - シリーズから 1 つを飛び出させる 138
 - タイトル 135
 - 定義 41, 393
 - で使用されるサブファイル 132
 - 内のメッセージ 132
 - に表示されるヘルプ 132, 134
 - 場所 130, 131
 - パフォーマンス 139
 - 表示される最大値 129
 - 複写 137
 - プログラミング例 141
 - 別のパネルへの追加 389
 - 変更 136
 - 枠 133
 - 2 つを活動状態にする 137
- ウィンドウ参照レコード 129
- ウィンドウ定義レコード 129
- ウィンドウ内のカーソル 131
- ウィンドウの位置 130, 131
- ウィンドウの除去 (RMVWDW) キーワード 138
- ウィンドウの配置 130, 131
- ウィンドウの場所 130, 131
- ウィンドウの複写 137
- ウィンドウの枠 133
- ウィンドウ枠 (WDWBORDER) キーワード 133
- ウィンドウ・タイトル
 - 例、DDS 135
- ウィンドウ・タイトル (WDWTITLE) キーワード 135
- 受け渡しレコード (PASSRCD) キーワード 284
- 埋め込み
 - ネストされるソース・ファイル 517
- 英字変換
 - 名前構文 515
- エラー
 - 回復、実行時 93
 - からの回復 257
 - スクロール中の 379
 - フィールド妥当性検査 77
 - 参照：メジャー/マイナー戻りコード
 - 参照：メッセージ
- エラーからの回復
 - 参照：メジャー/マイナー戻りコード
- エラー条件 379
- エラー・メッセージ
 - ウィンドウ 132
 - ダイアログ変数 367
- エラー・メッセージ (ERRMSG) キーワード 132, 248
- エラー・メッセージ識別コード (ERRMSGID) キーワード 132, 248
- エラー・メッセージ・サブファイル (ERRSFL) キーワード
 - から受け取るメッセージ 132
 - 使用 250
 - 例の DDS ソース 250
 - USRDSPMGT キーワードでの制限 774
- 演算子
 - 関係 523
 - 条件 540
- オーバーライド
 - オーバーライド削除 (DLTOVR) コマンド 240
 - オーバーライド表示 (DSPOVR) コマンド 240
 - 同じファイル・タイプ 240
 - 異なるファイル・タイプ 240
 - 削除 240
 - 使用するかしないかの判別 235
 - 制御言語プログラム
 - 含める 235
 - QCMDEXC、使用 235
 - 属性 61
 - そのプロパティ 235
 - 他のジョブに対する影響 235
 - ディスプレイ・ファイル・オーバーライド (OVRDSPF) コマンド
 - ファイル属性 235
 - ファイル名 237
 - ファイル名および属性 238
 - 適用のタイミング 235
- オーバーライド (続き)
 - 入力 235
 - パフォーマンスに関する考慮事項 296
 - 表示 240
 - ファイルのリダイレクト
 - 同じファイル・タイプ 240
 - 組み合わせに対するデフォルトの表 242
 - 異なるファイル・タイプ 240
 - 定義 240
 - 入出力に有効 241
 - レベル検査 (LVLCHK) 240
 - プログラムのコンパイル時の適用 238
 - 無視または制限するコマンド 243
 - 有効である期間 235
- 例
 - 異なるディスプレイ・ファイル 237
 - 異なるディスプレイ・ファイルおよび属性 238
 - ディスプレイ・ファイルの属性 236
 - プログラムのコンパイル時の適用 239
 - HLL プログラムでのディスプレイ・ファイル属性 235
 - HLL プログラムでのディスプレイ・ファイル名 237
 - HLL プログラムでのディスプレイ・ファイル名および属性 238
- オーバーライド削除 (DLTOVR) コマンド 240
- オーバーライド表示 (DSPOVR) コマンド 240
- オープン操作
 - 情報の入手 47
 - ジョブ内で共用されるファイル 92
 - 定義 46
 - ファイル・リソース割り振り 45
 - ILE COBOL ステートメント 91
 - ILE C/C++ 関数 91
 - ILE RPG 操作 91
- オープン・データ・パス (ODP)
 - 共用、同じジョブの 296
 - システム・パフォーマンスの向上のため 296
 - 定義 92
- オープン・フィードバック域
 - 個別の説明 721
 - 装置定義リスト 724
 - 定義 47
 - 内容 47
- 応答時間
 - ウィンドウ・アプリケーション 139
 - リスト画面 374
 - リスト画面の 374

- 応答標識
 - 機能キー、それとの使用 27
 - 定義 27
 - 同時に使用可能にする 30
 - レコード域からの除去 30
 - CAnn キー、それとの使用 28
 - CFnn キー、それとの使用 28
 - HLPAMDKEY キーワード、使用 419
 - ROLLDOWN キーワード、それとの使用 30
 - ROLLUP キーワード、それとの使用 30
 - SETOF または SETOFF キーワード、それとの使用 31
 - USRDSMGT での制限 775
- 押しボタン
 - 作成 183
 - 定義 182
 - 例、DDS 183
- 押しボタン選択項目 (PSHBTNCHC) キーワード 183
- 押しボタン・フィールド (PSHBTNFLD) キーワード 183
- オブジェクト
 - 作成 305
 - 定義 17
 - パネル・グループ 513
- オブジェクト名
 - 名前構文
 - 制約事項 515
- オブジェクト割り振り (ALCOBJ) コマンド 45
- オプション行 (OPTLINE) タグ
 - 概要 647
- オプション行、リスト画面 476
- オプション標識
 - パネル・グループ、定義 410
 - 例
 - オプション標識を使用した HLPPNLGRP キーワード 410
 - オプション標識を使用した HLPRCD キーワード 412
 - 2 次ヘルプ 417
 - MNUBARHC (メニュー・バー選択項目) キーワード 158
 - レコード、定義 412
 - レコード域からの除去 30
- オフに設定 (SETOF または SETOFF) キーワード
 - その応答標識 31
- 折り返し
 - 語 35, 191
- オンライン・ヘルプ情報
 - カーソル位置に基づいて表示されるヘルプ・テキスト 380
 - 概要 422
- オンライン・ヘルプ情報 (続き)
 - 索引検索
 - アクセス 423
 - 索引検索機能
 - 機能 442
 - 同義語、トピック用の選択 445
 - トピック、定義 444
 - トピックの決定 444
 - の UIM タグ 442
 - ルート・ワード、定義 444
 - ルート・ワード、同義語として組み込む 444
 - ルート・ワード、トピック用の選択 445
 - ルート・ワードの同義語、指定 444
 - 参照：InfoSeeker 機能
 - サンプル画面の作成 3
 - 終了、System/36 アプリケーション互換性 772
 - 情報画面
 - 説明 497
 - ヘルプ域 497
 - DDS の考慮事項 498
 - 説明 495
 - 全般ヘルプ 405
 - ディスプレイへの追加 403
 - 入力画面
 - 説明 496
 - タイプ 496
 - ヘルプ域 496
 - DDS の考慮事項 497
 - ハイパーテキスト
 - 参照句、定義 449
 - 定義 448
 - リンクの作成 449
 - リンクの設計 448
 - パネル・グループ 442, 444
 - 同じファイル内の文書およびレコード 410
 - 同じファイル内のレコードおよび文書 410
 - オプション標識、指定 410
 - オプション標識を使用した HLPPNLGRP キーワード 410
 - 検索索引からの項目の除去 451
 - 検索索引の削除 452
 - 検索索引の作成 451
 - 検索索引への項目の追加 451
 - コマンド・ヘルプ、割り当て 450
 - 索引検索、使用可能化 407, 423
 - 索引検索機能 442
 - 削除 450
 - 作成 450
 - 使用 437
 - 全画面ヘルプ、指示 407
- オンライン・ヘルプ情報 (続き)
 - パネル・グループ (続き)
 - 段落と注記、ヘルプ・モジュール中の定義 439
 - 同義語、索引検索トピック用の選択 445
 - トピック、定義 444
 - ハイパーテキスト 448
 - ハイパーテキスト参照句 449
 - ハイパーテキスト・リンクの作成 449
 - ハイパーテキスト・リンクの設計 448
 - プログラミング・キーワードおよびプログラミング変数、ヘルプ・モジュールでの識別 441
 - 文書とレコードの比較 404
 - 別のものに含まれるヘルプ・モジュールの使用 438
 - ヘルプ、指定 407
 - ヘルプ画面の名前の定義 407
 - ヘルプ・モジュール、複数回使用 438
 - ヘルプ・モジュール、リンク 448
 - ヘルプ・モジュール中の構造化テキスト、指示 441
 - ヘルプ・モジュール中のテキストの強調とフォーマット設定 439
 - ヘルプ・モジュール中のテキストの強調表示 440
 - ヘルプ・モジュール中のリストの作成 440
 - ヘルプ・モジュールを使用した編成 438
 - 見出しのヘルプ・モジュールへの追加 440
 - ユーザー定義、使用可能化 423
 - ルート・ワード、索引検索同義語として組み込む 444
 - ルート・ワード、索引検索トピック用の選択 445
 - ルート・ワード、定義 444
 - ルート・ワードの同義語、指定 444
 - 2 次ヘルプ、除外 407
 - DDS 機能 407
 - HLPPNLGRP キーワード 409
 - UIM ソースの入力 437
 - UIM タグの目的 437
 - 文書、オフィス
 - パネル・グループとレコードの比較 404
 - 文脈ヘルプ 405
 - ヘルプ域
 - 定義 405
 - パネル・グループ、指定 407

オンライン・ヘルプ情報 (続き)

ヘルプ域 (続き)

必要な DDS ソース 405

レコード 406

レコード、指定 411

DDS キーワード 406

ヘルプ・キー

押さずにプログラムに制御を戻す
420

押した後にプログラムに制御を戻す
419

使用可能 404

ヘルプ・キーの使用可能化 404

ヘルプ・キーを押さずにプログラムに
制御を戻す 420

ヘルプ・キーを押した後にプログラム
に制御を戻す 419

メニュー画面

タイプ 495

ヘルプ域 495

DDS の考慮事項 496

ユーザー定義パネル・グループ

アクセス 423

リスト画面

説明 498

ヘルプ域 498

DDS の考慮事項 499

例

インポートされるヘルプ・モジュール
439

応答標識を使用した HLPCMDKEY
キーワード 419

オプション標識を使用した
HLPPNLGRP キーワード 410

オプション標識を使用した
HLPRCD キーワード 412

コマンド・ヘルプ 450

ディスプレイ・ファイルによる作成
3

パネル・グループ中に組み込まれる
ヘルプ・モジュール 438

ヘルプ域を表示する画面 406

ヘルプ・モジュールを使用したパネ
ル・グループ 438

2 次ヘルプ 417

HLPCMDKEY キーワード 419

HLPPNLGRP キーワード 409

HLPRCD キーワード 412

ISCH タグ 444

ISCHSYN タグ 444

LINK タグ 449

レコード、DDS

応答標識を使用した HLPCMDKEY
キーワード 419

同じファイルでのパネル・グループ
の使用 411

オンライン・ヘルプ情報 (続き)

レコード、DDS (続き)

同じファイルでの文書の使用 413

オプション標識、指定 412

オプション標識を使用した

HLPRCD キーワード 412

制約 414

ソースの入力 413

パネル・グループと文書の比較
404

ヘルプ、指定 411

ヘルプ、の定義 452

ヘルプ画面間でのページング 415

ヘルプ境界 416

ヘルプのサブリスト 416

ヘルプ・グループ 416

ヘルプ・リスト 415

1 次ヘルプ 415

2 次、順序の決定 416

2 次ヘルプ 415, 417

2 次ヘルプ、表示 416

2 次ヘルプに関する制約事項 418

H 仕様を使用したページング 415

HLPCMDKEY キーワード 419

HLPRCD キーワード 412

H 仕様

定義 405

DDS ソースでの順序 406

DDS ソースの入力 406

[力行]

カーソル

アプリケーションへの位置の返送 72

位置決め

サブファイル 118

サブファイル・レコードの個々のペ
ージ 117

出力操作後 71

入出力操作中 83

位置指定

ディスプレイ・ファイルでの

System/36 アプリケーション互換
性 767

リスト画面の規則 484

移動

カーソル・キー 209

垂直項目選択フィールド 164

水平選択フィールド 165

タブ・キー 208

入力可能位置にのみ 31

右から左 31

移動先

入力フィールド 33

処置 207

カーソル (続き)

制限

PULLDOWN キーワード 172

制約

WINDOW キーワード 131

ヘルプ・テキストが表示されるときの
位置 380

カーソル位置に基づいて表示されるヘル
プ・テキスト 380

カーソル位置戻し (RTNCSRLOC) キーワ
ード 72

カーソル移動先フィールド (FLDCSRPRG)
キーワード 33

カーソルの制限

PULLDOWN キーワード 172

カーソルの制約

WINDOW キーワード 131

カーソル・キー

移動 209

解決

HTML フィールドのオーバーラップ
229

開始行番号 (SLNO) キーワード

値と結果の表 54

使用 54

制約事項 55

他の DDS キーワードと一緒に使用
55

*VAR 値を指定 55

開始属性文字

定義 24

階層、索引検索 446

ガイドライン

操作

情報画面 474

入力画面 470

メニュー画面 464

リスト画面 479

メッセージ 500

CUA、単一選択フィールド操作 503

外部記述データ

使用 19

その利点 19

定義 19

解放操作

定義 90

ILE COBOL ステートメント 91

ILE C/C++ 関数 91

ILE RPG 操作 91

書き込み操作 108

サブファイル

定義 108

BASIC ステートメント 109

ILE COBOL ステートメント 109

ILE C/C++ 関数 109

ILE RPG 演算命令 109

書き込み操作 (続き)

定義 49

読み取り要求の実行時までの据え置き
68

CL コマンド 91

ILE COBOL ステートメント 91

ILE C/C++ 関数 91

ILE RPG 操作 91

書き込み・読み取り操作

待機を指定した

CL コマンド 91

ILE C/C++ 関数 91

ILE RPG 操作 91

待機を指定しない、CL コマンド 91

隠しフィールド

使用 25

定義 25

拡張英字

定義 289

拡張処置項目域

リスト画面 478

拡張表示 (ENHDSP) パラメーター

レコードの書き込み 153

拡張ヘルプ (EXTHELP) ダイアログ・コマンド 710

拡張ヘルプの見出し (XH1 から XH4) タグ

概要 696

フォーマットの規則 696

ヘルプ・モジュールでの使用 440

例、サンプル見出し 697

獲得されるディスプレイ装置 46

追加、入出力操作 46

入出力操作からの解放 90

獲得操作

定義 46

ファイル・リソースの割り振り 45

ILE COBOL ステートメント 91

ILE C/C++ 関数 91

ILE RPG 操作 91

確認処理

アプリケーション・プログラミング
333

自動的に 332

確認パネルのソース 330

確認リスト・パネル、作成 329

各国語に関する考慮事項および

DBCS 447

各国語の翻訳

文末文字 512

活動サブファイル・レコード

定義 98

可変長レコード 113

画面印刷 (PRINT) ダイアログ・コマンド
716

画面サイズ

条件名 39

それによるパフォーマンスの向上 295

代替 69

有効な 39

例

2 つのための DDS 39

2 つのためのサブファイル用の

DDS 40

1 次および 2 次 40

画面サイズ (DSPSIZ) キーワード 39

画面設計機能 (SDA)

定義 8

DDS ソースの直接入力との比較 20

画面の復元 (RSTDSP) パラメーター

システム・パフォーマンス考慮事項
297

必要なキーワード 297

画面復元 (RSTDSP) パラメーター

使用 85

例 85

カラー、画面に追加 37

カラー属性 187

関係演算子 523

漢字 (IGC)

書式設定

BOTINST (最下部指示) タグ 521

DATA C (データ項目選択) タグ
555

TOPINST (最上部指示) タグ 688

フォーマット

PDFLDC (プルダウン・フィールド
選択項目) タグ 663

関数、組み込み

CHKOBJ (オブジェクト検査) 541,
600

CHKPGM (COND 評価プログラ
ム) 542

CHKPGM (条件評価プログラム) 601

CHKUSRCLS (ユーザー・クラス検
査) 542, 601

管理

パネル機能 377

キー

機能

定義 27

System/36 アプリケーション互換性
771

コマンド

妥当性検査の考慮事項 87

定義 28

コマンド機能 (CFnn) 28

コマンド・アテンション (CAnn) 28

取り消しおよび終了、メニューでの使
用 262

CAnn および CFnn、定義 28

キー (続き)

CFnn (コマンド機能)、妥当性検査の考
慮事項 87

キー定義

例 595

キーボード

入出力操作中のロック 83

プログラムによるデータ処理中のロッ
ク解除 77

ロック 54

キーボード機能

継続入力フィールド 192

キーボードのロックおよびカーソルの位置
決め 83

キーワード、DDS 54, 69, 111, 283

ウィンドウ・タイトル
(WDWTITLE) 135

エラー・サブファイル (ERRSFL)
説明 250

例の DDS ソース 250

エラー・メッセージ (ERRMSG) 248

エラー・メッセージ識別コード
(ERRMSGID) 248

押しボタン選択項目

(PSHBTNCHC) 183

押しボタン・フィールド

(PSHBTNFLD) 183

オフに設定 (SETOF または
SETOFF) 31

カーソル位置戻し (RTNCSRLOC) 72

カーソル移動先フィールド

(FLDCSRPRG) 33

開始行番号 (SLNO)

使用 54

*VAR 値を指定した例 55

画面サイズ (DSPSIZ)

使用 39

機能キー、それによる定義 27

繰り返し入力されるデータ 65

グリッド消去 (GRDCLR) 227

グリッド・ボックス (GRDBOX) 227

グリッド・ライン (GRDLIN) 227

グリッド・ライン属性

(GRDATR) 227

グリッド・レコード (GRDRCD) 227

語の折り返し (WRDWRAP) 35, 191

サブファイル行 (SFLIN) 98

サブファイル切り捨て

(SFLDROP) 98

サブファイル継続 (SFLFOLD) 98

サブファイル削除 (SFLDLT) 98

サブファイル実行キー

(SFLENTER) 98

サブファイル終了 (SFLEND) 98

サブファイル消去 (SFLCLR) 98

サブファイル初期設定 (SFLINZ) 98

キーワード、DDS (続き)

- サブファイル制御 (SFLCTL) 97
- サブファイル選択制御 (SFLCHCCTL) 175
- サブファイル単一選択項目選択 (SFLSNGCHC) 174
- サブファイル表示 (SFLDSP) 97
- サブファイル表示制御 (SFLDSPCTL) 98
- サブファイル戻り選択済み項目 (SFLRTNSEL) 176
- サブファイル・カーソル移動先 (SFLCSRPRG) 33
- サブファイル・カーソル相対レコード番号 (SFLCSRRRN) 72
- サブファイル・サイズ (SFLSIZ)
 - サブファイル制御レコード様式での使用 97
 - ROLLUP および SFLPAG の指定例 112
- サブファイル・スクロール (SFLSCROLL) 176
- サブファイル・プログラム・メッセージ待ち行列 (SFLPGMQ)
 - 説明 98
 - プログラム・メッセージ待ち行列からメッセージを表示する 250
- サブファイル・ページ (SFLPAG)
 - サブファイル制御レコード様式での使用 97
 - ROLLUP および SFLSIZ の指定例 112
- サブファイル・メッセージ (SFLMSG)
 - サブファイル制御レコードの書き込み時にメッセージ行に表示する 248
 - 説明 98
 - 定義 98
- サブファイル・メッセージ識別コード (SFLMSGID)
 - サブファイル制御レコードの書き込み時にメッセージ行に表示する 248
 - サンプル DDS 251
 - 説明 98
 - 定義 98
- サブファイル・メッセージ・キー (SFLMSGKEY)
 - 説明 98
 - プログラム・メッセージ待ち行列からメッセージを表示する 250
- サブファイル・メッセージ・レコード (SFLMSGRCD)
 - 説明 98
 - プログラム・メッセージ待ち行列からメッセージを表示する 250

キーワード、DDS (続き)

- サブファイル・モード (SFLMODE) 72
- サブファイル・レコード番号 (SFLRCDNBR)
 - サブファイル・パフォーマンスの向上 295
 - 説明 98
- サブファイル・レコード非活動 (SFLRNA) 98
- サブファイル・ロール値 (SFLROLVAL) 98
- 前ページ (PAGEUP)
 - 定義 27
- 属性一時変更 (OVRATR)
 - システム・パフォーマンス考慮事項 296
 - 使用 61
- 代替ヘルプ (ALTHELP) 29
 - 妥当性検査 79
- 次に変更するサブファイル (SFLNXTCHG)
 - 説明 98
 - 例 115
- 次ページ (PAGEDOWN) 27
- データ一時変更 (OVRDTA) 63
 - システム・パフォーマンス考慮事項 296
- データ保持 PUT (PUTRETAIN)
 - 説明 66
- 入力可能位置にのみカーソル移動 (CSRINPONLY) 31
- 入力消去 (ERASEINP)
 - システム・パフォーマンス考慮事項 297
 - 使用 66
- 標識域 (INDARA) 30
- 表示属性 (DSPATR)
 - 値の表 36, 37
 - フィールドにカラーを加える方法 37
 - フィールドの強調方法 36
- ASSUME キーワードと併用 283
 - PC 値 71, 118
- 表示モードが変更される場合に無視されるキーワード 70
- フィールド
 - カラー (表) 37
 - 強調 (表) 36
- フィールド・レベル 67
- プログラマブル・マウス・ボタン (MOUBTN) 216
- ヘルプ域 (HLPARA) 406
- ヘルプからの戻り (HLPRTN) 420
- ヘルプ境界 (HLPBDY)
 - 使用 416

キーワード、DDS (続き)

- ヘルプの検索索引 (HLPSCIDX) 407
- ヘルプの順序付け (HLPSEQ) 416
- ヘルプの除外 (HLPEXCLD) 407
- ヘルプの全画面表示 (HLPFULL) 407
- ヘルプのタイトル (HLPPTITLE) 407
- ヘルプ・コマンド・キー (HLPCMDKEY)
 - 使用 419
- ヘルプ・パネル・グループ (HLPPNLGRP)
 - オプション標識の使用 410
 - 説明 407
- ヘルプ・レコード (HLPRCD)
 - オプション標識の使用 412
 - 説明 411
- 右から左へ表示 (DSPRL) キーワード 34
- 明示的な一時変更を伴う PUT (PUTOVR)
 - システム・パフォーマンス考慮事項 296
 - 説明 61
- より効果的なコーディング 63
- レコード初期設定 (INZRCD)
 - 使用 77
- ロールアップ (ROLLUP)
 - 応答標識 30
 - 機能キー用 27
 - サブファイル・パフォーマンスの向上 295
 - SFLSIZ および SFLPAG の指定例 112
- ロールダウン (ROLLDOWN)
 - 応答標識 30
 - 定義 27
- ALTHELP (代替ヘルプ) 29
- ALTPAGEDWN (代替次ページ) 29
- ALTPAGEUP (代替前ページ) 29
- ALWROL (ロール可能)
 - 使用 59
 - 例 60
- ASSUME (想定)
 - 使用 283
 - DDS ソース例 283, 284
 - DFT キーワードと併用 283
- CAnn (コマンド・アテンション) 27
- CFnn (コマンド機能) 27
- CHANGE (変更) 78
- CHECK (検査)
 - 記述 31
 - AB 値 79
 - M10 値 79
 - M11 値 79
 - ME 値 79
 - MF 値 79

- キーワード、DDS (続き)
- CHECK (検査) (続き)
 - VN 値 79
 - VNE 値 79
 - CHRID (文字 ID) キーワード 289
 - CLEAR (消去) 27
 - CLRL (行の消去) 57
 - システム・パフォーマンス考慮事項 296
 - OVERLAY を指定した例 58
 - *NO 値を指定した例 57
 - COLOR (カラー) 37
 - COMP (比較) 79
 - CSRINPONLY (入力可能位置にのみカーソル移動) 31
 - CSRLOC (カーソル位置) 71
 - DFT (デフォルト)
 - 説明 69
 - ASSUME キーワードと併用 283
 - DFTVAL (デフォルト値) 69
 - DSPATR (表示属性)
 - 値の表 36
 - カラー効果の表 37
 - フィールドにカラーを加える方法 37
 - フィールドの強調方法 36
 - ASSUME キーワードと併用 283
 - PC 値 71, 118
 - DSPMOD (表示モード)
 - サブファイルの例 111
 - 説明 70
 - *DS3 のサブファイル・ディスプレイ 112
 - *DS4 のサブファイル・ディスプレイ 111
 - DSPRL (右から左へ表示) キーワード 34
 - DSPSIZ (画面サイズ)
 - 使用 39
 - パフォーマンスの向上のために指定 295
 - EDTCDE (編集コード) 38
 - EDTWRD (編集語) 38
 - ERASE (消去)
 - 出力操作に対する影響 54
 - OVERLAY を指定した影響の例 54
 - ERASEINP (入力消去)
 - システム・パフォーマンス考慮事項 297
 - 使用 66
 - ERRMSG (エラー・メッセージ) 248
 - ERRMSGID (エラー・メッセージ識別コード) 248
 - ERRSFL (エラー・サブファイル)
 - 説明 250
- キーワード、DDS (続き)
- ERRSFL (エラー・サブファイル) (続き)
 - 例の DDS ソース 250
 - FLDCSRPRG (カーソル移動先フィールド) 33
 - GETRETAIN (データ保持 GET) 78
 - GRDATR (グリッド・ライン属性) 227
 - GRDBOX (グリッド・ボックス) 227
 - GRDCLR (グリッド消去) 227
 - GRDLIN (グリッド・ライン) 227
 - GRDRCD (グリッド・レコード) 227
 - H 仕様
 - 参照: H 仕様
 - HELP (ヘルプ)
 - 機能キーの使用 27
 - ヘルプ・キーの使用可能化 404
 - HLPARA (ヘルプ域) 406
 - HLPBDY (ヘルプ境界)
 - 使用 416
 - 2 次ヘルプ 417
 - HLPCMDKEY (ヘルプ・コマンド・キー)
 - 応答標識を使用したサンプル・ソース 419
 - 使用 419
 - DDS ソースのサンプル 419
 - HLPEXCLD (ヘルプの除外) 407
 - HLPFULL (ヘルプの全画面表示) 407
 - HLPPNLGRP (ヘルプ・パネル・グループ)
 - オプション標識の使用 410
 - オプション標識を使用したサンプル・ソース 410
 - 説明 407
 - DDS ソースのサンプル 409
 - HLPRCD (ヘルプ・レコード)
 - オプション標識の使用 412
 - オプション標識を使用したサンプル・ソース 412
 - 説明 411
 - 2 次ヘルプ 417
 - DDS ソースのサンプル 412
 - HLPRTN (ヘルプからの戻り) 420
 - HLPSCHIDX (ヘルプの検索索引) 407
 - HLPSEQ (ヘルプの順序付け) 416
 - HLPTITLE (ヘルプのタイトル) 407
 - HOME (ホーム) 27
 - HTML (ハイパーテキスト・マークアップ言語)
 - 使用 229
 - 制約事項 233
 - フィールドのオーバーラップの解決 229
 - INDARA (標識域) 30
- キーワード、DDS (続き)
- INVITE (送信勧誘)
 - システム・パフォーマンス考慮事項 297
 - 説明 73
 - INZINP (入力初期設定)、システム・パフォーマンス考慮事項 297
 - INZRCD (レコード初期設定)
 - 使用 77
 - UNLOCK キーワードを指定した例 79
 - KEEP (保持)
 - 説明 283
 - DDS ソース例 284
 - LOCK (ロック) 84
 - MDTOFF (変更データ・タグ・オフ) 66
 - MOUBTN (プログラマブル・マウス・ボタン) 216
 - MSGALARM (メッセージ・アラーム) 253
 - MSGID (メッセージ識別コード) 249
 - OVERLAY (オーバーレイ)
 - システム・パフォーマンス考慮事項 297
 - 出力操作に対する影響 54
 - 説明 53
 - CLRL を指定した例 58
 - ERASE を指定した影響の例 54
 - OVRATR (属性一時変更)
 - システム・パフォーマンス考慮事項 296
 - 使用 61
 - OVRDTA (データ一時変更) 63
 - システム・パフォーマンス考慮事項 296
 - PAGEDOWN (次ページ) 27
 - PAGEUP (前ページ) 27
 - PASSRCD (受け渡しレコード) 284
 - PRINT (印刷)
 - 得られる結果 (表) 40
 - 機能キー用 27
 - 使用可能化 40
 - PSHBTNCHC (押しボタン選択項目) 183
 - PSHBTNFLD (押しボタン・フィールド) 183
 - PUTOVR (明示的な一時変更を伴う PUT) 63, 65
 - システム・パフォーマンス考慮事項 296
 - 説明 61
 - 例 62
 - PUTRETAIN (データ保持 PUT) 67
 - 説明 66
 - レコード・レベル 67

キーワード、DDS (続き)

RANGE (範囲) 79

ROLLDOWN (ロールダウン)

応答標識 30

定義 27

ROLLUP (ロールアップ)

機能キー用 27

サブファイル・パフォーマンスの向上 295

その応答標識 30

SFLSIZ および SFLPAG の指定例 112

RTNCSRLOC (カーソル位置戻し) 72

SETOF または SETOFF (オフに設定) 31

SFL (サブファイル) 97

SFLCHCCTL (サブファイル選択制御) 175

SFLCLR (サブファイル消去) 98

SFLCSRPRG (サブファイル・カーソル移動先) 33

SFLCSRPRN (サブファイル・カーソル相対レコード番号) 72

SFLCTL (サブファイル制御) 97

SFLDLT (サブファイル削除) 98

SFLDROP (サブファイル切り捨て) 98

SFLDSP (サブファイル表示) 97

SFLDSPCTL (サブファイル表示制御) 98

SFLEND (サブファイル終了) 98

SFLENTER (サブファイル実行キー) 98

SFLFOLD (サブファイル継続) 98

SFLINZ (サブファイル初期設定) 98

SFLLIN (サブファイル行) 98

SFLMODE (サブファイル・モード) 72

SFLMSG (サブファイル・メッセージ)
サブファイル制御レコードの書き込み時にメッセージ行に表示する 248

説明 98

定義 98

SFLMSGID (サブファイル・メッセージ識別コード)

サブファイル制御レコードの書き込み時にメッセージ行に表示する 248

サンプル DDS 251

説明 98

定義 98

SFLMSGKEY (サブファイル・メッセージ・キー)

説明 98

キーワード、DDS (続き)

SFLMSGKEY (サブファイル・メッセージ・キー) (続き)

プログラム・メッセージ待ち行列からメッセージを表示する 250

SFLMSGRC (サブファイル・メッセージ・レコード)

説明 98

プログラム・メッセージ待ち行列からメッセージを表示する 250

SFLNXTCHG (次に変更するサブファイル)

説明 98

例 115

SFLPAG (サブファイル・ページ)

サブファイル制御レコード様式での使用 97

ROLLUP および SFLSIZ の指定例 112

SFLPGMQ (サブファイル・プログラム・メッセージ待ち行列)

説明 98

プログラム・メッセージ待ち行列からメッセージを表示する 250

SFLRCDNBR (サブファイル・レコード番号)

サブファイル・パフォーマンスの向上 295

説明 98

SFLRNA (サブファイル・レコード非活動) 98

SFLROLVAL (サブファイル・ロール値) 98

SFLRTNSEL (サブファイル戻り選択済み項目) 176

SFLSCROLL (サブファイル・スクロール) 176

SFLSIZ (サブファイル・サイズ)
サブファイル制御レコード様式での使用 97

ROLLUP および SFLPAG の指定例 112

SFLSNGCHC (サブファイル単一選択項目選択) 174

SLNO (開始行番号)

使用 54

*VAR 値を指定した例 55

UNLOCK (ロック解除)

入力操作に対する影響 79

プログラムによるデータ処理中のアンロック 77

INZRCD キーワードを指定した例 79

USRDFN (ユーザー定義) 277

VALNUM 79

VALUES (値) 79

キー割り当て、共通

参照: 共通キーの割り当て

キー・リスト (KEYL) タグ

概要 596

許可されるタグの表 596

例、キー・リスト 596

キー・リスト項目 (KEYI) タグ

概要 593

例、キー定義 595

記述

ファイル・レベル

使用 19

定義 19

フィールド・レベル

使用 18

定義 18

レコード・レベル

使用 18

定義 18

記述、ファイル

決定 18

定義 3

ディスプレイ・ファイル用、作成 20

変更

その理由 18

CHGDISPF コマンドによる 21

変更の検出 22

記述テキスト域、入力画面 469

規則

推奨

プルダウン・フィールド選択項目 (PDFLDC) タグ 664

推奨される

リスト・アクション (LISTACT) タグ 619

入力フィールド 468

必須

プルダウン・フィールド選択項目 (PDFLDC) タグ 663

必須の

リスト・アクション (LISTACT) タグ 619

編集記述の使用法 760

リスト画面

エラー条件 485

カーソル位置 484

機能キー

コマンドの保持、System/36 アプリケーション互換性 773

定義

DDS における 27

UIM 内 383

割り当て、テーブル 490

CUA 基本ダイアログ処置、その表 501
DDS キーワード 27

- 機能キー (続き)
 - i5/OS スタイル画面での共通割り当て 490
 - i5/OS スタイルの画面に任意指定、その表 462
 - i5/OS スタイルの画面に必須、その表 462
 - System/36 アプリケーション互換性 771
 - UIM での書式設定 383
 - UIM での処理 384
 - UIM でのスクロール 378
 - VARUPD 値 384
- 機能キー域
 - メニュー表示、単一選択 462
 - CUA 502
- 機能キー域のヘルプ 433
- 機能キーおよび VARUPD 値の処理 384
- 機能キー続き表示 (MOREKEYS) ダイアログ・コマンド 713
- 機能キー保持 (RETKEY) キーワード
 - System/36 アプリケーション互換性 774
- 基本ダイアログ処置、CUA の表 501 行
 - 消去、指定した数 57
 - データのロール 59
- 行、オプション 476
- 行、指示
 - 参照： 指示行
- 強調、フィールド 36
- 強調値表 627
- 強調表示
 - 選択文字 520, 682
- 強調表示およびフォント 398
- 強調表示句 (HP0 から HP9) タグ
 - 概要 582
 - ヘルプ・モジュールでの使用 440
- 強調表示句の表 583
- 共通キーの割り当て 490
- 共通フィールドバック域
 - 定義 47
- 共通ユーザー・アクセス (CUA)
 - 機能キー域 502
 - 基本ダイアログ処置、その表 501
 - システムとの関係 455
 - 情報画面 505
 - 説明 501
 - 選択項目 503
 - 選択項目入力フィールド 503
 - 単一選択フィールド操作、ガイドライン 503
 - 入力画面 504
 - 入力フィールド 504
 - メッセージ行の関係 502
 - メニュー画面、単一選択 502
- 共通ユーザー・アクセス (CUA) (続き)
 - リスト画面、例 506
- 行の消去 (CLRL) キーワード
 - システム・パフォーマンス考慮事項 296
 - 使用のタイミング 57
 - 例
 - OVERLAY との相違 58
 - (*NO) 値を指定 57
 - S/36 環境の考慮事項 764
- 共用
 - オープン・データ・パス 296
- 区切り記号
 - 内容 511
 - マークアップ 511
- 国別の選択文字 377
- 組み込み関数
 - CHKOBJ (オブジェクト検査) 541, 600
 - CHKPGM (COND 評価プログラム) 542
 - CHKPGM (条件評価プログラム) 601
 - CHKUSRCLS (ユーザー・クラス検査) 542, 601
- 組み込みヘルプ (IMHELP) タグ
 - 概要 584
 - 例、組み込みヘルプ 585
 - UIM ソース中の配置 438
- クラス定義 (CLASS) タグ
 - 概要 525
 - 許可されるタグの表 525
 - 数値の例の表 538
 - 表示画面形式
 - 時刻値 538
 - 数値 537
 - 日付値 538
 - 文字値 538
 - 方向付け用の決定木 534
 - 例
 - クラス定義 536
- BASETYPE 要約表 530
- グラフィカル・ユーザー・インターフェース (GUI)
 - サポート 378
- グラフィカル・ユーザー・インターフェース (GUI) の使用可能化
 - サポート 378
- グラフィックス
 - フィールド 517
 - リテラル 517
- グリッド消去 (GRDCLR) 227
- グリッド・ボックス (GRDBOX) キーワード 227
- グリッド・ライン (GRDLIN) 227
- グリッド・ライン構造
 - ウィンドウ 228
- グリッド・ライン構造 (続き)
 - 概要 226
 - 定義 226
 - ハードウェア要件 228
 - 例、DDS 227
- グリッド・ライン属性 (GRDATR) キーワード 227
- グリッド・レコード (GRDRCD) 227
- グループ・オブジェクト、パネル 513
- クローズ操作
 - ジョブ内で共用されるファイル 93
 - 定義 91
 - CL コマンド 91
 - ILE COBOL ステートメント 91
 - ILE C/C++ 関数 91
 - ILE RPG 操作 91
- 継続属性
 - 例 511
- 継続入力フィールド
 - キーボード機能 192
 - 後方フィールド出口処理 197
 - 語の折り返し 191
 - 作成 190
 - 前方フィールド出口処理 196
 - 例 190
 - DBCS に関する考慮事項 191
- 経路指定ステップ 377
 - 同じものの中にあるプログラム間でのデータの受け渡し 283
 - ステップ間でのデータの受け渡し 284
- 検索索引
 - 参照： オンライン・ヘルプ情報
- 検索索引開始 (STRSCHIDX) コマンド 443
- 検索索引項目削除 (RMVNSCHIDX) コマンド 452
- 検索索引項目除去 (RMVNSCHIDX) コマンド 451
- 検索索引項目追加 (ADDSCHIDX) コマンド 451
- 検索索引作成 (CRTSCHIDX) コマンド 451
- 原始ステートメント入力ユーティリティ (SEU)
 - 定義 6
 - 「Edit」画面 6
- コード、メジャー/マイナー戻り
 - 参照： メジャー/マイナー戻りコード
- コード化文字セット ID 変換なし (NOCCSID) キーワード 291
- コード・ページ
 - 代替画面 289
 - ディスプレイ・ファイルの CHRID キーワードの一部 289
 - ページ 420 533
 - ページ 424 533

コード・ページ (続き)
変換、UIM 532
コード・ページ変換の場合のページ
420 533
コード・ページ変換の場合のページ
424 533
向上
システム・パフォーマンス、ディスプレ
イでの 295
UIM による生産性の 301
向上、UIM による生産性の 301
更新
UIM 内のリスト 373
高水準言語 (HLL)
サポートされる入出力要求の認識 91
プログラムでのディスプレイ・ファイ
ル属性のオーバーライド 235
プログラムでのディスプレイ・ファイ
ル名および属性のオーバーライド
238
プログラムでのディスプレイ・ファイ
ル名のオーバーライド 237
System/36 アプリケーション互換性
763
後方フィールド出口処理 197
項目固有ヘルプ 205
固定フィールド
使用 25
定義 25
メッセージ記述からのテキストを挿入
31
語の折り返し (WRDWRAP) キーワード
35, 191
コマンド、CL
オーバーライド
制限するコマンド 243
そのパフォーマンスの向上 296
オーバーライド削除 (DLTOVR) 240
オーバーライド表示 (DSPOVR) 240
オブジェクト割り振り (ALCOBJ) 45
検索索引開始 (STRSCHIDX) 443
検索索引項目除去
(RMVSCHIDX) 451
検索索引項目追加
(ADDSCHIDX) 451
検索索引削除 (DLTSCHIDX) 452
検索索引作成 (CRTSCHIDX) 451
コマンド作成 (CRTCMD) 450
コマンド・デフォルト変更
(CHGCMDFFT) 275
受信終了 (ENDRCV) 74
ジョブ変更 (CHGJOB) 253
装置記述の変更 (CHGDEVVSP) 41
テーブル作成 (CRTTBL) 292

コマンド、CL (続き)
ディスプレイ・ファイル作成
(CRTDSPF)
画面の作成 7
ディスプレイ・ファイルの作成 20
ディスプレイ・ファイル・メニュー
268
プログラム・メニュー 271
DFRWRT パラメーター 295
SHARE パラメーター 296
ディスプレイ・ファイル変更
(CHGDSPF)
実行 21
ファイル記述の変更 21
SHARE パラメーター 296
ディスプレイ・ファイル・オーバーラ
イド (OVRDSPF)
ファイル属性 235
ファイル名 237
ファイル名および属性 238
CHGDSPF との比較 18
独自のものの作成 450
パネル・グループ削除
(DLTPNLGRP) 450
パネル・グループ作成 (CRTPNLGRP)
画面の作成 8
使用 450
ファイル記述表示 (DSPFD) 23
ファイル・フィールド記述表示
(DSPFFD) 22
複製オブジェクト作成 (CRTDUPOBJ)
使用 275
プログラム開発管理機能開始
(STRPDM) 3
プログラム参照表示
(DSPPGMREF) 23
ヘルプ、割り当て 450
編集記述作成 (CRTEDTD) 39
編集記述表示 (DSPEDTD) 39
メッセージ記述追加 (ADDMSGD) コ
マンド 269
メッセージ・ファイル作成
(CRTMSGF) 269
メニュー削除 (DLTMNU) 276
メニュー作成 (CRTMNU)
ディスプレイ・ファイル・メニュー
269
プログラム・メニュー 272
メニュー属性表示 (DSPMNUA) 276
メニュー表示へ (GO)
一般メニュー仕様 275
使用 261
メニュー変更 (CHGMNU) 276
読み取り操作の取り消し 91
ADDMSGD (メッセージ記述追
加) 269

コマンド、CL (続き)
ADDSCHIDX (検索索引項目追
加) 451
ALCOBJ (オブジェクト割り振り) 45
CHGCMDFFT (コマンド・デフォルト
変更) 275
CHGDEVVSP (装置記述の変更) 41
CHGDSPF (ディスプレイ・ファイル変
更)
ファイル記述の変更 21
DFRWRT パラメーター 295
SHARE パラメーター 296
CHGJOB (ジョブ変更) 253
CHGMNU (メニュー変更) 276
CL プログラム作成 (CRTCLPGM) コ
マンド
プログラム・メニュー 272
CRTCLPGM (CL プログラム作成)
プログラム・メニュー 272
CRTCMD (コマンド作成) 450
CRTDSPF (ディスプレイ・ファイル作
成)
画面の作成 7
ディスプレイ・ファイルの作成 20
ディスプレイ・ファイル・メニュー
268
プログラム・メニュー 271
DFRWRT (書き込み据え置き) パラ
メーター 295
SHARE パラメーター 296
CRTDUPOBJ (複製オブジェクト作成)
使用 275
CRTEDTD (編集記述作成) 39
CRTMNU (メニュー作成)
ディスプレイ・ファイル・メニュー
269
プログラム・メニュー 272
CRTMSGF (メッセージ・ファイル作
成) 269
CRTPNLGRP (パネル・グループ作成)
画面の作成 8
使用 450
CRTSCHIDX (検索索引作成) 451
CRTTBL (テーブル作成) 292
DLTMNU (メニュー削除) 276
DLTOVR (オーバーライド削除) 240
DLTPNLGRP (パネル・グループ削
除) 450
DLTSCHIDX (検索索引項目削
除) 452
DSPEDTD (編集記述表示) 39
DSPFD (ファイル記述表示) 23
DSPFFD (ファイル・フィールド記述表
示) 22
DSPMNUA (メニュー属性表示) 276
DSPOVR (オーバーライド表示) 240

コマンド、CL (続き)
DSPPGMREF (プログラム参照表示) 23
ENDRCV (受信終了) 74
GO (メニュー表示へ)
一般メニュー仕様 275
使用 261
OVERLAY (オーバーレイ) キーワード
ヘルプ・リストの更新に対する影響
415
OVRDSPF (ディスプレイ・ファイル・
オーバーライド)
ファイル属性 235
ファイル名 237
ファイル名および属性 238
CHGDSPF との比較 18
RMVSCIDX (検索索引項目除去) 451
SEU 開始 (STRSEU) 22
STRPDM (プログラム開発管理機能開始) 3
STRSCHIDX (検索索引開始) 443
STRSEU (SEU 開始) 22
コマンドおよび機能キーの保持
System/36 アプリケーション互換性
773
コマンド機能 (CFnn) キー
オンライン・ヘルプ情報からプログラム
に戻る 419
システム・パフォーマンス考慮事項
297
妥当性検査の考慮事項 87
定義 28
コマンド機能 (CFnn) キーワード 28
コマンド行
ウィンドウ 394
ウィンドウの例 394
解釈 382
制約事項 382
ツール 490
長すぎる入力 383
任意指定の 487
ポップアップ・ウィンドウ 394
コマンド行 (CMDLINE) ダイアログ・コ
マンド 707
コマンド行 (CMDLINE) タグ
概要 539
コマンド行の解釈 382
コマンド作成 (CRTCMD) コマンド 450
コマンド・アテンション (CAnn) キー
オンライン・ヘルプ情報からプログラ
ムに戻る 419
システム・パフォーマンス考慮事項
297
妥当性検査の考慮事項 87
定義 28

コマンド・アテンション (CAnn) キーワ
ード
キーボード制御用 27
コマンド・キー
参照: コマンド機能 (CFnn) キー
参照: コマンド・アテンション
(CAnn) キー
コマンド・キー保持 (RETCMDKEY) キー
ワード
System/36 アプリケーション互換性
773
コマンド・ストリング取得 (RETRIEVE)
ダイアログ・コマンド 719
コマンド・デフォルト変更
(CHGCMDDFT) コマンド 275
コマンド・プロンプト 717
コロンの(:) 記号 515
混合画面
例 474

[サ行]

最下部指示 (BOTINST) タグ
概要 521
最上部指示 (TOPINST) タグ
概要 687
サイズ
ウィンドウ 130
サインオン
サインオン・メニューの選択 263
索引検索
F18=続きの見出し
アクセス権の除去 424
参照: オンライン・ヘルプ情報
索引検索 (ISCH) タグ
パラメーター・リスト 588
UIM ソース中の配置 444
索引検索階層 446
索引検索キー (F11) 442
索引検索機能
参照: オンライン・ヘルプ情報
索引検索サブトピック (ISCHSUBT) タグ
パラメーター・リスト 590
UIM ソース中の配置 446
索引検索同義語 (ISCHSYN) タグ
パラメーター・リスト 591
UIM ソース中の配置 444
索引検索のルート・ワード
参照: オンライン・ヘルプ情報
削除
ウィンドウ 138
オーバーライド 240
検索索引 452
パネル・グループ 450
メニュー表示 276

作成
ウィンドウ 129
オブジェクト 305
オンライン・ヘルプ情報のあるアプリ
ケーション表示、そのステップ 3
確認リスト・パネル 329
グラフィカルな表示 151
継続入力フィールド 190
作成 183
垂直単一項目選択フィールド 162
垂直複数項目選択フィールド 163
水平選択フィールド 163
スクロール・バー 179
選択リスト 175
データ表示パネル 333
ディスプレイ・ファイルと記述 20
プルダウン・メニュー 167
編集記述 761
メッセージ 247
メニュー・バー 157
メニュー・バー・パネル 351
メニュー・パネル 309
リスト・パネル 316
サブファイル
いっばいの場合の拡張 103
ウィンドウ 132
オプション機能、表 98
カーソル、位置決め
継続または切り捨てキー使用時の
120
最初の 117
複数のレコードが表示される場合の
120
ロール・キー使用時の 117
外観の制御 109
活動状態のレコード、更新 106
機能 98
高水準言語の入出力要求の表
BASIC ステートメント 109
ILE COBOL ステートメント 109
ILE C/C++ 関数 109
ILE RPG 演算命令 109
高水準言語プログラム、使用 104
サブファイル制御、DDS キーワードの
処理順序 103
サブファイル制御レコード様式、目的
97
サブファイル・レコード様式、目的
97
サブファイル・レコード様式とサブフ
ァイル制御レコード様式の順序 98
指定された場所での活動状態のレコー
ドの読み取り 107
指定された場所でのレコードの追加
106
使用法の理解 95

サブファイル (続き)

- 情報の変更 96
- 選択項目のあるサンプル画面 95
- タスクの組み合わせ 96
- 入力専用、妥当性検査なし 96
- 表示専用 95
- 処理のためにディスプレイ上のレコードを挿入する 109
- 次の変更済みレコードの読み取り 107
- データの妥当性検査 115
- 定義 25
- ディスプレイ装置に対して活動状態にする最大数 102
- ディスプレイ・ファイルに指定できる最大数 102
- 入出力操作の要求 106
- パフォーマンスに関する考慮事項 295
- 表示
 - エラー・メッセージ 116
 - およびレコードの処理を同時に行う 109
 - 縦方向 109
 - 表示モードで横方向に 111
 - 横方向 109
 - レコード 108
- 複数のレコード 95
- プログラム論理 124
- ページ・サイズと等しいサイズ 112
- ページ・サイズと等しくないサイズ 114
- 例
 - 可変長レコード 113
 - サブファイルのプログラム論理 124
 - 情報変更用のサンプル画面 96
 - 選択項目のあるサンプル画面 95
 - タスクの組み合わせ用サンプル画面 96
 - 縦方向および横方向、同時に表示 109
 - 入力専用サンプル画面 96
 - 表示専用サンプル画面 95
 - 複数のレコードをロールする際のサンプル画面 121
 - メッセージ・サブファイル 116
 - 2つの画面サイズ用の DDS 40
 - DSPATR(PC) キーワード、使用 118
 - DSPMOD キーワード、使用 111
 - ROLLUP、SFLSIZ、および SFLPAG キーワード 112
 - SFLNXTCHG キーワード、使用 115
 - SFLPAG、SFLSIZ、および ROLLUP キーワード 112

サブファイル (続き)

- 例 (続き)
- SFLSIZ、ROLLUP、および SFLPAG キーワード 112
- *DS3 モード表示 112
- *DS4 モード表示 111
- レコード
 - 活動サブファイル 98
 - 非活動サブファイル 98
- レコードのページごとのカーソルの位置決め 117
- ローリングおよび妥当性検査 103
- ローリング前の妥当性検査 103
- DDS ソースでの記述 97
- SFL キーワード、使用 97
- SFLCTL キーワード、使用 97
- SFLDSP キーワード、使用 97
- SFLPAG キーワード、使用 97
- SFLSCROLL キーワード、使用法 176
- SFLSIZ キーワード、使用 97
- USRDSPMGT キーワードの制限 774
- 参照：ディスプレイ・ファイル
- サブファイル行 (SFLIN) キーワード 98
- サブファイル切り捨て (SFLDROP) キーワード 98
- サブファイル継続 (SFLFOLD) キーワード 98
- サブファイル削除 (SFLDLT) キーワード 98
- サブファイル実行キー (SFLENTER) キーワード 98
- サブファイル終了 (SFLEND) キーワード 98
- サブファイル消去 (SFLCLR) キーワード 98
- サブファイル初期設定 (SFLINZ) キーワード 98
- サブファイル制御 (SFLCTL) キーワード 97
- サブファイル制御レコード様式
 - 書き込み操作 108
 - 書き込み・読み取り操作 109
 - サブファイル・レコード様式の後の配置 98
 - 定義 97
 - 入出力操作の要求 108
 - 必要な DDS キーワード 97
 - 読み取り操作 109
 - SFLCTL キーワード、使用 97
 - SFLDSP キーワード、使用 97
 - SFLPAG キーワード、使用 97
 - SFLSIZ キーワード、使用 97
- サブファイル選択制御 (SFLCHCCTL) キーワード 175

サブファイル単一選択項目選択

- (SFLSNGCHC) キーワード 174
- サブファイルに関する書き込み・読み取り操作
 - 定義 109
 - ILE C/C++ 関数 109
 - ILE RPG 演算命令 109
- サブファイルに関する更新操作
 - 定義 106
 - BASIC ステートメント 109
 - ILE COBOL ステートメント 109
 - ILE C/C++ 関数 109
 - ILE RPG 演算命令 109
- サブファイルに関する次変更読み取り操作
 - BASIC ステートメント 109
 - ILE COBOL ステートメント 109
 - ILE C/C++ 関数 109
 - ILE RPG 演算命令 109
- サブファイルに関する相対取得操作
 - BASIC ステートメント 109
 - ILE COBOL ステートメント 109
 - ILE C/C++ 関数 109
 - ILE RPG 演算命令 109
- サブファイルに関する相対挿入操作
 - 定義 106
 - BASIC ステートメント 109
 - ILE COBOL ステートメント 109
 - ILE C/C++ 関数 109
 - ILE RPG 演算命令 109
- サブファイル表示 (SFLDSP) キーワード 97
- サブファイル表示制御 (SFLDSPCTL) キーワード 98
- サブファイル複数選択項目選択 (SFLMLTCHC) キーワード 174
- サブファイル戻り選択済み項目 (SFLRTNSEL) キーワード 176
- サブファイル・カーソル移動先 (SFLCSRPRG) キーワード 33
- サブファイル・カーソル相対レコード番号 (SFLSCRRRN) キーワード 72
- サブファイル・サイズ (SFLSIZ) キーワード
 - サブファイル制御レコード様式での使用 97
 - ROLLUP および SFLPAG の指定例 112
- サブファイル・スクロール (SFLSCROLL) キーワード 176
- サブファイル・プログラム・メッセージ待ち行列 (SFLPGMQ) キーワード 98
- プログラム・メッセージ待ち行列からメッセージを表示する 250

サブファイル・ページ (SFLPAG) キーワード
 サブファイル制御レコード様式での使用 97
 ROLLUP および SFLSIZ の指定例 112

サブファイル・メッセージ (SFLMSG) キーワード
 からの結果のメッセージ 132
 サブファイル制御レコードの書き込み時にメッセージ行に表示する 248
 説明 98

サブファイル・メッセージ識別コード (SFLMSGID) キーワード 251
 からの結果のメッセージ 132
 サブファイル制御レコードの書き込み時にメッセージ行に表示する 248
 説明 98
 ディスプレイでの定義 250

サブファイル・メッセージ・キー (SFLMSGKEY) キーワード
 説明 98
 プログラム・メッセージ待ち行列からメッセージを表示する 250

サブファイル・メッセージ・レコード (SFLMSGRCRD) キーワード
 説明 98
 プログラム・メッセージ待ち行列からメッセージを表示する 250

サブファイル・モード (SFLMODE) キーワード 72

サブファイル・レコード番号 (SFLRCNBR) キーワード
 説明 98

サブファイル・レコード非活動 (SFLRNA) キーワード 98

サブファイル・レコード様式
 更新操作 106
 サブファイル制御レコード様式の前の配置 98
 次変更読み取り操作 107
 相対取得操作 107
 相対挿入操作 106
 定義 97
 入出力操作の要求 106
 SFL キーワード、使用 97

サブファイル・ロール値 (SFLROLVAL) キーワード 98

参考文献 781
 参照フィールド
 定義 26

サンプル図表
 例 578

サンプル見出し
 例 697

式
 条件 540, 600
 識別コード、レベル 22
 時刻値
 表示画面形式 538
 自己検査、System/36 アプリケーション互換性 765
 指示行
 情報画面 474
 入力画面 465
 メニュー表示、単一選択 463
 リスト画面 476
 システム・コマンド (CMD) ダイアログ・コマンド 706
 システム・パフォーマンス
 参照：パフォーマンス
 システム・メッセージ
 参照：メッセージ
 システム・メニュー
 参照：メニュー表示
 実行
 システム・メニューとユーザー定義メニュー 261
 実行時エラーからの回復 93
 指定
 拡張表示機能 153
 右から左への表示処理 34
 自動確認処理 332
 自動選択
 単一項目選択フィールド 185
 自動入力
 単一項目選択フィールド 185
 次変更読み取り操作
 定義 107
 終了キー、メニューでの使用 262
 終了属性文字
 定義 24
 受信
 データ 48
 受信終了 (ENDRCV) コマンド 74
 出力専用ファイル、定義 43
 出力操作
 アプリケーションへのカーソル位置の返送 72
 カーソルの位置決め 71
 OVERLAY および ERASE の影響 54
 出力データ
 書き込みおよび入力読み取りの同時実行 83
 ディスプレイへの書き込み 49
 出力ファイル
 定義 43
 出力フィールド
 使用 25
 定義 25
 編集 38

出力フィールド (続き)
 例
 要求に対する応答で表示される、入力フィールドのある画面 26
 出力/入力フィールド
 使用 25
 消去 66
 定義 25
 取得属性フィールドバック域 733
 順不同リスト (UL) タグ
 概要 690
 許可されるタグの表 690
 ヘルプ・モジュールでの使用 440
 例、順不同リスト 691
 消去
 画面の行
 CLRL キーワードを使用 57
 S/36 環境の考慮事項 763
 環境の考慮事項 763
 消去、入力 66
 条件演算子 540
 条件式 540, 600
 条件付け
 オプション
 例 543
 条件定義 (COND) タグ
 概要 540
 例、条件付けオプション 543
 CHKOBJ (オブジェクト検査) 組み込み関数 541
 CHKPGM (COND 評価プログラム) 542
 CHKUSRCLS (ユーザー・クラス検査) 組み込み関数 542
 条件名、画面サイズ 39
 小数点、編集記述 759
 情報、位置
 参照：位置情報、表示
 情報、オンライン・ヘルプ
 参照：オンライン・ヘルプ情報
 情報、ヘルプ
 参照：オンライン・ヘルプ情報
 情報域
 印刷 588
 スクロール 379
 情報域 (INFO) タグ
 印刷のフォーマットの考慮事項 588
 概要 587
 情報域のスクロール 379
 情報画面
 位置情報 472
 オンライン・ヘルプ情報
 説明 497
 ヘルプ域 497
 DDS の考慮事項 498
 指示行 474

- 情報画面 (続き)
 - 説明 471
 - 操作ガイドライン 474
 - タイトル 472
 - プロンプト域 473
 - 例 471
 - CUA 505
- 初期化
 - 出力/入力フィールド 72
 - リスト画面 372
- 初期メニュー
 - 定義 263
- 除去
 - ウィンドウ 138, 393
 - プルダウン・メニュー 202
 - ポップアップ・ウィンドウ 393
 - リストからの項目 373
 - API を使用するポップ・ウィンドウ 393
- 書式設定
 - 機能キー 383
 - 数値
 - 値の例の表 538
 - パネル 385
 - 概念 385
 - ユーザー制御 387
 - 表示画面形式 538
 - 時刻値 538
 - 日付値 538
 - 文字値 538
 - 2 進数値 537
 - IGC (漢字)
 - BOTINST (最下部指示) タグ 521
 - DATAIC (データ項目選択) タグ 555
 - TOPINST (最上部指示) タグ 688
 - 処置リスト定義 375
 - 処置リストのオプション、プロンプト 717
 - ジョブ
 - 共用されるファイルのオープン操作 92
 - 共用されるファイルのクローズ操作 93
 - 共用されるファイルの入出力操作 93
 - ジョブ変更 (CHGJOB) コマンド 253
 - シンボル
 - アンパーサンド (&) 515
 - コロン (:) 515
 - ピリオド (.) 515
 - 右スラッシュ (/) 515
 - メッセージ・テキスト
 - メッセージ・ファイル・デフォルト 515
 - 連結 515
 - 真理値表 (TT) タグ
 - 概要 688
 - 許可されるタグの表 689
 - 例、真理値表 689
 - 真理値表行 (TTROW) タグ
 - 概要 689
 - 推奨される規則
 - プルダウン・フィールド選択項目 (PDFLDC) タグ 664
 - リスト・アクション (LISTACT) タグ 619
 - 垂直
 - 単一項目選択フィールド 162
 - 複数項目選択フィールド
 - カーソル移動 164
 - 例 163
 - 水平選択フィールド
 - カーソル移動 165, 166
 - 例 163
 - 数値
 - 表示画面形式
 - 値の例の表 538
 - 2 進数 537
 - 数値入力データの処理、負 82
 - スクロール
 - エラー条件 379
 - 機能キー 378
 - サポート 378
 - 状況情報 378
 - 情報域 379
 - スクロール可能な領域 378
 - データ項目グループ 380
 - 定義 378
 - テキスト域 380
 - 不完全リスト 379
 - メッセージ行 378
 - メニュー域 379
 - リスト域 379
 - 領域の定義 378
 - スクロール可能域の定義 378
 - スクロール・バー
 - サイズ変更 180
 - 作成 179
 - 操作 182
 - 定義 173, 178
 - 例 173
 - 例、DDS 181
 - ストリーム、ユーザー定義のデータ
 - 参照：ユーザー定義のデータ・ストリーム (UDDS)
 - 図表 (FIG) タグ
 - 概要 577
 - ヘルプ・モジュールでの使用 441
 - 例、サンプル図表 578
 - スラッシュ (/) 記号 515
 - 制御
 - 選択項目の選択可能性 184
 - リスト画面のリスト項目 374
 - 制御値 184
 - 制御言語 (CL) プログラム
 - 使用した入出力操作 91
 - 送信勧誘された装置からの読み取り 75
 - 入力の送信勧誘 73
 - 例
 - プログラム・メニュー 271
 - プログラム・メニューに対する RTNPNT(*NO) 273
 - 制御装置、ディスプレイ装置 765
 - 制限
 - パネル・フォーマッター 387
 - 生産性、UIM による向上 301
 - 制約事項
 - コマンド行 382
 - ダイアログ変数の使用 367
 - 名前構文
 - オブジェクト名 515
 - HELP タグ 515
 - 5250 ラベル付きキー 378
 - 選択
 - 選択、共通ユーザー・アクセス 503
 - 単一の項目を選択可能なリスト、リスト画面 486
 - 選択可能な項目情報、入力画面 469
 - 選択項目入力フィールド、共通ユーザー・アクセス 503
 - 選択項目の選択可能性
 - 制御 184
 - 選択項目レベルのヘルプ 203
 - 選択標識
 - 定義 166
 - 抑制
 - 選択フィールド内 166
 - 選択リスト内 176
 - プルダウン・メニュー内 169
 - 選択フィールド
 - 垂直
 - カーソル移動 164
 - 作成 162, 163
 - 水平 163
 - カーソル移動 165
 - 選択標識 166
 - 単一項目
 - 自動選択 185
 - 自動入力 185
 - 垂直 162
 - 定義 160
 - 複数項目
 - 垂直 163
 - 定義 160
 - プルダウン・メニュー 167

選択フィールド (続き)
例 160
例、DDS 161
選択フィールド選択項目
カラーおよび表示属性 188
選択フィールド内で抑制された選択標識
例 166
選択不可能な項目
例 185
選択文字
強調表示 520, 682
国別の 377
選択リスト
作成 175
選択標識 176
単一 175
定義 173
複数 175
例、DDS 174
選択リスト (UIM) 376
選択リストで使用可能になった選択標識
例 177
先頭へ移動 (MOVETOP) ダイアログ・コマンド 713
全般ヘルプ
定義 405, 422
前ページ (PAGEUP) キーワード
機能キー用 27
代替の指定 29
前方フィールド出口処理 196
ソース
確認パネルの例 330
データ表示パネルの例 337
メニューの例 311
メニュー・バー・パネルの例 354
リスト・パネルの例 319
ソース・ファイル
定義 4
ソース・ファイルの埋め込み
ネストされる埋め込み 517
ソース・メンバー
定義 4
相互作用
メニュー・バーとアプリケーションの間 197
メニュー・バーとプルダウン・メニュー間 207
操作ガイドライン
情報画面 474
入力画面 470
メニュー表示、単一選択 464
リスト画面
参照: リスト画面
送信
データ 48

送信勧誘された装置からの読み取り操作
入力の送信勧誘後 73
理解 75
CL プログラム、送信勧誘された装置からの読み取り 75
ILE COBOL ステートメント 91
ILE C/C++ 関数 91
ILE RPG 操作 91
送信勧誘された入力、ディスプレイからの読み取り 74
送信勧誘操作
システム・パフォーマンス考慮事項 297
定義 73
CL コマンド 91
ILE COBOL ステートメント 91
ILE C/C++ 関数 91
ILE RPG 操作 91
送信勧誘取り消し操作
使用のタイミング 83
CL コマンド 91
ILE COBOL ステートメント 91
ILE C/C++ 関数 91
ILE RPG 操作 91
System/36 アプリケーション互換性 772
相対取得操作
定義 107
装置記述の変更 (表示装置)
(CHGDEV DSP) コマンド 41
装置定義リスト
オープン・フィールドバック域の一部 724
定義 47
挿入
リスト画面からの項目 373
HTML タグ 229
双方向の考慮事項
フォーマット 601
属性
オーバーライド 61
継続
例 511
ディスプレイ装置 48
入力フィールド 34
表示 187
フィールド
カラー 37
強調 36
System/36 アプリケーション互換性 766
変更データ・タグ 26
属性一時変更 (OVRATR) キーワード
システム・パフォーマンス考慮事項 296
使用 61

属性取得操作
使用可能な情報の表 48
定義 48
ILE COBOL ステートメント 91
ILE C/C++ 関数 91
ILE RPG 操作 91
属性の相互作用表 617
属性文字 24
属性を開始する文字 24
属性を終了する文字 24

[タ行]

ダイアログ変数
エラー・メッセージ 367
初期値 366
制約事項 367
それを使用する表示パネルのフィールド値 368
定義 365
不正な表示文字 368
変数プール 365
UIM によって定義 693
Z36ENV 693
ZCANCEL 693
ZDBCS 693
ZDSPSIZ 693
ZEXIT 693
ZJOB 693
ZJOBNBR 693
ZLMTCPB 693
ZMENU 693
ZMNULIB 693
ZSYSNAM 693
ZUSER 693
ダイアログ変数および特殊値 369
ダイアログ変数内の不正な文字 368
ダイアログ変数を使用する表示パネルのフィールド値 368
ダイアログ・コマンド
アプリケーションへの制御戻り (RETURN) 719
概要 699
拡張ヘルプ (EXTHELP) 710
画面印刷 (PRINT) 716
機能キー続き表示 (MOREKEYS) 713
コマンド行 (CMDLINE) 707
コマンド・ストリング取得 (RETRIEVE) 719
システム・コマンド (CMD) 706
先頭へ移動 (MOVETOP) 713
その影響 699
入力フィールド、プロンプト 717
入力フィールドのプロンプト 717
ビュー変更 (Change View) 705
表示終了 (EXIT) 709

ダイアログ・コマンド (続き)

プログラム呼び出し (CALL) 703
ヘルプ表示 (DSPHELP) 707
変数更新 (VARUPD) 属性 702
ホーム・メニュー表示 (HOME) 711
メッセージ表示 (MSG) 714
メニュー・バー・カーソル・アクション (ACTIONS) 702
有効な使用法 699
ACTIONS (メニュー・バー・カーソル・アクション) 702
CALL (プログラム呼び出し) 703
CANCEL (取り消し) 705
CHGVIEW (ビュー変更) 705
CMD (システム・コマンド) 706
CMDLINE (コマンド行) 707
DSPHELP (ヘルプ表示) 707
ENTER (Enter) 708
EXIT (表示終了) 709
EXTHelp (拡張ヘルプ) 710
HELP (ヘルプ) 710
HELPHelp (ヘルプ・ヘルプ) 710
HELPIIDX (ヘルプ索引) 711
HOME (ホーム・メニュー表示) 711
KEYSHelp (キー・ヘルプ) 712
MENU (メニュー) 712
MOREKEYS (機能キー続き表示) 713
MOVETOP (先頭へ移動) 713
MSG (メッセージ表示) 714
PAGEDOWN (次ページ) 715
PAGEUP (前ページ) 715
PRINT (画面印刷) 716
PROMPT (プロンプト) 716
PULLDOWN (プルダウン・メニュー表示) 718
RETRIEVE (コマンド・ストリング取得) 719
RETURN (アプリケーションへの制御戻り) 719
VARUPD 属性 702

対称文字

アラビア語 536
ヘブライ語 536

代替キーワード、指定

前ページ (PAGEUP) 29
次ページ (PAGEDOWN) 29
ヘルプ (HELP) 29

代替文字セットおよびコード・ページ

289

タイトル

ウィンドウ 135
情報画面 472
単一選択メニュー表示 463
入力画面 465
リスト画面 475

タイトル行

表 671

タイトルの引用 (CIT) タグ

概要 524
ヘルプ・モジュールでの使用 440
例

タイトルの引用 524

タイプ

ディスプレイ装置 128, 151

タグ

定義 511
MENUGRP (メニュー・グループ) 641

タグ、変更データ 26

タグ、UIM

APPFMT (アプリケーション・フォーマット済み領域) 518
BOTINST (最下部指示) 521
CHECK (妥当性検査) 522
CIT (タイトルの引用) 524
CLASS (クラス定義) 525
CMDLINE (コマンド行) 539
COND (条件定義) 540
COPYR (著作権) 544
DATA (データ表示域) 544
DATAC (データ項目選択) 554
DATACOL (データ列) 555
DATAGRP (データ・グループ) 556
DATAI (データ項目) 559
DATAIX (データ項目エクステンダー) 564
DATASLT (データ選択フィールド) 568
DATASLTC (データ選択フィールド選択項目) 573
DL (定義リスト) 575
FIG (図表) 577
H1 から H4 (見出し) 583
HELP (ヘルプ・モジュール) 579
HP0 から HP9 (強調表示句) 582
IMHELP (組み込みヘルプ) 584
IMPORT (インポート) 586
INFO (情報域) 587
ISCH (索引検索) 588
ISCHSUBT (索引検索サブトピック) 590
ISCHSYN (索引検索同義語) 591
KEYI (キー・リスト項目) 593
KEYL (キー・リスト) 596
LINES (不定形式行) 597
LINK (ハイパーテキスト・リンク定義) 598
LIST (リスト域) 603
LISTACT (リスト・アクション) 614
LISTCOL (リスト列) 621
LISTDEF (リスト定義) 625

タグ、UIM (続き)

LISTGRP (リスト列グループ) 628
LISTVIEW (リスト・ビュー) 631
LP (リストの一部) 633
MBAR (メニュー・バー) 633
MBARC (メニュー・バー選択項目) 636
MENU (メニュー域) 637
MENUI または MI (メニュー項目) 642
NT または NOTE (注記) 644
OL (番号付きリスト) 645
OPTLINE (オプション行) 647
P (段落) 648
PANEL (表示パネル) 649
PARML (パラメーター・リスト) 656
PC (段落の連結) 658
PDACCEL (プルダウン・アクセラレーター) 659
PDFLD (プルダウン・フィールド) 659
PDFLDC (プルダウン・フィールド選択項目) 660
PK (プログラミング・キーワード) 664
PNLGRP (パネル・グループ) 665
PRTHEAD (印刷ヘッド・パネル) 668
PRTPNL (印刷パネル) 673
PRTRAIL (トレーラー・メッセージ印刷) 675
PV (プログラミング変数) 675
RT (テキスト反転) 676
SL (単純リスト) 678
TEXT (テキスト域) 680
TI (変換リスト項目) 685
TL (変換リスト) 686
TOPINST (最上部指示) 687
TT (真理値表) 688
TTROW (真理値表行) 689
UL (順不同リスト) 690
VAR (変数定義) 692
VARRCD (変数レコード定義) 695
XH1 から XH4 (拡張ヘルプの見出し) 696
XMP (例) 697

タグ言語

参照: パネル・グループ定義言語

妥当性検査

キーワード 79
機能の指定 79
サブファイル・データ 115
実行 366
制約事項、CAnn キーの使用時 29
妥当性検査 (CHECK) タグ
概要 522
例、妥当性検査 524

- 妥当性検査キーワード 132
- タブ・キー
 - 移動 208
- 単一項目選択フィールド
 - カラーおよび表示属性 188
 - 垂直 162
 - 水平 163
 - 定義 160
 - プルダウン・メニュー内 167
- 単一項目選択リスト
 - 作成 175
 - 定義 173
 - 例 173
- 単一選択フィールド操作、CUA ガイドライン 503
- 単一選択メニュー表示
 - 参照：メニュー表示
- 短縮定義リスト
 - 例 577
- 単純なメニュー域
 - 例 638
- 単純リスト (SL) タグ
 - 概要 678
 - 許可されるタグの表 678
 - ヘルプ・モジュールでの使用 440
 - 例、単純リスト 679
- 段落 (P) タグ
 - 概要 648
 - ヘルプ・モジュールでの使用 439
 - 例、段落タグ 648
 - UIM ソース中の配置 438
- 段落の連結 (PC) タグ
 - 概要 658
 - ヘルプ・モジュールでの使用 439
 - 例、段落の連結 658
- チェック・ボックス
 - 定義 151
 - 例 151
- 注記 (NOTE) タグ
 - ヘルプ・モジュールでの使用 439
- 注記 (NT または NOTE) タグ
 - 概要 644
- 注記 (NT) タグ
 - 許可されるタグの表 644
 - ヘルプ・モジュールでの使用 439
 - 例、注記の使用 645
- 注記の使用
 - 例 645
- 注釈 517
- 著作権 (COPYR) タグ
 - 概要 544
- ツール、コマンド行 490
- 追加
 - スタックを表示するためのウィンドウ 393
 - 不完全な処理の際の項目 373
- 追加 (続き)
 - 別のパネルへのウィンドウ 393
 - 別のパネルへのポップアップ・ウィンドウ 393
 - API を使用するポップアップ・ウィンドウ 393
- 次に変更するサブファイル (SFLNXTCHG) キーワード
 - 説明 98
 - 例の DDS 115
- 次ページ (PAGEDOWN) キーワード
 - 機能キー用 27
 - 代替の指定 29
- データ 78, 284
 - 受け渡し
 - 同じ経路指定ステップ内のプログラム間 283
 - 経路指定ステップ間での 284
 - プログラム間 283
 - 受け渡しのための ASSUME キーワード考慮事項 283
 - 受け渡しのための KEEP キーワード考慮事項 283
- 外部記述
 - 使用 19
 - 定義 19
- 記述のレベル 18
 - 受信 48
 - 送信および受信 48
 - 妥当性検査 79
- データ
 - 受信 48
 - ディスプレイ上の 2 行間のロール 59
 - ディスプレイ・ファイルとの関係 18
- 入力
 - 負の数値の処理 82
 - 保持 78
- プログラム記述
 - 使用 42
 - 説明 20
 - 定義 19
- プログラムの内部または外部での記述 19
- データ域のヘルプ 428
- データ一時変更 (OVRDTA) キーワード
 - システム・パフォーマンス考慮事項 296
 - 説明 63
- データ記述仕様 (DDS)
 - ウィンドウ 127
 - サブファイル、記述 97
 - サブファイル制御の処理順序 103
 - 使用 17
 - 情報画面 498
 - 妥当性検査キーワード 132
 - 定義 3
- データ記述仕様 (DDS) (続き)
 - ディスプレイ・ファイル
 - 経路指定ステップ間でのデータの受け渡し 284
 - 使用 23
 - プログラム間でのデータの受け渡し 283
 - 入力画面 497
 - 表示モードが変更される場合に無視されるキーワード 70
 - フィールド
 - カラー (表) 37
 - 強調 (表) 36
 - プログラミング例 141
 - プログラム論理 124
 - ヘルプ・レコード 132
 - 変更 21
 - メニュー画面 496
 - リスト画面 499
 - 例
 - 応答標識を使用した HLPCMDKEY キーワード 419
 - 押しボタン 183
 - オプション標識を使用した HLPPNLGRP キーワード 410
 - オプション標識を使用した HLPRCD キーワード 412
 - 可変長レコード 113
 - グリッド・ライン構造 227
 - スクロール・バー 181
 - 選択フィールド 161
 - 選択リスト 174
 - ディスプレイ・ファイル・ソース 23
 - ディスプレイ・ファイル・メニュー 267
 - プログラム・メニュー 271
 - メッセージ・サブファイル 116
 - 略号 186
 - 2 次ヘルプ 417
 - 2 つの画面サイズ 39
 - 2 つの画面サイズ用のサブファイル 40
 - 5250 ディスプレイ装置の問題 87
 - CLRL と OVERLAY キーワードの相違 58
 - DSPATR(PC) キーワード 118
 - DSPMOD キーワード 70
 - DSPMOD キーワードとサブファイル 111
 - ERRSFL キーワード 250
 - GRDATR (グリッド・ライン属性) キーワード 227
 - GRDBOX (グリッド・ボックス) キーワード 227

データ記述仕様 (DDS) (続き)

例 (続き)

GRDCLR (グリッド消去) キーワード 227
 GRDLIN (グリッド・ライン) キーワード 227
 GRDRCD (グリッド・レコード) キーワード 227
 HLPCMDKEY キーワード 419
 HLPNPLGRP キーワード 409
 HLPKCD キーワード 412
 OVERLAY と CLRL キーワードの相違 58
 PULLINPUT キーワード 200
 PUTOVR キーワード 62
 PUTOVR キーワード、繰り返し入力されるデータ 65
 PUTOVR キーワード、より効果的なコーディング 63
 PUTRETAIN キーワード、フィールド・レベル 67
 PUTRETAIN キーワード、レコード・レベル 67
 ROLLUP、SFLSIZ、および SFLPAG キーワード 112
 SFLMSGID キーワード 251
 SFLNXTCHG キーワード 115
 SFLPAG、SFLSIZ、および ROLLUP キーワード 112
 SFLSIZ、ROLLUP、および SFLPAG キーワード 112
 CAnn キー、制約事項 29
 CHECK(FE) (フィールド終了) キーワード 167
 DDS 用紙 23
 DSPRL (右から左へ表示) キーワード 34
 ERRMSG (エラー・メッセージ) キーワード 132
 ERRMSGID (エラー・メッセージ識別コード) キーワード 132
 ERRSFL (エラー・サブファイル) キーワード 132
 MSGLOC (メッセージ位置) キーワード 132
 NOCCSID (コード化文字セット ID 変換なし) キーワード 291
 RMVVDW (ウィンドウの除去) キーワード 138
 SFLMSG (サブファイル・メッセージ) キーワード 132
 SFLMSGID (サブファイル・メッセージ識別コード) キーワード 132
 USRRSTDSP (ユーザー復元画面) キーワード 139

データ記述仕様 (DDS) (続き)

WDWBORDER (ウィンドウ枠) キーワード 133
 WINDOW (ウィンドウ) キーワード 129
 参照: キーワード、DDS
 参照: ディスプレイ・ファイル
 データ項目 (DATAI) タグ
 概要 559
 データ項目エクステンダー (DATAIX) タグ
 概要 564
 データ項目グループのスクロール 380
 データ項目選択 (DATAC) タグ
 概要 554
 データ項目用の 2 つの表示域
 例 551
 データ選択フィールド (DATASLT) タグ
 概要 568
 許可されるタグの表 568
 例
 データ入力パネル 570
 複数選択フィールド 571
 データ選択フィールド選択項目 (DATASLTC) タグ
 概要 573
 データ入力パネル
 概要 350
 ネストされたデータ・グループの例 553
 例 548
 データ入力パネルでの 2 列形式
 例 549
 データの受け渡し
 同じ経路指定ステップ内の 283
 経路指定ステップ間での 284
 データ表示域 (DATA) タグ
 概要 544
 許可されるタグの表 545
 例
 データ項目用の 2 つの表示域 551
 データ入力パネル 548
 データ入力パネルでの 2 列形式 549
 ネストされたデータ・グループを持つデータ入力パネル 553
 メニュー域を持つデータ表示域 552
 データ表示パネルのアプリケーション・プログラミング 347
 データ表示パネルの作成 333
 データ表示パネルのソース 337
 データ保持 PUT (PUTRETAIN) キーワード
 使用 66

データ保持 PUT (PUTRETAIN) キーワード (続き)

例

フィールド・レベル 67
 レコード・レベル 67
 System/36 アプリケーション互換性 764
 USRDSPMGT キーワードでの制限 774
 データ待ち行列
 ディスプレイ・ファイルおよび ICF ファイル 287
 データ列 (DATACOL) タグ
 概要 555
 データ・グループ (DATAGRP) タグ
 概要 556
 データ・ストリーム、ユーザー定義の
 参照: ユーザー定義のデータ・ストリーム (UDDS)
 データ・ストリーム、5250 ディスプレイ
 参照: ユーザー定義のデータ・ストリーム (UDDS)
 データ・タグ、変更 26
 テーブル作成 (CRTTBL) コマンド 292
 定義 267
 アクセラレーター・キー 170
 アプリケーションのポップアップ・ウィンドウ 393
 アプリケーション・ウィンドウ 393
 アプリケーション・ヘルプ 203
 機能キー 383
 機能キーおよび VARUPD 値 384
 スクロール可能域 378
 選択項目のカラーおよび属性 187
 選択項目レベルのヘルプ 203
 ダイアログ変数 365
 取り消しキー 172
 入力フィールドのカーソル移動先 33
 入力フィールドの属性 34
 パネル 513
 パネル・グループ・オブジェクト 315
 フィールド用ヘルプ 206
 プロローグ域 398
 ヘッダー域 399
 メニュー・オブジェクト 308
 メニュー・バー切り替えキー 171
 リスト 372
 略号 186
 MNUBARDSP (メニュー・バー表示) キーワード
 アプリケーション・レコード 197, 209
 メニュー・バー・レコード 199, 212
 UIM での文脈ヘルプ 380

- 定義リスト
 - 装置 724
- 定義リスト (DL) タグ
 - 概要 575
 - 許可されるタグの表 576
 - ヘルプ・モジュールでの使用 440
 - 例
 - 短縮定義リスト 577
 - 定義リスト 577
- ディスプレイ
 - ユーザー・インターフェース管理機能 315
 - リスト 372
 - UIM 315
 - UIM の作成 315
- ディスプレイ装置 46
 - 拡張機能 153
 - 獲得された、入出力操作からの解放 90
 - 情報の入手 47
 - 装置記述、プリンター用の、変更 41
 - タイプ 128, 151
 - フィールドの最大数の計算 81
 - モデル
 - 3180-2 69
 - 3197 69
 - 5250 上のレコード様式の問題の回避 87
 - ユーザー定義のデータ・ストリーム、違い 277
 - UNLOCK を指定 77
- ディスプレイ装置制御装置、System/36 アプリケーション互換性 765
- ディスプレイ入出力フィールドバック領域 731
- ディスプレイ・ファイル
 - ウィンドウの使用法 127
 - オーバーライド
 - HLL プログラムでの属性 235
 - HLL プログラムでのファイル名 237
 - HLL プログラムでのファイル名および属性 238
 - オープン 46
 - オペレーター変更、System/36 アプリケーション互換性 765
 - オンライン・ヘルプ情報
 - 参照： オンライン・ヘルプ情報
 - カーソルの位置決め、System/36 アプリケーション互換性 767
 - 画面サイズ、パフォーマンスの向上のために指定 295
 - 記述
 - 作成 20
 - 変更の適用方法 21
- ディスプレイ・ファイル (続き)
 - 機能キー、System/36 アプリケーション互換性 771
 - クローズ 91
 - コマンドおよび機能キーの保持、System/36 アプリケーション互換性 773
 - コマンド・キー妥当性検査 87
 - 作成
 - 方法 20
 - DDS を使用しない 42
 - DDS を持つ 20
 - サポートされない System/36 の機能 774
 - 自己検査、System/36 アプリケーション互換性 765
 - システムおよび高水準言語でサポートされる操作
 - CL コマンド 91
 - ILE COBOL ステートメント 91
 - ILE C/C++ 関数 91
 - ILE RPG 操作 91
 - 出力専用、定義 43
 - 使用 45
 - 情報の入手 47
 - 設定 17
 - 送信勧誘取り消し操作、System/36 アプリケーション互換性 772
 - 中断状態 85
 - データとの関係 18
 - データの受け渡し
 - 経路指定ステップ間での 284
 - プログラム間 283
 - 定義 3
 - ディスプレイ装置制御装置、System/36 アプリケーション互換性 765
 - 同一ジョブ内での共用 92
 - 特性 17
 - 入出力の定義 43
 - 入力専用、定義 43
 - 入力データ
 - System/36 アプリケーション互換性 765
 - 入力データ、System/36 アプリケーション互換性 764
 - 入力フィールド消去機能、System/36 アプリケーション互換性 766
 - 表示の定義 17
 - ファイル入力フィールド属性 287
 - フィールドの属性、System/36 アプリケーション互換性 766
 - 符号付き数字データ、System/36 アプリケーション互換性 770
 - 編集記述 757
 - 編集コード、ユーザー定義 759
- ディスプレイ・ファイル (続き)
 - 明示的な一時変更を伴う
 - PUT、System/36 互換性 769
 - メッセージの表示、System/36 アプリケーション互換性 767
 - 戻りコード 739
 - 例
 - オンライン・ヘルプ付きの作成 3
 - ディスプレイ上のレコード様式の置換 52
 - プログラム記述へのスペースの割り当て 42
 - DDS ソース 23
 - 例、プログラミング 141
 - レコード様式の最大数 30
 - ロック 45
 - ASSUME (想定)
 - 考慮事項 283
 - HELP および HLPRTN キーワード、System/36 アプリケーション互換性 771
 - ICF ファイル、データ待ち行列でデータ待ち 287
 - System/36 アプリケーション互換性 763
 - 参照： ウィンドウ
 - 参照： サブファイル
 - 参照： データ記述仕様 (DDS)
- ディスプレイ・ファイル作成 (CRTDSPF) コマンド
 - 画面の作成 7
 - ディスプレイ・ファイルの作成 20
 - ディスプレイ・ファイル・メニュー 268
 - プログラム・メニュー 271
 - DFRWRT (書き込み据え置き) パラメーター
 - 書き込み操作 68
 - それによるパフォーマンスの向上 295
 - ENHDSP (拡張表示) パラメーター 154
 - SHARE パラメーター 296
- ディスプレイ・ファイル属性
 - 参照： ファイル属性
- ディスプレイ・ファイル変更 (CHGDSPF) コマンド
 - 実行 21
 - ディスプレイ・ファイル変更 (CHGDSPF)
 - ファイル記述の変更 18
 - 変更
 - ディスプレイ・ファイル 21
 - ファイル記述 18

ディスプレイ・ファイル変更 (CHGDSPF)
コマンド (続き)
DFRWRT (書き込み据え置き) パラメ
ーター
説明 295
ディスプレイ・ファイルの作成 68
ENHDSP (拡張表示) パラメーター
154
SHARE パラメーター 296
ディスプレイ・ファイル・オーバーライド
(OVRDSPF) コマンド
ファイル属性 235
ファイル名 237
ファイル名および属性 238
CHGDSPF との比較 18
ディスプレイ・ファイル・メニュー
参照: メニュー表示
テキスト (TEXT 域) タグ
概要 680
テキスト域
スクロール 380
テキスト域 (TEXT) タグ
テキスト・データ 682
例
テキスト域 684
テキスト域、記述
入力画面 469
テキスト域のスクロール 380
テキスト反転 (RT) タグ
概要 676
例
左から右方向のパネル上の、左から
右方向のフォーマット設定 678
右から左方向のパネル上の、左から
右方向のフォーマット設定 677
テキスト・データ
選択文字 682
出口プログラム
よく尋ねられる質問 400
デフォルト (DFT) キーワード
使用 69
ASSUME キーワードと併用 283
デフォルト値
フィールド 69
同義語、索引検索
参照: オンライン・ヘルプ情報
動的リスト列見出しのフォーマット
例 612
特殊値およびダイアログ変数 369
トピック、索引検索
参照: オンライン・ヘルプ情報
取り消し、読み取り操作、CL コマンド
91
取り消しキー
定義 172
メニュー・パネル 262

トレーラーの印刷 398
トレーラー・メッセージ
例 675
トレーラー・メッセージ印刷 (PRTRAIL)
タグ
概要 675
例、トレーラー・メッセージ 675

[ナ行]

内容区切り記号 511
長すぎるコマンド 383
名前構文
英字変換 515
オブジェクト名
制約事項 515
HELP タグ上の
制約事項 515
二重印字 398
入出力操作
獲得されたディスプレイ装置の解放
90
キーボードのロックおよびカーソルの
位置決め 83
情報の入手 47
ジョブ内で共用されるファイル 93
ディスプレイ装置の獲得 46
入出力操作へのマッピング 91
要求
サブファイル制御レコード様式の
108
サブファイルの 106
サブファイル・レコード様式の
106
HLL 操作へのマッピング
CL コマンド 91
ILE COBOL ステートメント 91
ILE C/C++ 関数 91
ILE RPG 操作 91
入出力フィールドバック域
共通 727
定義 28, 47
ディスプレイ 731
内容 47
入出力要求
高水準言語、サブファイルの表 109
高水準言語でサポートされる 91
入力
長すぎるコマンド 383
入力可能位置にのみカーソル移動
(CSRINPONLY) キーワード 31
入力可能フィールド
数の制限 81
入力画面
オンライン・ヘルプ情報
説明 496

入力画面 (続き)
オンライン・ヘルプ情報 (続き)
タイプ 496
ヘルプ域 497
DDS の考慮事項 497
記述テキスト域 469
指示行 465
説明 465
選択可能な項目情報 469
操作ガイドライン 470
タイトル 465
入力フィールド 467, 468
フィールド・プロンプト 466
プロンプト域 466
例
CUA 入力フィールド 504
i5/OS スタイル 465
CUA 入力フィールド 504
入力消去 (ERASEINP) キーワード
システム・パフォーマンス考慮事項
297
使用 66
入力専用ファイル
定義 43
入力操作
INZRCD および UNLOCK キーワード
の影響 79
入力データ
出力書き込み中の読み取り 83
送信勧誘
ディスプレイに対する 73
CL プログラム 73
ディスプレイから
システムが読み取る方法 82
読み取り 76
読み取り、送信勧誘 74
負の数値の処理 82
保持 78
System/36 アプリケーション互換性
765
入力データの受け取り
プルダウン・メニューから 200
PULLINPUT の使用 200
入力の取り消し、待機されなかった 83
入力ファイル
定義 43
入力フィールド
画面の規則 468
使用 25
消去 66
属性 34
そのカーソル移動先 33
定義 25
入力画面 467
プロンプト 717
編集マスクを使用した保護 34

入力フィールド (続き)
メニュー選択 464
例、出力フィールドのある画面 26
レコードの送信時のパフォーマンスの
向上 296
CUA 504
入力フィールド消去機能、System/36 アプ
リケーション互換性 766
入力フィールド属性
作成 34
ネスト
埋め込み
ソース・ファイル 517
ノード (ハイパーテキスト)、定義
参照: オンライン・ヘルプ情報、ハイ
パーテキスト

[八行]

バイナリー戻りコード、プログラム・メニ
ュー用の 270
ハイパーテキスト
定義 448
参照: オンライン・ヘルプ情報
ハイパーテキスト参照句
定義 449
参照: オンライン・ヘルプ情報
ハイパーテキスト・マークアップ言語
(HTML) キーワード
使用 229
フィールドのオーバーラップの解決
229
ハイパーテキスト・リンク
例 601
ハイパーテキスト・リンク定義 (LINK)
タグ
概要 598
条件式
CHKOBJ 組み込み関数 600
CHKUSRCLS (ユーザー・クラス検
査) 組み込み関数 601
例、ハイパーテキスト・リンク 601
CHKPGM (条件評価プログラム) 601
UIM ソース中の配置 449
パネル
アウトライン 513
グループ・オブジェクト 513
定義 304, 513
ヘルプ 513
編成 513
領域 513
UIM メニュー・アウトライン 513
パネル機能の管理 377
パネル内容の定義 385
パネルの書式設定
アプリケーション制御 386

パネルの書式設定 (続き)
概念 385
定義 385
パネルの書式設定のアプリケーション制御
386
パネルを表示する際の要求処理プログラム
395
パネル・グループ
検索索引
検索索引からの項目の除去 451
削除 452
作成 451
への項目の追加 451
コマンド・ヘルプ、割り当て 450
索引検索機能 444
機能 442
同義語、トピック用の選択 445
トピック、決定 444
ルート・ワード、同義語として組み
込む 444
ルート・ワード、トピック用の選択
445
ルート・ワードの同義語、指定
444
索引検索機能に関する UIM タグの追
加 442
索引検索のアクセス 423
削除 450
作成 450
使用 437
段落と注記、ヘルプ・モジュール中で
の定義 439
定義 3, 304
トピック、定義 444
ハイパーテキスト、定義 448
プログラミング・キーワードおよびプ
ログラミング変数、識別 441
文書とレコードの比較 404
ヘルプ、指定 407
ヘルプ・モジュール
テキストの強調とフォーマット設定
439
テキストの強調表示 440
ハイパーテキスト 448
複数回使用 438
別のヘルプ・パネル・グループに含
まれるものの使用 438
リンク 448
ヘルプ・モジュール中の構造化テキス
ト、指示 441
ヘルプ・モジュール中のリストの作成
440
ヘルプ・モジュールを使用した編成
438
見出し、ヘルプ・モジュールへの追加
440

パネル・グループ (続き)
ユーザー定義のアクセス 423
例
インポートされるヘルプ・モジュー
ル 439
オプション標識を使用した
HLPPNLGRP キーワード 410
組み込みヘルプ・モジュール 438
コマンド・ヘルプ 450
ヘルプ・モジュール 438
HLPPNLGRP キーワード 409
ISCH タグ 444
ISCHSYN タグ 444
LINK タグ 449
UIM ソースの入力 437
参照: タグ、UIM
パネル・グループ (PNLGRP) タグ
概要 665
許可されるタグの表 665
UIM ソース中の配置 438
パネル・グループ削除 (DLTPNLGRP) コ
マンド 450
パネル・グループ作成 (CRTPNLGRP) コ
マンド
画面の作成 8
使用 450
パネル・グループ定義言語
概要 512
各国語の翻訳
文末文字 512
ソース・ファイルの埋め込み 517
マークアップ/内容区切り文字 511
パネル・グループ内のエレメント 305
パネル・グループ・エレメント 305
パネル・グループ・オブジェクト
作成 315
定義 513
パネル・フォーマッターの制限 387
パネル・マークアップ・タグ
定義 439
参照: タグ
パネル・リストをまとめて閉じる 388
パフォーマンス 296
ウィンドウ・アプリケーション 139
オーバーライド・コマンド 296
書き込み操作の据え置き 295
画面の復元 297
共用、同じジョブのオープン・デー
タ・パス 296
コマンド・アテンション・キーとコマ
ンド機能キー 297
サブファイルの使用 295
送信勧誘操作 297
ディスプレイを使用する際の向上方法
295

パフォーマンス (続き)	必須規則	表示終了 (EXIT) ダイアログ・コマンド
入力フィールドを持つレコードの送信	プルダウン・フィールド選択項目	709
296	(PDFLDC) タグ 663	表示装置ファイル
レコードを重複させるが削除しない	必須の規則	参照: ディスプレイ・ファイル
296	リスト・アクション (LISTACT) タグ	表示属性
1 次画面サイズの指定 295	619	カラー・フィールド 37
DDS ウィンドウ 298	ビュー変更 (CHGVIEW) ダイアログ・コ	強調、フィールド 36
ERASEINP キーワード、考慮事項	マンド 705	選択項目 187
297	表示	表示属性 (DSPATR) キーワード
INZINP キーワード、考慮事項 297	オーバーライド 240	使用 71
OVERLAY キーワード、考慮事項	ダイアログ変数を使用する表示パネル	プログラム-システム間フィールドで
297	のフィールド値 368	37
PUTOVR、OVRDTA、および	メッセージ	ASSUME キーワードと併用 283
OVRATR キーワード、その考慮事項	サブファイル制御レコードの書き込	PC 値のソースの例 118
296	み時にメッセージ行に 248	表示属性 (*DSPATR) パラメーター 187
RSTDSP (画面復元)	サブファイルを使用した 250	表示パネル (PANEL) タグ
システム・パフォーマンス考慮事項	について 247	概要 649
297	フィールドに 249	許可されるタグの表 649
必要なキーワード 297	プログラム・メッセージ待ち行列中	例
パラメーター	250	パネル定義 656
画面復元 (RSTDSP)	メッセージ行に 248	表示モード
例 85	メッセージ・フィールドを使用して	サブファイルを横方向に表示する 111
データ待ち行列 (DTAQ) 287	メッセージ行に 249	定義 111
プログラム・メニュー用の引き渡し	要求処理プログラムを使用するパネル	変更 111
270	の表示 395	変更される場合に無視される DDS キ
戻り点 (RTNPNT) パラメーター	リスト 372	ーワード 70
ディスプレイ・ファイル・メニュー	System/36 アプリケーション互換性	表示モード (DSPMOD) キーワード
用 261	767	サブファイルの使用 111
プログラム・メニュー用 261	表示、アプリケーション	サブファイルの例 111
DTAQ (データ待ち行列) 287	参照: アプリケーション表示	説明 70
RSTDSP (画面復元)	表示画面形式	*DS3 のサンプル・サブファイル・デ
例 85	時刻値 538	ィスプレイ 112
RTNPNT (戻り点) パラメーター	数値 537	*DS4 のサンプル・サブファイル・デ
ディスプレイ・ファイル・メニュー	数値の例 538	ィスプレイ 111
用 261	日付値 538	表示文字
プログラム・メニュー用 261, 273	文字値 538	正しくない 368
SHARE (共用) 296	2 進数値 537	開く
パラメーター・リスト (PARML) タグ	標識	UIM アプリケーション 365
概要 656	応答	ピリオド (.) 記号 515
許可されるタグの表 657	定義 27	ヒント
ヘルプ・モジュールでの使用 440	ディスプレイ・ファイル用 30	語の折り返し (WRDWRAP) キーワー
例、パラメーター・リスト 658	オプション	ド 36
番号付きリスト (OL) タグ	定義 30	ファイル
概要 645	ディスプレイ・ファイル用 30	ジョブ内で共用される
許可されるタグの表 646	オフに設定する 31	オープン操作 92
ヘルプ・モジュールでの使用 440	選択	クローズ操作 93
例、番号付きリスト 646	選択フィールド内 166	入出力操作 93
非活動サブファイル・レコード	選択リスト内 176	ソース
定義 98	プルダウン・メニュー内 169	参照: ソース・ファイル
左から右方向のパネル上の、左から右方向	それによる情報の引き渡し 30	定義 19
のフォーマット設定	データ変更時の設定 78	ディスプレイ
例 678	定義 30	参照: ディスプレイ・ファイル
日付値	ディスプレイ・ファイル用のタイプ	ファイル依存フィールドバック域 47
表示画面形式 538	30	ファイル記述
	標識域 (INDARA) キーワード 30	決定 18
		定義 3

ファイル記述 (続き)
 ディスプレイ・ファイル用、作成 20
 変更 21
 その理由 18
 CHGDSPF コマンドによる 21
 変更の検出 22
 ファイル記述表示 (DSPFD) コマンド 23
 ファイル属性
 オーバーライド
 HLL プログラムでディスプレイ・
 ファイル名と共に 238
 HLL プログラムでのディスプレ
 イ・ファイル 235
 構築 235
 ファイルのリダイレクト
 同じファイル・タイプ 240
 組み合わせに対するデフォルトの表
 242
 異なるファイル・タイプ 240
 定義 240
 入出力に有効 241
 レベル検査 240
 ファイル・フィールド記述表示 (DSPFFD)
 コマンド 22
 ファイル・リソース
 割り振りの方法 45
 ファイル・レベル記述 19
 フィールドバック域
 エラー情報 47
 オープン
 個別の説明 721
 装置定義リスト 724
 定義 47
 機能キー 28
 共通 47
 取得属性 733
 入出力
 機能キー用 28
 共通 727
 定義 47
 ディスプレイ 731
 ファイル依存 47
 フィールド
 以前に記述した 26
 カーソル移動先
 定義 33
 隠し 25
 カラー 37
 強調 36
 固定 25
 最大数の計算、ディスプレイ装置 81
 参照 26
 出力 25
 出力/入力 25
 消去、無保護の入力および出力/入力
 66

フィールド (続き)
 数値
 最大長 26
 選択項目入力、共通ユーザー・アクセ
 ス 503
 ゾーン 10 進数、最大長 26
 属性
 定義 34
 属性、System/36 アプリケーション互
 換性 766
 属性または内容のオーバーライド 61
 妥当性検査エラー 77
 定義 18
 カーソル移動先 33
 属性 34
 ディスプレイ上での保持 66
 デフォルト値の指定 69
 特性 18
 入出力、要求に対する応答で表示され
 る 26
 入力 25
 参照：入力フィールド
 入力可能、数の制限 81
 表示 23, 24
 プログラム-システム間
 用法について 25
 DSPATR キーワードで 37
 メッセージ 25
 メニュー選択項目 464
 文字、最大長 26
 リスト
 参照：リスト・フィールド
 レコード様式 25
 レコード様式における最大数 26
 フィールド (*FLD) 特殊値 204
 フィールド、図形 517
 フィールド位置
 指定のタイミング 26
 入出力レコード 25
 フィールド終了 (CHECK(FE)) キーワード
 167
 フィールド属性文字 24
 フィールド出口処理
 後方 197
 前方 196
 フィールド戻し (RTNFLD) パラメーター
 説明 202
 例 202
 フィールド用ヘルプ 206
 フィールド・プロンプト、入力画面 466
 フィールド・レベル記述
 使用 18
 定義 18
 フォーマット
 印刷
 情報域 588

フォーマット (続き)
 印刷 (続き)
 リスト域 608
 双方向の考慮事項 601
 列データ 625
 IGC (漢字)
 PDFLDC (プルダウン・フィールド
 選択項目) タグ 663
 フォーマット設定
 例 698
 フォント
 強調表示 398
 複数項目選択フィールド
 カラーおよび表示属性 188
 垂直 163
 定義 160
 複数項目選択リスト
 作成 175
 定義 173
 例 173
 複数選択フィールド (UIM)
 例 571
 複製オブジェクト作成 (CRTDUPOBJ) コ
 マンド
 使用 275
 符号付き数字データ、System/36 アプリケ
 ーション互換性 770
 不正な表示文字 368
 不正なプリンター文字 368
 不定形式行 (LINES) タグ
 概要 597
 ヘルプ・モジュールでの使用 441
 例、不定形式行 597
 負の数値入力データの処理 82
 ブランク行による分離 398
 プリンター文字、不正な 368
 プリンター・ファイルのオーバーライドお
 よび共用 399
 プリンター・ファイルの共用およびオーバ
 ーライド 399
 フルスクリーン・モード
 ヘルプ情報 581
 プルダウン・アクセラレーター
 (PDACCEL) タグ
 概要 659
 プルダウン・フィールド (PDFLD) タグ
 概要 659
 許可されるタグの表 660
 プルダウン・フィールド選択項目
 (PDFLDC) タグ
 概要 660
 推奨される規則 664
 必須規則 663
 プルダウン・メニュー
 カーソルの制限 172
 作成 167

- プルダウン・メニュー (続き)
 - 単一項目選択フィールド 167
 - 定義 157
 - 入力データの受け取り 200
 - 入力データを受け取った後の除去 202
 - 表示前の更新 202
 - メニュー・バーとの相互作用 207
 - 例 167, 209
 - CHECK(FE) キーワード 167
- プルダウン・メニューとポップアップ・ウィンドウとの違い 391
- プルダウン・メニュー内で抑制された選択標識
 - 例 169
- プルダウン・メニューの更新 202
- プログラマブル・マウス・ボタン (MOUBTN) キーワード
 - イベント処理の状態 219
 - イベント処理の優先順位
 - シフト左ボタン押し 223
 - シフト左ボタン・リリース 224
 - シフト右ボタン押し 225
 - 他のポインター装置 226
 - 非シフト左ボタン押し 220
 - 非シフト左ボタン・ダブルクリック 223
 - 非シフト左ボタン・リリース 222
 - 概要 216
 - 操作 218
 - ポインター装置イベント 216
 - 戻される AID コード 217
 - 利点 217
 - NWS の考慮事項 219
- プログラミング変数 (PV) タグ
 - 概要 675
 - ヘルプ・モジュールでの使用 441
- プログラミング例
 - ウィンドウの使用法 141
- プログラミング・キーワード (PK) タグ
 - 概要 664
 - ヘルプ・モジュールでの使用 441
- プログラム、CL
 - 参照: CL プログラム
- プログラム、IBM 提供の
 - QRCVDTAQ (データ待ち行列から受信) 287
 - QSNDDTAQ (データ待ち行列へ送信) 287
- プログラム開発管理機能 (PDM)
 - 定義 4
- プログラム開発管理機能開始 (STRPDM)
 - コマンド 3
- プログラム記述データ
 - 記述 20
 - 使用 42
 - 定義 19
- プログラム記述データ (続き)
 - 表示スペースの割り当ての例 42
 - プログラムとシステム視点 20
- プログラム参照表示 (DSPPGMREF) コマンド 23
- プログラム呼び出し (CALL) ダイアログ・コマンド 703
- プログラム-システム間フィールド
 - 使用法について 25
 - DSPATR キーワードで 37
- プログラム・メニュー
 - 定義 261
 - 戻りコード値、表 270
 - 参照: メニュー表示
- プロローグ域
 - 定義 398
- プロローグ・セクション
 - パネル・グループ 514
- プロンプト
 - コマンド 717
 - 処置リストのオプション 717
 - 入力フィールド 717
- プロンプト、フィールド
 - 参照: フィールド・プロンプト
- プロンプト域
 - 入力画面 466
 - リスト画面 476
- 文書、オフィス
 - ヘルプの定義 8
- 分析
 - システム・エラー・メッセージ 254
- 文末文字 512
- 文脈ヘルプ
 - 画面上の定義 405
 - 定義 422
 - UIM での定義 380
- 分離のためのブランク行 398
- ページ、コード
 - 参照: コード・ページ
- ページング
 - 位置情報、リスト画面 478
 - ヘルプ画面、DDS レコード 415
- ヘッダー域
 - 定義 399
- ヘルプ
 - アプリケーションのための定義 203
 - ウィンドウに表示される 132, 134
 - 機能キー域 433
 - 項目固有 205
 - 選択項目レベル 203
 - データ域 428
 - パネル 514
 - フィールド 206
 - メニュー域 426
 - メニュー・バー域 431
 - リスト域 422
- ヘルプ域
 - 参照: オンライン・ヘルプ情報
- ヘルプ域 (HLPARA) キーワード 406
- ヘルプからの戻り (HLPRTN) キーワード
 - ヘルプの後にプログラムに戻る 420
- System/36 アプリケーション互換性 771
- ヘルプ境界
 - 定義 416
- ヘルプ境界 (HLPBDY) キーワード
 - 使用 416
 - 2 次ヘルプ 417
- ヘルプ情報
 - 参照: オンライン・ヘルプ情報
- ヘルプの検索索引 (HLPSEARCH) キーワード 407
- ヘルプのサブリスト
 - 定義 416
- ヘルプの順序付け (HLPSEQ) キーワード 416
- ヘルプの除外 (HLPXCLD) キーワード 407
- ヘルプの全画面表示 (HLPFULL) キーワード 407
- ヘルプのタイトル (HLPRTITLE) キーワード 407
- ヘルプ表示
 - 参照: オンライン・ヘルプ情報
- ヘルプ表示 (DSPHELP) ダイアログ・コマンド 707
- ヘルプ・キー 421
 - 押さずにプログラムに戻る 420
 - 押した後にプログラムに戻す 419
 - 押した際の結果 403
 - 使用可能 404
 - 参照: ALTHELP (代替ヘルプ) キーワード
 - 参照: HELP (ヘルプ) キーワード
- ヘルプ・グループ
 - 定義 416
- ヘルプ・コマンド・キー (HLPCMDKEY) キーワード
 - 使用 419
 - USRDSMGT キーワードでの制限 774
- ヘルプ・テキスト
 - 参照: オンライン・ヘルプ情報
- ヘルプ・パネル・グループ (HLPNLRGRP) キーワード
 - オプション標識の使用 410
 - 使用 407
 - HLPRTD および HLPDOC キーワードの使用 410
- ヘルプ・モジュール
 - 構造化テキスト、指示 441
 - 段落と注記の定義 439

ヘルプ・モジュール (続き)

- 定義 438
- テキストの強調表示 440
- 中のテキストの強調とフォーマット設定 439
- 名前 438
- ハイパーテキスト 448
- 複数回使用 438
- プログラミング・キーワードおよびプログラミング変数、識別 441
- 別のヘルプ・パネル・グループに含まれる、使用 438
- 見出しの追加 440
- リストの作成 440
- リンク 448
- 例
 - インポートされる 439
 - 組み込み 438
 - コマンド・ヘルプ 450
 - を使用したパネル・グループ 438
 - ISCH タグ 444
 - ISCHSYN タグ 444
 - LINK タグ 449
 - を使用したパネル・グループの編成 438
- ヘルプ・モジュール (HELP) タグ
 - 概要 579
 - 例、ヘルプ・パネル定義 581
 - UIM ソース中の配置 438
- ヘルプ・モジュール中のタグ
 - 拡張ヘルプの見出し (XH1-XH4)
 - ヘルプ・モジュールでの使用 440
 - 強調表示句 (HP0-HP9)
 - ヘルプ・モジュールでの使用 440
 - 組み込みヘルプ (IMHELP)
 - パネル・グループ・ソース中に配置 438
 - 索引検索 (ISCH)
 - UIM ソース中の配置 444
 - 索引検索同義語 (ISCHSYN)
 - UIM ソース中の配置 444
 - 順不同リスト (UL)
 - ヘルプ・モジュールでの使用 440
 - 図表 (FIG)
 - ヘルプ・モジュールでの使用 441
 - タイトルの引用 (CIT)
 - ヘルプ・モジュールでの使用 440
 - 単純リスト (SL)
 - ヘルプ・モジュールでの使用 440
 - 段落 (P)
 - ヘルプ・モジュールでの使用 439
 - 段落の連結 (PC)
 - ヘルプ・モジュールでの使用 439
 - 注記 (NOTE)
 - ヘルプ・モジュールでの使用 439

ヘルプ・モジュール中のタグ (続き)

- 注記 (NT)
 - ヘルプ・モジュールでの使用 439
- 定義リスト (DL)
 - ヘルプ・モジュールでの使用 440
- ハイパーテキスト・リンク (LINK)
 - UIM ソース中の配置 449
- パネル・グループ (PNLGRP)
 - UIM ソース中の配置 438
- パネル・マークアップ
 - 定義 439
- パラメーター・リスト (PARML)
 - ヘルプ・モジュールでの使用 440
- 番号付きリスト (OL)
 - ヘルプ・モジュールでの使用 440
- 不定形式行 (LINES)
 - ヘルプ・モジュールでの使用 441
- プログラミング変数 (PV)
 - ヘルプ・モジュールでの使用 441
- プログラミング・キーワード (PK)
 - ヘルプ・モジュールでの使用 441
- ヘルプ・モジュール (HELP)
 - ヘルプ・モジュールを使用したパネル・グループ 438
 - UIM ソース中の配置 438
- 見出し (H1 から H4)
 - ヘルプ・モジュールでの使用 440
- 目的 437
- リストの一部 (LP)
 - ヘルプ・モジュールでの使用 440
- 例 (XMP)
 - ヘルプ・モジュールでの使用 441
- CIT (タイトルの引用)
 - ヘルプ・モジュールでの使用 440
- DL (定義リスト)
 - ヘルプ・モジュールでの使用 440
- FIG (図表)
 - ヘルプ・モジュールでの使用 441
- H1 から H4 (見出し)
 - ヘルプ・モジュールでの使用 440
- HELP (ヘルプ・モジュール)
 - ヘルプ・モジュールを使用したパネル・グループ 438
 - UIM ソース中の配置 438
- HP0-HP9 (強調表示句)
 - ヘルプ・モジュールでの使用 440
- IMHELP (組み込みヘルプ)
 - パネル・グループ・ソース中に配置 438
 - を使用したパネル・グループ 438
- IMPORT (インポート)
 - を使用したパネル・グループ 439
 - UIM ソース中の配置 438
- ISCH (索引検索)
 - を使用したパネル・グループ 444
 - UIM ソース中の配置 444

ヘルプ・モジュール中のタグ (続き)

- ISCHSYN (索引検索同義語)
 - を使用したパネル・グループ 444
 - UIM ソース中の配置 444
- LINES (不定形式行)
 - ヘルプ・モジュールでの使用 441
- LINK (ハイパーテキスト・リンク)
 - を使用したパネル・グループ 449
 - UIM ソース中の配置 449
- LP (リストの一部)
 - ヘルプ・モジュールでの使用 440
- NOTE (注記)
 - ヘルプ・モジュールでの使用 439
- NT (注記) 439
- OL (番号付きリスト)
 - ヘルプ・モジュールでの使用 440
- P (段落)
 - ヘルプ・モジュールでの使用 439
 - UIM ソース中の配置 438
- PARML (パラメーター・リスト)
 - ヘルプ・モジュールでの使用 440
- PC (段落の連結)
 - ヘルプ・モジュールでの使用 439
- PK (プログラミング・キーワード)
 - ヘルプ・モジュールでの使用 441
- PNLGRP (パネル・グループ)
 - ヘルプ・モジュールを使用したパネル・グループ 438
 - UIM ソース中の配置 438
- PV (プログラミング変数)
 - ヘルプ・モジュールでの使用 441
- SL (単純リスト)
 - ヘルプ・モジュールでの使用 440
- UL (順不同リスト)
 - ヘルプ・モジュールでの使用 440
- XH1 から XH4 (拡張ヘルプの見出し)
 - ヘルプ・モジュールでの使用 440
- XMP (例)
 - ヘルプ・モジュールでの使用 441
- ヘルプ・リスト
 - アプリケーションのための定義 205
 - 定義 415
- ヘルプ・レコード (HLPRCD) キーワード
 - 同じファイルでの HLPDOC キーワードの使用 413
 - 同じファイルでの HLPNLGRP キーワードの使用 411
 - オプション標識の使用 412
- 変換
 - 英字 515
 - 文字セットおよびコード・ページ、DDS 289
 - 文字セットおよびコード・ページ、UIM 532
- 変換テーブル 291

変換リスト (TL) タグ
 概要 686
 許可されるタグの表 686
 リスト、変換 687

変換リスト項目 (TI) タグ
 概要 685

変更
 ウィンドウ 136
 ディスプレイ・ファイル 21
 ファイル記述 18, 21

変更データ・タグ (MDT)
 定義 26
 文字変換 290
 リセット 66
 ASSUME キーワードと併用 283

変更データ・タグ・オフ (MDTOFF) キーワード 66

編集記述
 規則 760
 作成 761
 について 761
 例 761
 小数点 759
 説明 759
 IBM 提供 759

編集記述作成 (CRTEDTD) コマンド
 実行 39
 使用 761
 例 761

編集記述表示 (DSPEDTD) コマンド 39

編集語
 定義 38

編集語 (EDTWRD) キーワード 38

編集コード
 使用 38
 説明 759
 定義 38
 独自の定義 38
 IBM 提供 757

編集コード (EDTCDE) キーワード 38

編集マスク
 作成 34

変数値 368

変数更新 (VARUPD) 属性
 機能キー 384
 ダイアログ・コマンド 702
 KEYI タグ 594
 PDFLDC タグ 662

変数定義 (VAR) タグ
 概要 692

変数プール
 サービス 368
 定義 365

変数レコード定義 (VARRCD) タグ
 概要 695

編成
 パネル・アウトライン 514
 UIM メニュー・アウトライン 514

ホーム・メニュー表示 (HOME) ダイアログ・コマンド 711

ポインター装置イベント
 処理の状態 219
 処理の優先順位
 シフト左ボタン押し 223
 シフト左ボタン・リリース 224
 シフト右ボタン押し 225
 他のポインター装置 226
 非シフト左ボタン押し 220
 非シフト左ボタン・ダブルクリック 223
 非シフト左ボタン・リリース 222
 プログラマブル・マウス・ボタン 216

保護、入力フィールドの 34

ポップアップ・ウィンドウ
 除去 393
 スタックを表示するための追加 393
 追加 393
 別のパネルへの追加 389
 ヘルプ情報 580

ポップアップ・ウィンドウとプルダウン・メニューとの違い 391

[マ行]

マークアップ区切り記号 511

マウス・ボタン、プログラマブル
 参照：プログラマブル・マウス・ボタン (MOUBTN) キーワード

待ち行列、データ
 参照：DTAQ (データ待ち行列) パラメーター

まとめて閉じる
 リスト・パネル 388
 EXIT の要求時 387
 EXIT の要求時に複数のパネルを 387

右から左へのカーソル移動 31

右から左への表示処理 34

右から左へ表示 (DSPRL) キーワード 34

右から左方向のパネル上の、左から右方向のフォーマット設定
 例 677

右スラッシュ (/) 記号 515

見出し (H1 から H4) タグ
 概要 583
 ヘルプ・モジュールでの使用 440
 例
 見出しタグ 584

見出し、列
 参照：列見出し

明示的な一時変更を伴う PUT (PUTOVR) キーワード
 繰り返し入力されるデータ 65
 サンプル DDS ソース 62
 より効果的なコーディング 63

メジャー/マイナー戻りコード
 エラーからの回復
 永続的なシステム・エラーまたはファイル・エラー 258
 オープン操作または獲得操作時の装置エラーまたはセッション・エラー 259
 正常完了 257
 入出力操作時の永続的な装置エラーまたはセッション・エラー 258
 入出力操作時の回復可能な装置エラーまたはセッション・エラー 259
 例外での完了 257

説明 256

メジャー戻りコードの定義
 00 256
 02 256
 03 256
 04 256
 08 256
 11 256
 34 256
 80 256
 81 256
 82 256
 83 256

メッセージ
 ウィンドウ 132
 エラー
 サブファイルからの表示 116
 エラーの自動処理 253
 ガイドライン 500
 行の関係、CUA 502
 作成および表示 247

システム
 永続 254
 エスケープ 254
 警報 254
 検出 254
 状況 254
 診断 254
 通知 254
 ファイル状況コード 254
 分析 254
 報告 254
 メッセージ待ち行列 254

説明 255, 499

ディスプレイ・ファイルに関する使用可能な機能 247
 に関するアラームの鳴動 253

- メッセージ (続き)
 - の理由 247
 - 表示
 - サブファイル制御レコードの書き込み時にメッセージ行に 248
 - サブファイルを使用した 250
 - の優先順位 249
 - フィールドに 249
 - プログラム・メッセージ待ち行列中 250
 - メッセージ行に 248
 - 用のメッセージ行 249
 - 例
 - ERRSFL キーワード 250
 - SFLMSGID キーワード 251
 - CPF4001-40FF 255
 - CPF4101-43FF 255
 - CPF4401-44FF 255
 - CPF4501-46FF 255
 - CPF4701-48FF 255
 - CPF4901-49FF 255
 - CPF5001-50FF 255
 - CPF5101-53FF 255
 - CPF5501-56FF 255
 - System/36 アプリケーション互換性 767
- メッセージ、モニター
 - 参照: メッセージ
- メッセージ位置 (MSGLOC) キーワード 132
- メッセージ記述 31
- メッセージ記述追加 (ADDMSGD) コマンド 269
- メッセージ行
 - 使用 25
 - スクロール 378
 - 定義 25
- メッセージ行に表示されるメッセージの優先順位 249
- メッセージ行にメッセージを表示するためのメッセージ・フィールド 249
- メッセージ行にメッセージを表示する場合の優先順位 249
- メッセージのモニター
 - 参照: メッセージ、モニター
- メッセージ表示 (MSG) ダイアログ・コマンド 714
- メッセージ・アラーム (MSGALARM) キーワード 253
- メッセージ・サブファイル 116
- メッセージ・テキスト記号 515
- メッセージ・ファイル
 - 作成 269
 - メッセージの追加 269
 - メニュー・アクションの記述 265
- メッセージ・ファイル作成 (CRTMSGF)
 - コマンド 269
- メニュー域
 - グループを持つ
 - 例 639
 - スクロール 379
 - ヘルプ 426
- メニュー域 (MENU) タグ
 - 許可されるタグの表 637
 - 例
 - グループを持つメニュー域 639
 - 単純なメニュー域 638
- メニュー画面 490
 - オプション 463
 - オンライン・ヘルプ情報
 - 説明 495
 - タイプ 495
 - ヘルプ域 495
 - DDS の考慮事項 496
 - 機能キー域 462
 - 指示行 463
 - 説明 462
 - 選択項目フィールド 464
 - 操作ガイドライン 464
 - 任意指定の 487
 - メニュー選択項目フィールド 464
 - メニュー表示、単一選択 463, 464
 - ユーザー・インターフェース管理機能 (UIM) 308
 - 例
 - 単一選択のサンプル 462
- CUA 502
- ID フィールド 487
- UIM の作成 308
- メニュー画面の ID フィールド 487
- メニュー項目 (MENU または MI) タグ
 - 概要 642
- メニュー削除 (DLTMNU) コマンド 276
- メニュー作成 (CRTMNU) コマンド
 - ディスプレイ・ファイル・メニュー 269
 - プログラム・メニュー 272
- メニュー属性表示 (DSPMNUA) コマンド 276
- メニュー定義 305
- メニューに戻る、GO コマンドの実行後 261
- メニュー表示
 - サインオン時に表示されるものを選択 263
 - 削除 276
 - システム、実行 261
 - 終了キー、使用 262
 - 初期 263
 - 属性の表示 276
 - 属性の変更 276
- メニュー表示 (続き)
 - 直前のメニューの判別 261
 - ディスプレイ・ファイル
 - 作成 267, 268
 - その一部 263
 - そのヘルプ 263
 - 定義 261, 263
 - ヘルプ・フォーマットの命名 265
 - メッセージ・ファイル、メッセージの追加 269
 - メッセージ・ファイル、メニュー・アクションの記述 265
 - メッセージ・ファイルの作成 269
 - メニューおよびヘルプ情報、記述 267
 - メニュー・オブジェクトの作成 269
 - DDS の考慮事項 264
 - DDS、指定 267
 - 取り消しキー、使用 262
 - プログラム
 - 記述 271
 - 作成 271
 - 制御言語プログラムの作成 272
 - 定義 261, 270
 - ディスプレイ・ファイルの作成 271
 - パラメーターの引き渡し 270
 - 前のメニューに戻らずに出る 273
 - メニューの作成 272
 - CL ソースの入力 271
 - 命名の競合、回避
 - 一般メニュー仕様 275
 - コマンド複製後のコマンド・デフォルトの変更 275
 - 命名、方法の決定 275
 - ライブラリーの指定 275
 - ライブラリー・リスト内に配置 275
 - 命名の競合、その回避
 - 方法 275
 - メニュー・スタック 261
 - ユーザー定義の
 - 実行 261
 - タイプ 261
 - 例
 - オプションおよび GO コマンドの使用 262
 - ディスプレイ・ファイル・メニュー 267
 - プログラム・メニュー 271
 - プログラム・メニュー、CL ソース 271
 - プログラム・メニューに対する RTNPNT(*NO)、CL ソース 273

メニュー表示 (続き)
 例 (続き)
 メッセージ・ファイル、メッセージの追加 269
 メッセージ・ファイル、メニュー・アクションの記述 265
 メッセージ・ファイル内でのヘルプ・フォーマットの命名 265
 GO コマンドを実行後、戻る 261
 参照：アプリケーション表示
 メニュー表示へ (GO) コマンド
 一般メニュー仕様 275
 使用 261
 メニュー変更 (CHGMNU) コマンド 276
 メニュー・オブジェクト、定義
 *DSPF メニュー 267
 *UIM メニュー 308
 メニュー・グループ (MENUGRP) タグ
 許可されるタグの表 641
 メニュー・グループ (MNUGRP) タグ
 概要 641
 メニュー・バー
 アプリケーションとの相互作用 197
 作成 157
 使用 390
 定義 157
 プルダウン・メニューとの相互作用 207
 領域のヘルプ 431
 例 157, 209
 メニュー・バー (MBAR) タグ
 概要 633
 許可されるタグの表 634
 例、メニュー・バー 634
 メニュー・バー切り替えキー
 定義 171
 メニュー・バー区切り記号
 作成 157
 定義 157, 158
 抑制 158
 メニュー・バー選択項目
 その変換 159
 定義 157
 メニュー・バー選択項目 (MBARC) タグ
 概要 636
 許可されるタグの表 636
 メニュー・バー選択項目のヘルプ
 例 203
 メニュー・バーとプルダウン・メニューのキーの相互作用 207
 メニュー・バー表示 (MNUBARDSP) キーワード
 アプリケーション・レコード 197, 209
 メニュー・バー・レコード 199, 212

メニュー・バー・カーソル・アクション (ACTIONS) ダイアログ・コマンド 702
 メニュー・バー・パネル
 アプリケーション・プログラミング 363
 作成 351
 ソース 354
 メニュー・バー・ヘルプ・リスト
 例 205
 メニュー・バー・レコード
 MNUBARDSP 212
 メンバー
 ソース
 定義 4
 モード、表示
 サブファイルを横方向に表示する 111
 定義 111
 変更 111
 文字
 選択 520, 682
 対称
 アラビア語 536
 ヘブライ語 536
 文末 512
 文字 ID (CHRID) キーワード
 説明 289
 例 289
 文字値
 表示画面形式 538
 文字セットおよびコード・ページ
 印刷時の考慮事項 369
 変換、DDS 289
 変換、UIM 532
 文字フィールド属性 24
 文字変換
 使用 290
 戻りコード
 ディスプレイ・ファイル
 説明 739
 メジャー 00 739
 メジャー 02 740
 メジャー 03 740
 メジャー 04 741
 メジャー 08 および 11 741
 メジャー 34 742
 メジャー 80 743
 メジャー 81 746
 メジャー 82 748
 メジャー 83 752
 バイナリー・プログラム・メニュー
 270
 メジャー/マイナー
 参照：メジャー/マイナー戻りコード
 戻り点 (RTNPNT) パラメーター
 プログラム・メニュー用 273

[ヤ行]

ユーザー画面管理 (USRDSMGT) キーワード
 使用 763
 共に使用できない DDS キーワード 764
 ユーザー定義 (USRDFN) キーワード
 参照：ユーザー定義のデータ・ストリーム (UDDS)
 ユーザー定義のデータ・ストリーム (UDDS)
 コマンド機能キーに対する影響 279
 コマンドを要求するシステム操作 280
 コマンド・アテンション・キーに対する影響 279
 出力データ・ストリームの定義 279
 出力バッファの定義 279
 使用 277
 使用可能な機能 278
 制限 278
 制限について 278
 それを含むディスプレイ・ファイルのオープン 279
 ディスプレイ装置の違いについて 277
 入出力要求に対する影響 278
 入力操作 280
 ヘルプ情報 278
 ユーザー定義の要求後のフィールド・レベル要求の処理 278
 読み書き機能 279
 読み書き操作
 CL (制御言語) 279
 ILE RPG 279
 レコード様式の出力レコード域 278
 3270 モデル 4 ディスプレイ装置サポート 279
 5250 ディスプレイ装置用のバッファースタイル 279
 USRDFN キーワードとともに有効なキーワード 278
 USRDFN (ユーザー定義) キーワード 277
 ユーザー定義の編集コード
 それに関する 38
 編集記述 759
 ユーザー定義のメニュー
 参照：メニュー表示
 ユーザー定義パネル・グループ
 参照：オンライン・ヘルプ情報
 ユーザー復元画面 (USRRSTDSP) キーワード 139
 ユーザー・インターフェース管理機能 (UIM)
 アクション変数定義 375, 376
 アプリケーションを開く 365

ユーザー・インターフェース管理機能
(UIM) (続き)
アプリケーション・ウィンドウ、定義
393
アプリケーション・ハンドル定義 365
アプリケーション・プログラミング
確認処理 333
データ表示パネル 347
メニュー・バー・パネル 363
リスト・パネル 327
印刷
印刷パネル 398
印刷ヘッド・パネル 398
概念 396
概要 396
強調表示およびフォント 398
共用プリンター・ファイルのオーバ
ーライドおよび共用 399
コード・ページおよび文字セット
369
トレーラー 398
フォントおよび強調表示 398
ブランク行による分離 398
プリンター・ファイル、共用および
オーバーライド 399
プリンター・ファイルの共用および
オーバーライド 399
プロローグ域 398
分離のためのブランク行 398
ページ排出 399
ヘッダー域 399
文字セットおよびコード・ページ
369
2 バイト文字セット (DBCS) 400
DBCS (2 バイト文字セット) 400
ウィンドウ
アプリケーションの定義 393
コマンド行 394
コマンド行内の例 394
使用 393
除去 393
スタックを表示するための追加
393
追加 393
追加および除去のための API 393
別のパネルへの追加 389
API を使用する除去 393
API を使用する追加および除去
393
カーソル位置に基づいて表示されるヘル
プ・テキスト 380
概要 303
確認処理
アプリケーション・プログラミング
333
自動的に 332

ユーザー・インターフェース管理機能
(UIM) (続き)
確認パネルのソース 330
確認リスト・パネル、作成 329
各国語に関する考慮事項および
DBCS 447
機能キー
書式設定 383
スクロール 378
定義 383
VARUPD 値 384
機能キーの書式設定 383
グラフィカル・ユーザー・インターフ
ェース (GUI) 378
グラフィカル・ユーザー・インターフ
ェース (GUI) の使用可能化 378
経路指定ステップ 377
考慮事項、使用前の 301
コマンド行
ウィンドウ 394
ウィンドウの例 394
解釈 382
制約事項 382
長すぎる入力 383
ポップアップ・ウィンドウ 394
索引検索
ヘルプ・パネル・アクセス権の付与
423
作成
オブジェクト 305
確認リスト・パネル 329
データ表示パネル 333
メニュー・バー・パネル 351
メニュー・パネル 309
リスト・パネル 316
サポートされる機能 304
サポートされるパネル・エレメント
304
自動確認処理 332
詳細 365
処置リスト定義 375
スクロール 378
エラー条件 379
機能キー 378
状況情報 378
情報域 379
スクロール可能域の定義 378
スクロール可能な領域 378
データ項目グループ 380
テキスト域 380
不完全リスト 379
メッセージ行 378
メニュー域 379
リスト域 379
領域の定義 378

ユーザー・インターフェース管理機能
(UIM) (続き)
生産性
アプリケーション・プログラマーの
増大 301
向上 301
生産性を高めるために使用する場合
301
制約事項
ダイアログ変数の使用 367
5250 ラベル付きキー 378
選択文字 377
選択リスト定義 376
ソース
確認パネルの例 330
データ表示パネルの例 337
メニュー・バー・パネルの例 354
リスト・パネルの例 319
その代わりに DDS を使用 301
ダイアログ変数
エラー・メッセージ 367
初期値 366
制約事項 367
定義 365
不正な文字 368
ダイアログ変数内の不正な文字 368
ダイアログ変数のエラー・メッセージ
367
ダイアログ変数の初期値 366
ダイアログ変数を使用する表示パネル
のフィールド値 368
データ入力パネル 350
データ表示パネル
作成 333
スクロール後の例 335
ソース 337
例 333
定義 3
アプリケーション・ウィンドウ
393
機能キー 383
機能キーおよび VARUPD 値 384
ダイアログ変数 365
パネル・グループ・オブジェクト
315
文脈ヘルプ 380
メニュー・オブジェクト 308
出口フラグ 387
パネル機能の管理 377
パネルの書式設定
アプリケーション制御 386
いつ行うか 385
概念 385
定義 385
パネルを表示する際の要求処理プログ
ラム 395

ユーザー・インターフェース管理機能
(UIM) (続き)

- パネル・グループ
 - 定義 304
 - 要素 305
- パネル・グループ内のエレメント 305
- パネル・グループ・オブジェクト、定義 315
- パネル・グループ・オブジェクトの定義 315
- パネル・フォーマッターの制限 387
- パネル・リストをまとめて閉じる 388
- 表示文字、不正な 368
- 不正な表示文字 368
- 不正なプリンター文字 368
- プリンター文字、不正な 368
- プルダウン・メニューとポップアップ・ウィンドウとの違い 391
- 文脈ヘルプ、定義 380
- 別のパネルへのウィンドウの追加 389
- 別のパネルへのポップアップ・ウィンドウの追加 389
- ヘルプ
 - 階層、索引検索 446
 - 索引検索 423
 - 索引検索階層 446
 - ユーザー定義パネル・グループ 423
- ヘルプ・ウィンドウ 134
- ヘルプ・テキストが表示されるときのカーソル位置 380
- 変数プール・サービス 368
- ポップアップ・ウィンドウ
 - アプリケーションの定義 393
 - コマンド行 394
 - コマンド行内の例 394
 - 使用 393
 - 除去 393
 - スタックを表示するための追加 393
 - 追加 393
 - 追加および除去のための API 393
 - 別のパネルへの追加 389
 - API を使用する除去 393
 - API を使用する追加および除去 393
- ポップアップ・ウィンドウとプルダウン・メニューとの違い 391
- メッセージ行、スクロール 378
- メニュー域、スクロール 379
- メニュー定義 305
- メニューの例のソース 311
- メニュー・オブジェクト、定義 308
- メニュー・オブジェクトの定義 308
- メニュー・バー 390
- メニュー・バー・パネル 351

ユーザー・インターフェース管理機能
(UIM) (続き)

- アプリケーション・プログラミング 363
 - 作成 351
 - ソース 354
 - 例 350
- メニュー・パネル
 - 作成 309
 - 例 309
- 文字セットおよびコード・ページの変換 532
- ユーザー生産性の増大 301
- ユーザー定義パネル・グループヘルプ・パネル・アクセス権の付与 423
- 要求処理プログラムを使用するパネルの表示 395
- よく尋ねられる質問 400
- リスト
 - 管理 371, 372
 - 更新 373
 - 項目の除去 373
 - 項目の挿入 373
 - 初期化 372
 - 定義 372
 - 表示 372
 - 不完全な処理 373
- リスト画面
 - リスト項目の制御 374
- リスト画面上的リスト項目の制御 374
- リスト画面の応答時間 374
- リスト画面の対話応答時間 374
- リストからの項目の除去 373
- リストからの項目の挿入 373
- リスト項目
 - ハンドル 373
 - ポインター 374
 - リスト画面上 374
- リストの更新 373
- リストの項目ポインター 374
- リストの初期化 372
- リストの表示 372
- リストの例の代替ビュー 317
- リスト・パネル
 - アプリケーション・プログラミング 327
 - 作成 316
 - ソース 319
 - まとめて閉じる 388
 - 例 316
- リスト・パネルをまとめて閉じる 388
- 例
 - メニューのソース 311
 - メニュー・バー・パネル 350
 - リストの代替ビュー 317

ユーザー・インターフェース管理機能
(UIM) (続き)

- 例 (続き)
 - リスト・パネル 316
- 5250 ラベル付きキーの制約事項 378
- API サービス 307
- DBCS および各国語に関する考慮事項 447
- EXIT の要求時に複数のパネルをまとめて閉じる 387
- EXIT の要求時にまとめて閉じる 387
- VARUPD 値および機能キー 384
- 参照: パネル・グループ
- 様式、レコード
 - 参照: レコード様式
- 呼び出し先プログラム
 - よく尋ねられる質問 400
- 読み取り
 - ウィンドウからのデータ 136
- 読み取り操作
 - サブファイル
 - BASIC ステートメント 109
 - ILE COBOL ステートメント 109
 - ILE C/C++ 関数 109
 - ILE RPG 演算命令 109
 - サブファイルの定義 109
 - 待機を指定した
 - CL コマンド 91
 - ILE COBOL ステートメント 91
 - ILE C/C++ 関数 91
 - ILE RPG 操作 91
 - 待機を指定しない
 - CL コマンド 91
 - ILE COBOL ステートメント 91
 - ILE RPG 操作 91
 - 定義 76
 - 読み取り操作の取り消し、CL コマンド 91

[ラ行]

ライブラリー

- QUSRSYS ライブラリー 291
- QUSRTOOL
 - 画面の例 455
 - 作成時にワード処理機能を使用したパネル・グループ 404
 - 定義 8
- ラジオ・ボタン
 - 定義 151
 - 例 151
- ラベル付きキーの制約事項 378
- リスト
 - 一部 633
 - 管理 371
 - キー 596

- リスト (続き)
 - 更新 373
 - 項目の除去 373
 - 項目の挿入 373
 - 項目の追加 373
 - 順序 645
 - 順不同 690
 - 初期化 372
 - 単純 678
 - パラメーター 657
 - 表示 372
 - 不完全な処理 373
- リスト域
 - 印刷 608
 - スクロール 379
 - ヘルプ 422
 - 例
 - リスト・グループ列を持つ 611
 - 3つのレイアウト列を持つ 610
- リスト域 (LIST) タグ
 - 印刷のフォーマットの考慮事項 608
 - 概要 603
 - 許可されるタグの表 604
 - 例
 - 動的リスト列見出しのフォーマット 612
 - リスト域 609
 - リスト・グループ列を持つリスト域 611
 - 3つのレイアウト列を持つリスト域 610
- リスト域のスクロール 379
- リスト画面
 - 位置指定
 - カーソルの規則 484
 - リスト 480
 - リストの小文字名 481
 - エラー条件の規則 485
 - 応答時間 374
 - オプション行 476
 - オンライン・ヘルプ情報
 - 説明 498
 - タイプ 498
 - ヘルプ域 498
 - DDSの考慮事項 499
 - 拡張処置項目域 478
 - 空の状態での形式 486
 - 指示行 476
 - 説明 474
 - 操作
 - オプションが指定されている場合のリスト 483
 - ガイドライン 479
 - 拡張処置項目域 481
 - リスト制御フィールド 480
 - そのリスト項目 374
- リスト画面 (続き)
 - タイトル 475
 - プロンプト域 476
 - ページング位置情報 478
 - リスト形式、空のリスト 486
 - リスト制御フィールドの変更 481
 - リスト・フィールド 478
 - ルール、エラー条件 485
 - 例
 - 拡張処置項目域 474
 - CUA 506
 - 列見出し 477
 - 1つの選択項目のリスト 486
- リスト画面上のリスト項目 374
- リスト画面の対話応答時間 374
- リスト画面のリスト・フィールド 478
- リストからの項目の挿入 373
- リスト項目のハンドル 373
- リスト項目ハンドル 373
- リスト項目ポインター 374
- リスト定義 (LISTDEF) タグ
 - 概要 625
 - 強調値表 627
- リストの一部 (LP) タグ
 - 概要 633
 - ヘルプ・モジュールでの使用 440
 - 例、リストの一部 633
- リストの管理 371
- リストの項目ポインター 374
- リストの初期化 372
- リスト列 (LISTCOL) タグ
 - 概要 621
 - フォーマットの考慮事項 625
- リスト列グループ (LISTGRP) タグ
 - 概要 628
- リスト・アクション (LISTACT) タグ
 - 概要 614
 - 推奨される規則 619
 - 必須の規則 619
 - 例、リスト・アクション 620
- リスト・パネル
 - アプリケーション・プログラミング 327
 - 作成 316
 - ソース 319
- リスト・ビュー (LISTVIEW) タグ
 - 概要 631
 - レイアウトおよび位置の要約表 632
- リソース割り振り
 - システムが割り振る方法についての理解 45
 - 説明 45
 - ディスプレイ装置のエラー処理に対する影響 45
- リテラル、図形 517
- 利点
 - プログラマブル・マウス・ボタン 217
- 略号
 - コーディング、無効 186
 - それが有効であるコーディング 186
 - 定義 151, 186
- 略号が有効である DDS
 - 例 186
- 領域
 - パネル上の 513
- 例
 - アクセラレーター・キー 171
 - アプリケーション・フォーマット済み領域 520
 - アプリケーション・レコードでの MNUBARDSP の定義 197, 209
 - 印刷タイトル行 673
 - ウィンドウの使用法 141
 - ウィンドウ・タイトル 136
 - オーバーライド
 - 異なるディスプレイ・ファイル 237
 - 異なるディスプレイ・ファイルおよび属性 238
 - ディスプレイ・ファイルの属性 236
 - プログラムのコンパイル時の適用 239
- 押しボタン 183
- オンライン・ヘルプ情報
 - 応答標識を使用した HLPCMDKEY キーワード 419
 - オプション標識を使用した HLPPNLGRP キーワード 410
 - オプション標識を使用した HLPRCD キーワード 412
 - コマンド・ヘルプ 450
 - パネル・グループ中に組み込まれるヘルプ・モジュール 438
 - パネル・グループ中のインポートされるヘルプ・モジュール 439
 - ヘルプ域を表示する画面 406
 - ヘルプ・モジュールを使用したパネル・グループ 438
 - 2次ヘルプ 417
 - HLPCMDKEY キーワード 419
 - HLPPNLGRP キーワード 409
 - HLPRCD キーワード 412
 - ISCH タグ 444
 - ISCHSYN タグ 444
 - LINK タグ 449
- オンライン・ヘルプのあるサンプル画面の作成 3
- キー定義 595
- キー・リスト 596
- 組み込みヘルプ 585

例 (続き)

クラス定義 536
グラフィック・イメージ
HTML キーワード 231
グリッド・ライン構造 227
グループを持つメニュー域 639
継続属性 511
継続入力フィールド 190
コマンド行ポップアップ・ウィンドウ
394
索引検索 589
索引検索同義語 592
サブファイル
可変長レコード 113
サブファイルのプログラム論理
124
情報変更用のサンプル画面 96
選択項目のあるサンプル画面 95
タスクの組み合わせ用サンプル画面
96
縦方向および横方向、同時に表示
109
入力専用サンプル画面 96
表示専用サンプル画面 95
複数のレコードをロールする際のサ
ンプル画面 121
メッセージ・サブファイル 116
DSPATR(PC) キーワード 118
DSPMOD キーワード 111
ROLLUP、SFLSIZ、および
SFLPAG キーワード 112
SFLNXTCHG キーワード 115
SFLPAG、SFLSIZ、および
ROLLUP キーワード 112
SFLSIZ、ROLLUP、および
SFLPAG キーワード 112
*DS3 モード表示 112
*DS4 モード表示 111
サンプル図表 578
サンプル見出し 697
出力操作
OVERLAY および ERASE の影響
54
順不同リスト 691
条件付けオプション 543
情報画面 471
真理値表 689
垂直単一項目選択フィールド 162
垂直複数項目選択フィールド 163
水平選択フィールド 163
数値の表 538
スクロール後のデータ表示パネル 335
スクロール・バー 173, 178
制御言語プログラム
プログラム・メニュー 271
選択フィールド 161, 162

例 (続き)

選択フィールド内で抑制された選択標
識 166
選択不可能な項目 185
選択リスト 174
選択リストで使用可能になった選択標
識 177
タイトルの引用 524
妥当性検査 524
単一項目選択フィールド選択項目のヘ
ルプ 204
単一項目選択リスト 173
短縮定義 577
単純なメニュー域 638
単純リスト 679
段落タグ 648
段落の連結 658
チェック・ボックス 151
注記 645
データ記述仕様 (DDS)
応答標識を使用した HLPCMDKEY
キーワード 419
オプション標識を使用した
HLPPNLGRP キーワード 410
オプション標識を使用した
HLPRCD キーワード 412
可変長レコード 113
画面サイズ、複数の 39
サブファイルのプログラム論理
124
ディスプレイ・ファイル・ソース
23
ディスプレイ・ファイル・メニュー
267
プログラム・メニュー 271
メッセージ・サブファイル 116
2 次ヘルプ 417
2 つの画面サイズ用のサブファイル
40
5250 ディスプレイ装置の問題 87
ALWROL キーワード 60
CHRID キーワード 289
CLRL と OVERLAY キーワードの
相違 58
CLRL(*NO) キーワード 57
DSPATR(PC) キーワード 118
DSPMOD キーワード 70
DSPMOD キーワードとサブファイ
ル 111
ERRSFL キーワード 250
HLPCMDKEY キーワード 419
HLPPNLGRP キーワード 409
HLPRCD キーワード 412
INZRCD および UNLOCK キーワ
ード 79

例 (続き)

データ記述仕様 (DDS) (続き)
OVERLAY と CLRL キーワードの
相違 58
PUTOVR キーワード 62
PUTOVR キーワード、繰り返し入
力されるデータ 65
PUTOVR キーワード、より効果的
なコーディング 63
PUTRETAIN キーワード、フィー
ルド・レベル 67
PUTRETAIN キーワード、レコー
ド・レベル 67
ROLLUP、SFLSIZ、および
SFLPAG キーワード 112
RSTDSP パラメーター 85
SFLMSGID キーワード 251
SFLNXTCHG キーワード 115
SFLPAG、SFLSIZ、および
ROLLUP キーワード 112
SFLSIZ、ROLLUP、および
SFLPAG キーワード 112
SLNO(*VAR) キーワード 55
UNLOCK および INZRCD キーワ
ード 79
データ入力パネル 548, 570
データ入力パネル、2 列形式 549
データの受け渡し
同じ経路指定ステップ内の 283
経路指定ステップ間での 284
データ表示パネル 333
定義リスト 577
テキスト域 684
動的リスト列見出し 612
入力画面
典型的な 465
CUA 504
ネストされたデータ・グループを持つ
データ入力パネル 553
ハイパーテキスト・リンク 601
パネル定義 656
パネル・グループ
インポートされるヘルプ・モジュ
ール 439
オプション標識を使用した
HLPPNLGRP キーワード 410
組み込みヘルプ・モジュール 438
コマンド・ヘルプ 450
ヘルプ・モジュール 438
HLPPNLGRP キーワード 409
ISCH タグ 444
ISCHSYN タグ 444
LINK タグ 449
パラメーター・リスト 658
番号付きリスト 646

例 (続き)

左から右方向のパネル上の、左から右方向のフォーマット設定 678
 フィールド戻し (RTNFLD) パラメーター 202
 複数項目選択リスト 173
 複数選択フィールド (UIM) 571
 不定形式行 597
 プルダウン・メニュー 167, 209
 プルダウン・メニュー内で抑制された選択標識 169
 プログラム記述のディスプレイ・ファイル 42
 ヘルプ・パネル定義 581
 ヘルプ・モジュール
 インポートされる 439
 組み込み 438
 コマンド・ヘルプ 450
 を使用したパネル・グループ 438
 ISCH タグ 444
 ISCHSYN タグ 444
 LINK タグ 449
 変換リスト 687
 編集記述、作成 761
 右から左方向のパネル上の、左から右方向のフォーマット設定 677
 見出しタグ 584
 メッセージ
 ERRSFL キーワード 250
 SFLMSGID キーワード 251
 メッセージ、トレーラー 675
 メニュー画面
 単一選択 462
 メニューのソース 311
 メニュー表示
 オプションおよび GO コマンドの使用 262
 ディスプレイ・ファイル・メニュー 267
 プログラム・メニュー 271
 プログラム・メニュー、CL ソース 271
 プログラム・メニューに対する RTNPNT(*NO)、CL ソース 273
 メッセージ・ファイル、メッセージの追加 269
 メッセージ・ファイル、メニュー・アクションの記述 265
 メッセージ・ファイル内でのヘルプ・フォーマットの命名 265
 メニュー・バー
 選択項目 157
 パネル 351
 DDS 209
 UIM 634

例 (続き)

メニュー・バー選択項目のヘルプ 203
 メニュー・バー・ヘルプ・リスト 205
 メニュー・バー・レコードでの MNUBARDSP の定義 199, 212
 メニュー・パネル
 作成 309
 要求に対する応答として表示される、出力および入力フィールドのある画面 26
 ラジオ・ボタン 151
 リスト
 順不同 691
 単純 679
 番号付き 646
 変換 687
 リスト域
 リスト列グループ 611
 例、UIM ソース 609
 3 つのレイアウト列 610
 リスト画面
 拡張処置項目域 474
 CUA 506
 リストの一部 633
 リストの代替ビュー 317
 リスト・アクション 620
 リスト・パネル 316
 略号、それが無効である DDS 186
 略号が有効である DDS 186
 例のフォーマット設定 698
 レコード、DDS
 応答標識を使用した HLPCMDKEY キーワード 419
 オプション標識を使用した HLPKRD キーワード 412
 2 次ヘルプ 417
 HLPCMDKEY キーワード 419
 HLPKRD キーワード 412
 レコード様式の置換 52
 ロール操作 60
 2 つの表示域
 データ項目 551
 メニュー域 552
 CL プログラム
 プログラム・メニューに対する RTNPNT(*NO)、CL ソース 273
 HTML キーワード
 グラフィック・イメージ 231
 フィールドのオーバーラップの解決 229
 PULLINPUT パラメーター 200
 SLNO (開始行番号) キーワード 55
 例 (XMP) タグ
 概要 697
 ヘルプ・モジュール 441
 例、例のフォーマット設定 698

レイアウト

タイトル行の表 671
 レイアウトおよび位置の要約表 632
 レコード、DDS
 オーバーレイ中にディスプレイから消去 54
 画面間のページング 415
 初期化 (図) 79
 制約 414
 送信時のパフォーマンスの向上 296
 重複させるが削除しないで、システム・パフォーマンスを向上させる 296
 定義 18
 ディスプレイ上での保持 66
 ディスプレイ上のスペースを占有しない 52
 ディスプレイへの配置 49
 入力フィールドを含む、それを持つ場合のパフォーマンスの向上 296
 パネル・グループと文書の比較 404
 表示、複数 54
 表示するモード 69
 複数の行に表示するための規則 49
 ヘルプ
 指定 411
 の定義 452
 ヘルプの定義 8
 ヘルプ用の入力 413
 変更データ・タグのリセット 66
 有効な配置の図 50
 例
 応答標識を使用した HLPCMDKEY キーワード 419
 オプション標識を使用した HLPKRD キーワード 412
 2 次ヘルプ 417
 HLPCMDKEY キーワード 419
 HLPKRD キーワード 412
 2 次ヘルプ、順序の決定 416
 2 次ヘルプに関する制約事項 418
 2 次ヘルプの表示 416
 I/O ステートメントで使用 48
 WAIT コマンドでの検索 75
 レコード域、そこからのオプション標識と応答標識の除去 30
 レコード初期設定 (INZRCD) キーワード 77
 レコード様式 48
 アクティブの判別 49
 画面上のビュー 24
 記述 23
 出力および入力 48
 それに対する変更の確認 22
 置換 52
 定義 3

レコード様式 (続き)
ディスプレイ上でのオーバーレイおよび消去 53
ディスプレイ上での置換 52
ディスプレイ上での変更 52
ディスプレイに書き込まれる順序 53
ディスプレイ・ファイル用の最大数 30
特定の行からの開始 54
内容 23
表示フィールドと機能の定義 23
フィールド、使用法 25
フィールドの最大数 26
レコード様式テーブル、アクティブ 49
レコード・レベル記述 18
定義 18
フィールド・レベル記述のある 18
フィールド・レベル記述のない 18
列データ
フォーマット 625
列見出し
リスト画面 477
レベル識別コード
決定 22
その変更 22
定義 22
連結記号 515
ロールアップ (ROLLUP) キーワード
SFLSIZ および SFLPAG の指定例 112
ロール可能 (ALWROL) キーワード
使用 59
ロールに対する影響 60
ロールダウン (ROLLDOWN) キーワード
機能キー用 27
ロックされたディスプレイ・ファイル 45

[ワ行]

ワード、索引検索のルート
参照： オンライン・ヘルプ情報
割り当て、共通キー
参照： 共通キーの割り当て

[数字]

1 次ヘルプ、定義 415
16 進データ 518
2 次ヘルプ
定義 415
レコードの順序を決定する 416
レコードの表示 416
2 進数値
書式設定 537
表示画面形式 537, 538

2 バイト文字セット
略号 187
5250 ディスプレイ装置
レコード様式の問題の回避 87
UNLOCK の結果 77
5250 ディスプレイ・データ・ストリーム
参照： ユーザー定義のデータ・ストリーム (UDDS)
5250 ラベル付きキーの制約事項 378

A

ACTIONS (メニュー・バー・カーソル・アクション) ダイアログ・コマンド 702
ADDMSGD (メッセージ記述追加) コマンド 269
ADDSCHIDX (探索索引項目追加) コマンド 451
ALCOBJ (オブジェクト割り振り) コマンド 45
ALTHELP (代替ヘルプ) キーワード 29
ALTPAGEDWN (代替次ページ) キーワード 29
ALTPAGEUP (代替前ページ) キーワード 29
ALWROL (ロール可能) キーワード
使用 59
ロールに対する影響 60
API (アプリケーション・プログラミング・インターフェース)
ウィンドウの追加および除去 393
サービス 307
APPFMT (アプリケーション・フォーマット済み領域) タグ
アプリケーション・フォーマット済みデータ 519
概要 518
例
アプリケーション・フォーマット済み領域 520
ASSUME (想定) キーワード
考慮事項 283
プログラム間でのファイルの共用 283
DDS ソース例 283, 284
OVERLAY キーワード、それとの併用 284
System/36 アプリケーション互換性 764
USRDSMGT キーワードでの制限 774

B

BASETYPE
値 526
要約表 530
BASIC ステートメント
サブファイルに関する入出力要求 109
BOTINST (最下部指示) タグ
概要 521

C

CALL (プログラム呼び出し) ダイアログ・コマンド 703
CANCEL ダイアログ・コマンド 705
CAnn (コマンド・アテンション) キー
オンライン・ヘルプ情報からプログラムに戻る 419
システム・パフォーマンス考慮事項 297
妥当性検査の考慮事項 87
定義 28
CAnn (コマンド・アテンション) キーワード
キーボード制御用 27
CFNN (コマンド機能) キー
オンライン・ヘルプ情報からプログラムに戻る 419
CFnn (コマンド機能) キー
システム・パフォーマンス考慮事項 297
妥当性検査の考慮事項 87
定義 28
CHCACCEL キーワード 170
CFnn (コマンド機能) キーワード 27, 28
CHANGE (変更) キーワード 78
CHECK (検査) キーワード
カーソル移動のパラメーター 31
RL および RLTB パラメーター 31
CHECK (妥当性検査) タグ
概要 522
例、妥当性検査 524
CHECK(FE) (フィールド終了) キーワード 167
CHGCMDDFT (コマンド・デフォルト変更) コマンド 275
CHGDEVDS (装置記述の変更 (表示装置)) コマンド 41
CHGDSPF (ディスプレイ・ファイル変更) コマンド
実行 21
変更
ディスプレイ・ファイル 21
ファイル記述 18
DFRWRT パラメーター 295

- CHGDSPPF (ディスプレイ・ファイル変更) コマンド (続き)
 ENHDSPP (拡張表示) パラメーター 154
 SHARE パラメーター 296
- CHGDSPPF (表示装置ファイル変更) コマンド
 DFRWRT (書き込み据え置き) パラメーター
 説明 68
 ディスプレイ・ファイルの作成時 68
- CHGJOB (ジョブ変更) コマンド 253
- CHGMNU (メニュー変更) コマンド 276
- CHGVVIEW (ビュー変更) ダイアログ・コマンド 705
- CHKOBJ (オブジェクト検査) 関数 541, 600
- CHKPGM (COND 評価プログラム) 542
- CHKPGM (条件評価プログラム) 601
- CHKUSRCLS (ユーザー・クラス検査) 関数 542, 601
- CHRID (文字 ID) キーワード
 説明 289
 例 289
- CHRID (文字 ID) パラメーター
 値、表 292
 指定 291
- CIT (タイトルの引用) タグ
 概要 524
 ヘルプ・モジュールでの使用 440
 例、タイトルの引用 524
- CL (制御言語) プログラム
 使用した入出力操作 91
 送信勧誘された装置からの読み取り 75
 入力の送信勧誘 73
 例
 プログラム・メニュー 271
 プログラム・メニューに対する RTNPNT(*NO) 273
- CL プログラム作成 (CRTCLPGM) コマンド 272
- CLASS (クラス定義) タグ
 概要 525
 許可されるタグの表 525
 表示画面形式
 時刻値 538
 数値 537
 日付値 538
 文字値 538
 方向付け用の決定木 534
 例
 クラス定義 536
 数値の表 538
 BASETYPE 要約表 530
- CLEAR (消去) キーワード 27
- CLRL (行の消去) キーワード
 システム・パフォーマンス考慮事項 296
 使用 57
 例
 OVERLAY との相違 58
 (*NO) 値を指定 57
 S/36 環境の考慮事項 764
- CMD (システム・コマンド) ダイアログ・コマンド 706
- CMDLINE (コマンド行) ダイアログ・コマンド 707
- CMDLINE (コマンド行) タグ
 概要 539
- COLOR (カラー) キーワード 37
- COND (条件定義) タグ
 概要 540
 例、条件付けオプション 543
- CHKOBJ (オブジェクト検査) 組み込み関数 541
- CHKPGM (COND 評価プログラム) 542
- CHKUSRCLS (ユーザー・クラス検査) 組み込み関数 542
- COPYR (著作権) タグ
 概要 544
- CPF4001-40FF メッセージ 255
- CPF4101-43FF メッセージ 255
- CPF4401-44FF メッセージ 255
- CPF4501-46FF メッセージ 255
- CPF4701-48FF メッセージ 255
- CPF4901-49FF メッセージ 255
- CPF5001-50FF メッセージ 255
- CPF5101-53FF メッセージ 255
- CPF5501-56FF メッセージ 255
- CRTCLPGM (CL プログラム作成) コマンド 272
- CRTCMD (コマンド作成) コマンド 450
- CRTDSPF (ディスプレイ・ファイル作成) コマンド
 画面の作成 7
 ディスプレイ・ファイルの作成 20
 ディスプレイ・ファイル・メニュー 268
 プログラム・メニュー 271
- DFRWRT (書き込み据え置き) パラメーター
 書き込み操作 68
 それによるパフォーマンスの向上 295
- ENHDSPP (拡張表示) パラメーター 154
- SHARE パラメーター 296
- CRTDUPOBJ (複製オブジェクト作成) コマンド
 使用 275
- CRTEDTD (編集記述作成) コマンド
 実行 39
 使用 761
 例 761
- CRTMNU (メニュー作成) コマンド
 ディスプレイ・ファイル・メニュー 269
 プログラム・メニュー 272
- CRTMSGF (メッセージ・ファイル作成) コマンド 269
- CRTPNLGRP (パネル・グループ作成) コマンド
 画面の作成 8
 使用 450
- CRTSCHIDX (検索索引作成) コマンド 451
- CRTTBL (テーブル作成) コマンド 292
- CSRINPONLY (入力可能位置にのみカーソル移動) キーワード 31
- CSRLOC (カーソル位置) キーワード 71
- CUA (共通ユーザー・アクセス)
 機能キー域、メッセージ行の関係 502
 基本ダイアログ処置、その表 501
 システムとの関係 455
 情報画面 505
 説明 501
 選択項目、選択項目入力フィールド 503
 単一選択フィールド操作、ガイドライン 503
 入力画面 504
 入力フィールド 504
 メニュー画面、単一選択 502
 リスト画面、例 506

D

- DATA (データ表示域) タグ
 概要 544
 許可されるタグの表 545
 例
 データ項目用の 2 つの表示域 551
 データ入力パネル 548
 データ入力パネルでの 2 列形式 549
 ネストされたデータ・グループを持つデータ入力パネル 553
 メニュー域を持つデータ表示域 552
- DATA (データ項目選択) タグ
 概要 554
- DATACOL (データ列) タグ
 概要 555

DATAGRP (データ・グループ) タグ
 概要 556

DATAI (データ項目) タグ
 概要 559

DATAIX (データ項目エクステンダー) タグ
 概要 564

DATASLT (データ選択フィールド) タグ
 概要 568
 許可されるタグの表 568
 例
 データ入力パネル 570
 複数選択フィールド 571

DATASLTC (データ選択フィールド選択項目) タグ
 概要 573

DBCS (2 バイト文字セット) 印刷 400

DBCS および各国語に関する考慮事項 447

DBCS グリッド・ライン構造
 参照: グリッド・ライン構造

DBCS フィールド
 継続入力フィールド 191

DDS キーワード
 参照: キーワード, DDS

DDS (データ記述仕様)
 ウィンドウ 127
 サブファイル、記述 97
 サブファイル制御の処理順序 103
 使用 17
 情報画面 498
 妥当性検査キーワード 132
 定義 3
 ディスプレイ・ファイル
 経路指定ステップ間でのデータの受け渡し 284
 使用 23
 プログラム間でのデータの受け渡し 283

入力画面 497
 表示モードが変更される場合に無視されるキーワード 70

フィールド
 カラー (表) 37
 強調 (表) 36

プログラミング例 141
 プログラム論理 124
 ヘルプ・レコード 132
 変更 21
 メニュー画面 496
 リスト画面 499

例
 応答標識を使用した HLPCMDKEY キーワード 419
 押しボタン 183

DDS (データ記述仕様) (続き)
 例 (続き)
 オプション標識を使用した HLPPNLGRP キーワード 410
 オプション標識を使用した HLPRCD キーワード 412
 可変長レコード 113
 グリッド・ライン構造 227
 スクロール・バー 181
 選択フィールド 161
 選択リスト 174
 ディスプレイ・ファイル・ソース 23
 ディスプレイ・ファイル・メニュー 267
 プログラム・メニュー 271
 メッセージ・サブファイル 116
 略号 186
 2 次ヘルプ 417
 2 つの画面サイズ 39
 2 つの画面サイズ用のサブファイル 40
 5250 ディスプレイ装置の問題 87
 CLRL と OVERLAY キーワードの相違 58
 DSPATR(PC) キーワード 118
 DSPMOD キーワード 70
 DSPMOD キーワードとサブファイル 111
 ERRSFL キーワード 250
 GRDATR (グリッド・ライン属性) キーワード 227
 GRDBOX (グリッド・ボックス) キーワード 227
 GRDCLR (グリッド消去) キーワード 227
 GRDLIN (グリッド・ライン) キーワード 227
 GRDRCD (グリッド・レコード) キーワード 227
 HLPCMDKEY キーワード 419
 HLPPNLGRP キーワード 409
 HLPRCD キーワード 412
 OVERLAY と CLRL キーワードの相違 58
 PULLINPUT キーワード 200
 PUTOVR キーワード 62
 PUTOVR キーワード、繰り返し入力されるデータ 65
 PUTOVR キーワード、より効果的なコーディング 63
 PUTRETAIN キーワード、フィールド・レベル 67
 PUTRETAIN キーワード、レコード・レベル 67

DDS (データ記述仕様) (続き)
 例 (続き)
 ROLLUP、SFLSIZ、および SFLPAG キーワード 112
 SFLMSGID キーワード 251
 SFLNXTCHG キーワード 115
 SFLPAG、SFLSIZ、および ROLLUP キーワード 112
 SFLSIZ、ROLLUP、および SFLPAG キーワード 112

CAnn キー、制約事項 29

CHECK(FE) (フィールド終了) キーワード 167

DDS 用紙 23

DSPRL (右から左へ表示) キーワード 34

ERRMSG (エラー・メッセージ) キーワード 132

ERRMSGID (エラー・メッセージ識別コード) キーワード 132

ERRSFL (エラー・サブファイル) キーワード 132

MSGLOC (メッセージ位置) キーワード 132

NOCCSID (コード化文字セット ID 変換なし) キーワード 291

RMVWDW (ウィンドウの除去) キーワード 138

SFLMSG (サブファイル・メッセージ) キーワード 132

SFLMSGID (サブファイル・メッセージ識別コード) キーワード 132

USRRSTDSP (ユーザー復元画面) キーワード 139

WDWBORDER (ウィンドウ枠) キーワード 133

WINDOW (ウィンドウ) キーワード 129
 参照: キーワード, DDS
 参照: ディスプレイ・ファイル

DDS 用紙 23

DDS レコード
 参照: レコード, DDS

DFRWRT (書き込み据え置き) パラメーター
 使用 68
 制約事項 69
 それによるパフォーマンスの向上 295

DFT (デフォルト) キーワード
 使用 69
 ASSUME キーワードと併用 283

DFTVAL (デフォルト値) キーワード 69

DL (定義リスト) タグ
 概要 575
 許可されるタグの表 576
 ヘルプ・モジュールでの使用 440

DL (定義リスト) タグ (続き)
例
短縮定義リスト 577
定義リスト 577
DLTMNU (メニュー削除) コマンド 276
DLTOVR (オーバーライド削除) コマンド
240
DLTPNLGRP (パネル・グループ削除) コ
マンド 450
DLTSCHIDX (検索索引項目削除) コマン
ド 452
DSPATR (表示属性) キーワード
使用 71
プログラム-システム間フィールドで
37
ASSUME キーワードと併用 283
PC 値のソースの例 118
DSPEDTD (編集記述表示) コマンド 39
DSPFD (ファイル記述表示) コマンド 23
DSPFFD (ファイル・フィールド記述表示)
コマンド 22
DSPHELP (ヘルプ表示) ダイアログ・コマ
ンド 707
DSPMNUA (メニュー属性表示) コマンド
276
DSPMOD (表示モード) キーワード
サブファイルの使用 111
サブファイルの例 111
使用 69
説明 70
表示モード (DSPMOD) 111
*DS3 のサンプル・サブファイル・デ
ィスプレイ 112
*DS4 のサンプル・サブファイル・デ
ィスプレイ 111
DSPOVR (オーバーライド表示) コマンド
240
DSPPGMREF (プログラム参照表示) コマ
ンド 23
DSPRL (右から左へ表示) キーワード 34
DSPSIZ (画面サイズ) キーワード
使用 39
パフォーマンスの向上のために指定
295
DTAQ (データ待ち行列) パラメーター
287

E

EDTCDE (編集コード) キーワード 38
EDTWRD (編集語) キーワード 38
embed
参照: imbed
ENDRCV (受信終了) コマンド 74
ENHDSP (拡張表示) パラメーター
レコードの書き込み 153

ENTER ダイアログ・コマンド 708
ERASE (消去) キーワード
レコードの消去 54
OVERLAY キーワードを指定 54
System/36 アプリケーション互換性
764
USRDSPPMGT キーワードでの制限
774
ERASEINP (入力消去) キーワード
システム・パフォーマンス考慮事項
297
使用 66
ERRMSG (エラー・メッセージ) キーワー
ド 132, 248
ERRMSGID (エラー・メッセージ識別コー
ド) キーワード 132, 248
ERRSFL (エラー・メッセージ・サブファ
イル) キーワード
から受け取るメッセージ 132
使用 250
例の DDS ソース 250
USRDSPPMGT キーワードでの制限
774
EXIT の要求時に複数のパネルをまとめて
閉じる 387
EXIT (表示終了) ダイアログ・コマンド
709
EXTHELP (拡張ヘルプ) ダイアログ・コ
マンド 710

F

F18=続きの見出し
アクセス権の除去 424
FIG (図表) タグ
概要 577
ヘルプ・モジュールでの使用 441
例、サンプル図表 578
FLDCSRPRG (カーソル移動先フィールド)
キーワード 33

G

GETRETAIN (データ保持 GET) キーワー
ド 78
GO (メニュー表示へ) コマンド
一般メニュー仕様 275
使用 261
GRDATR (グリッド・ライン属性) キーワ
ード 227
GRDBOX (グリッド・ボックス) キーワー
ド 227
GRDCLR (グリッド消去) 227
GRDLIN (グリッド・ライン) 227
GRDRCD (グリッド・レコード) 227

GUI (グラフィカル・ユーザー・インター
フェース)
サポート 378

H

H 仕様
定義 405
ページングの順序 415
ヘルプ・リストの更新に対する影響
415
DDS ソースでの順序 406
DDS ソースの入力 406
H1 から H4 (見出し)
概要 583
ヘルプ・モジュール 440
例
見出しタグ 584
HELP ダイアログ・コマンド 710
HELP (ヘルプ) キーワード
機能キーの使用 27
代替の指定 29
ヘルプ・キーの使用可能化 404
System/36 アプリケーション互換性
771
HELP (ヘルプ・モジュール) タグ
概要 579
ヘルプ・モジュールを使用したパネ
ル・グループを表示する UIM ソー
ス 438
例、ヘルプ・パネル定義 581
UIM ソース中の配置 438
HELPHelp ダイアログ・コマンド 710
HELPIDX ダイアログ・コマンド 711
HLL (高水準言語)
サポートされる入出力要求の認識 91
プログラムでのディスプレイ・ファイ
ル属性のオーバーライド 235
プログラムでのディスプレイ・ファイ
ル名および属性のオーバーライド
238
プログラムでのディスプレイ・ファイ
ル名のオーバーライド 237
System/36 アプリケーション互換性
763
HLPARA (ヘルプ域) キーワード 406
HLPBDY (ヘルプ境界) キーワード
使用 416
2 次ヘルプ 417
HLPAMDKEY (ヘルプ・コマンド・キー)
キーワード
応答標識を使用したサンプル・ソース
419
使用 419
DDS ソースのサンプル 419

HLPCMDKEY (ヘルプ・コマンド・キー) キーワード (続き)
 USRDSPMGT キーワードでの制限 774

HLPEXCLD (ヘルプの除外) キーワード 407

HLPFULL (ヘルプの全画面表示) キーワード 407

HLPPNLGRP (ヘルプ・パネル・グループ) キーワード
 オプション標識の使用 410
 使用 407
 例
 オプション標識を使用したサンプル・ソース 410
 DDS ソースのサンプル 409
 HLPKRD および HLPDOC キーワードの使用 410

HLPKRD (ヘルプ・レコード) キーワード
 同じファイルでの HLPDOC キーワードの使用 413
 同じファイルでの HLPPNLGRP キーワードの使用 411
 オプション標識の使用 412
 例
 オプション標識を使用したサンプル・ソース 412
 2 次ヘルプ 417
 DDS ソースのサンプル 412

HLPRTN (ヘルプからの戻り) キーワード
 ヘルプの後にプログラムに戻る 420
 System/36 アプリケーション互換性 771

HLPSCHIDX (ヘルプの検索索引) キーワード 407

HLPSEQ (ヘルプの順序付け) キーワード 416

HLPTITLE (ヘルプのタイトル) キーワード 407

HOME (ホーム) キーワード 27

HOME (ホーム・メニュー表示) ダイアログ・コマンド 711

HP0 から HP9 (強調表示句) タグ
 概要 582
 ヘルプ・モジュールでの使用 440

HTML (ハイパーテキスト・マークアップ言語) キーワード
 制約事項 233
 挿入 229
 フィールドのオーバーラップの解決 229
 例 229

ICF ファイル
 ディスプレイ・ファイル、データ待ち行列でデータ待ち 287
 ファイル入力フィールド属性 287

IGC (漢字) 書式設定
 BOTINST (最下部指示) タグ 521
 DATAC (データ項目選択) タグ 555
 TOPINST (最上部指示) タグ 688
 フォーマット
 PDFLDC (プルダウン・フィールド選択項目) タグ 663

ILE COBOL ステートメント
 サブファイルに関する入出力要求 109
 入出力操作へのマッピング 91

ILE C/C++ 関数
 サブファイルに関する入出力要求 109
 入出力操作へのマッピング 91

ILE RPG 演算命令
 サブファイルに関する入出力要求 109

ILE RPG 操作
 入出力操作へのマッピング 91

IMHELP (組み込みヘルプ) タグ
 概要 584
 ヘルプ・モジュールを使用したパネル・グループを表示する UIM ソース 438
 例、組み込みヘルプ 585
 を使用したパネル・グループを表示する UIM ソース 438
 UIM ソース中の配置 438

IMPORT (インポート) タグ
 概要 586
 を使用したパネル・グループを表示する UIM ソース 439
 UIM ソース中の配置 438

INDARA (標識域) キーワード 30

INFO (情報域) タグ
 印刷のフォーマットの考慮事項 588
 概要 587

INVITE (送信勧誘) キーワード
 システム・パフォーマンス考慮事項 297
 使用 73

INZINP (入力初期設定) キーワード
 パフォーマンスに関する考慮事項 297

INZRCD (レコード初期設定) キーワード
 使用 77
 UNLOCK キーワードを指定した例 79

ISCH (索引検索) タグ
 パラメーター・リスト 588
 UIM ソース中の配置 444

ISCHSUBT (索引検索サブトピック) タグ
 パラメーター・リスト 590
 UIM ソース中の配置 446

ISCHSYN (索引検索同義語) タグ
 パラメーター・リスト 591
 UIM ソース中の配置 444

K

KEEP (保持) キーワード
 考慮事項 283
 プログラム間でのファイルの共用 283
 DDS ソース例 284
 System/36 アプリケーション互換性 764
 USRDSPMGT キーワードでの制限 774

KEYI (キー・リスト項目) タグ
 概要 593
 例、キー定義 595

KEYL (キー・リスト) タグ
 概要 596
 許可されるタグの表 596
 例、キー・リスト 596

KEYSHelp ダイアログ・コマンド 712

L

LINES (不定形式行) タグ
 概要 597
 ヘルプ・モジュールでの使用 441
 例、不定形式行 597

LINK (ハイパーテキスト・リンク) タグ
 概要 598
 条件式
 CHKOBJ 組み込み関数 600
 CHKUSRCLS (ユーザー・クラス検査) 組み込み関数 601
 例、ハイパーテキスト・リンク 601
 を使用したパネル・グループを表示する UIM ソース 449
 CHKPGM (条件評価プログラム) 601
 UIM ソース中の配置 449

LIST (リスト域) タグ
 印刷のフォーマットの考慮事項 608
 概要 603
 許可されるタグの表 604
 例
 動的リスト列見出しのフォーマット 612
 リスト域 609
 リスト・グループ列を持つリスト域 611
 3 つのレイアウト列を持つリスト域 610

LISTACT (リスト・アクション) タグ
概要 614
推奨される規則 619
必須の規則 619
例、リスト・アクション 620

LISTCOL (リスト列) タグ
概要 621
フォーマットの考慮事項 625

LISTDEF (リスト定義) タグ
概要 625
強調値表 627

LISTGRP (リスト列グループ) タグ
概要 628

LISTVIEW (リスト・ビュー) タグ
概要 631
レイアウトおよび位置の要約表 632

LOCK (ロック) キーワード 84

LP (リストの一部) タグ
概要 633
ヘルプ・モジュールでの使用 440
例、リストの一部 633

M

Main Menu 画面 262

MBAR (メニュー・バー) タグ
概要 633
許可されるタグの表 634
例、メニュー・バー 634

MBARC (メニュー・バー選択項目) タグ
概要 636
許可されるタグの表 636

MDT (変更データ・タグ)
定義 26
文字変換 290
リセット 66
ASSUME キーワードと併用 283

MDTOFF (変更データ・タグ・オフ) キーワード 66

MENU オブジェクト・タイプ 261

MENU ダイアログ・コマンド 712

MENU (メニュー域) タグ
許可されるタグの表 637
例
グループを持つメニュー域 639
単純なメニュー域 638

MENUGRP (メニュー・グループ) タグ
概要 641
許可されるタグの表 641

MENUI または MI (メニュー項目) タグ
概要 642

MNUBARDSP (メニュー・バー表示) キーワード
アプリケーション・レコード 197,
209
メニュー・バー・レコード 199, 212

MOREKEYS (機能キー続き表示) ダイアログ・コマンド 713

MOUBTN (プログラマブル・マウス・ボタン) キーワード 216

MOVETOP (先頭へ移動) ダイアログ・コマンド 713

MSG (メッセージ表示) ダイアログ・コマンド 714

MSGALARM (メッセージ・アラーム) キーワード 253

MSGID (メッセージ識別コード) キーワード 249

MSGLOC (メッセージ位置) キーワード 132

N

NOCCSID (コード化文字セット ID 変換なし) キーワード 291

NOTE (注記) タグ
ヘルプ・モジュールでの使用 439

NT (注記) タグ
許可されるタグの表 644
ヘルプ・モジュールでの使用 439
例、注記の使用 645

NT または NOTE (注記) タグ
概要 644

O

ODP (オープン・データ・パス)
共用、同じジョブの 296
システム・パフォーマンスの向上のため 296
定義 92

OL (番号付きリスト) タグ
概要 645
許可されるタグの表 646
ヘルプ・モジュールでの使用 440
例、番号付きリスト 646

OPTLINE (オプション行) タグ
概要 647

OVERLAY (オーバーレイ) キーワード
システム・パフォーマンス考慮事項 297
説明 53
ヘルプ・リストの更新に対する影響 415
例
CLRL との相違 58
ERASE キーワードを指定 54
レコードの消去 54
System/36 アプリケーション互換性 764

OVRATR (属性一時変更) キーワード
システム・パフォーマンス考慮事項 296
使用 61

OVRDSPF (ディスプレイ・ファイル・オーバーライド) コマンド
ファイル属性 235
ファイル名 237
ファイル名および属性 238
CHGDSPF との比較 18

OVRDTA (データ一時変更) キーワード
システム・パフォーマンス考慮事項 296
説明 63

P

P (段落) タグ
概要 648
ヘルプ・モジュールでの使用 439
例、段落タグ 648
UIM ソース中の配置 438

PAGEDOWN (次ページ) キーワード
機能キー用 27
代替の指定 29

PAGEDOWN ダイアログ・コマンド 715

PAGEUP (前ページ) キーワード
機能キー用 27
代替の指定 29

PAGEUP ダイアログ・コマンド 715

PANEL (表示パネル) タグ
概要 649
許可されるタグの表 649
例、パネル定義 656

PARM (パラメーター・リスト) タグ
概要 656
許可されるタグの表 657
ヘルプ・モジュールでの使用 440
例、パラメーター・リスト 658

PASSRCD (受け渡しレコード) キーワード
使用 284
DDS ソース例 284

PC (段落の連結) タグ
概要 658
ヘルプ・モジュールでの使用 439
例、段落の連結 658

PDACCEL (ブルダウン・アクセラレータ) タグ
概要 659

PDFLD (ブルダウン・フィールド) タグ
概要 659
許可されるタグの表 660

PDFLDC (ブルダウン・フィールド選択項目) タグ
概要 660

PDFLDC (プルダウン・フィールド選択項目) タグ (続き)
推奨される規則 664
必須規則 663

PDM (プログラム開発管理機能)
定義 4

PK (プログラミング・キーワード) タグ
概要 664
ヘルプ・モジュールでの使用 441

PNLGRP (パネル・グループ) タグ
概要 665
許可されるタグの表 665
ヘルプ・モジュールを使用したパネル・グループを表示する UIM ソース 438
UIM ソース中の配置 438

PRINT (印刷) キーワード
得られる結果 (表) 40
機能キー用 27
使用可能化 40

PRINT (画面印刷) ダイアログ・コマンド 716

Print キー
得られる結果 (表) 40
使用可能化 40

PRINTER パラメーター 41

Problem Handling メニュー 262

Programming メニュー 262

PROMPT ダイアログ・コマンド 716

PRTHEAD (印刷ヘッド・パネル) タグ
概要 668
許可されるタグの表 669
タイトル行のレイアウトの表 671, 672
例、印刷タイトル行 673

PRTPNL (印刷パネル) タグ
概要 673
許可されるタグの表 674

PRTRAIL (トレーラー・メッセージ印刷) タグ
概要 675
例、トレーラー・メッセージ 675

PSHBTNCHC (押しボタン選択項目) キーワード 183

PSHBTNFLD (押しボタン・フィールド) キーワード 183

PULLDOWN ダイアログ・コマンド 718

PULLINPUT (プルダウン・メニュー入力) パラメーター
例 200

PUTOVR (明示的な一時変更を伴う PUT) キーワード
システム・パフォーマンス考慮事項 296
説明 61

PUTOVR (明示的な一時変更を伴う PUT) キーワード (続き)
例
繰り返し入力されるデータ 65
サンプル DDS ソース 62
より効果的なコーディング 63
System/36 アプリケーション互換性 769

PUTRETAIN (データ保持 PUT) キーワード
使用 66
例
フィールド・レベル 67
レコード・レベル 67
System/36 アプリケーション互換性 764
USRDSMGT キーワードでの制限 774

PV (プログラミング変数) タグ
概要 675
ヘルプ・モジュールでの使用 441

Q

QRCVDTAQ (データ待ち行列から受信) プログラム 287

QSNDDTAQ (データ待ち行列へ送信) プログラム 287

QUSRSYS ライブラリー 291

QUSRTOOL ライブラリー
画面の例 455
作成時にワード処理機能を使用したパネル・グループ 404
定義 8

R

RETCMDKEY (コマンド・キー保持) キーワード
System/36 アプリケーション互換性 773

RETKEY (機能キー保持) キーワード
System/36 アプリケーション互換性 774

RETRIEVE (コマンド・ストリング取得) ダイアログ・コマンド 719

RETURN (アプリケーションへの制御戻り) ダイアログ・コマンド 719

RMVSHCHDXE (探索索引項目除去) コマンド 451

RMVWDW (ウィンドウの除去) キーワード 138

ROLLDOWN (ロールダウン) キーワード
応答標識 30
機能キー 27

ROLLUP (ロールアップ) キーワード
応答標識 30
機能キー 27
サブファイル・パフォーマンスの向上 295
SFLSIZ および SFLPAG の指定例 112

RSTDSP (画面復元) パラメーター
システム・パフォーマンス考慮事項 297
使用 85
必要なキーワード 297
例 85

RT (テキスト反転) タグ
概要 676
例
左から右方向のパネル上の、左から右方向のフォーマット設定 678
右から左方向のパネル上の、左から右方向のフォーマット設定 677

RTNCSRLOC (カーソル位置戻し) キーワード 72

RTNFLD (フィールド戻し) パラメーター
説明 202
例 202

S

SDA (画面設計機能)
定義 8
DDS ソースの直接入力との比較 20

SETOF または SETOFF (オフに設定) キーワード
その応答標識 31

SEU 開始 (STRSEU) コマンド 22

SEU (原始ステートメント入力キューティリティー)
定義 6
「Edit」画面 6

SFL (サブファイル) キーワード 97

SFLCHCCTL (サブファイル選択制御) キーワード 175

SFLCLR (サブファイル消去) キーワード
説明 98

SFLCSRPRG (サブファイル・カーソル移動先) キーワード 33

SFLCSRPRN (サブファイル・カーソル相対レコード番号) キーワード 72

SFLCTL (サブファイル制御) キーワード 97

SFLDLT (サブファイル削除) キーワード 98

SFLDROP (サブファイル切り捨て) キーワード 98

SFLDSP (サブファイル表示) キーワード 97

- SFLDSPCTL (サブファイル表示制御) キーワード 98
- SFLEND (サブファイル終了) キーワード 98
- SFLENTER (サブファイル実行キー) キーワード 98
- SFLFOLD (サブファイル継続) キーワード 98
- SFLINZ (サブファイル初期設定) キーワード 98
- SFLLIN (サブファイル行) キーワード 98
- SFLMLTCHC (サブファイル複数選択項目選択) キーワード 174
- SFLMODE (サブファイル・モード) キーワード 72
- SFLMSG (サブファイル・メッセージ) キーワード
からの結果のメッセージ 132
サブファイル制御レコードの書き込み時にメッセージ行に表示する 248
説明 98
- SFLMSGID (サブファイル・メッセージ識別コード) キーワード
からの結果のメッセージ 132
サブファイル制御レコードの書き込み時にメッセージ行に表示する 248
説明 98
例の DDS ソース 251
- SFLMSGKEY (サブファイル・メッセージ・キー) キーワード
説明 98
プログラム・メッセージ待ち行列からメッセージを表示する 250
- SFLMSGRCDD (サブファイル・メッセージ・レコード) キーワード
説明 98
プログラム・メッセージ待ち行列からメッセージを表示する 250
- SFLNXTCHG (次に変更するサブファイル) キーワード
説明 98
例の DDS 115
- SFLPAG (サブファイル・ページ) キーワード
サブファイル制御レコード様式での使用 97
ROLLUP および SFLSIZ の指定例 112
- SFLPGMQ (サブファイル・プログラム・メッセージ待ち行列) キーワード
説明 98
プログラム・メッセージ待ち行列からメッセージを表示する 250
- SFLRCDNBR (サブファイル・レコード番号) キーワード
サブファイル・パフォーマンスの向上 295
説明 98
- SFLRNA (サブファイル・レコード非活動) キーワード 98
- SFLROLVAL (サブファイル・ロール値) キーワード 98
- SFLRTNSEL (サブファイル戻り選択済み項目) キーワード 176
- SFLSCROLL (サブファイル・スクロール) キーワード 176
- SFLSIZ (サブファイル・サイズ) キーワード
サブファイル制御レコード様式での使用 97
ROLLUP および SFLPAG の指定例 112
- SFLSNGCHC (サブファイル単一選択項目選択) キーワード
サブファイル複数選択項目選択 (SFLMLTCHC) 174
SFLMLTCHC (サブファイル複数選択項目選択) 174
- SHARE (共用) パラメーター
システム・パフォーマンス考慮事項 296
同一ジョブ内でのディスプレイ・ファイルの共用 92
- SL (単純リスト) タグ
概要 678
許可されるタグの表 678
ヘルプ・モジュールでの使用 440
例、単純リスト 679
- SLNO (開始行番号) キーワード
値と結果の表 54
使用 54
制約事項 55
他の DDS キーワードと一緒に使用 55
例 55
*VAR 値を指定 55
「Specify Members to Work with」画面 4
「Start Source Entry Utility (SEU)」画面 5
- STRPDM (プログラム開発管理機能開始) コマンド 3
- STRSCHIDX (検索索引開始) コマンド 443
- STRSEU (SEU 開始) コマンド 22
- System/36 アプリケーション互換性機能キー 771
高水準言語 763
送信勧誘取り消し操作 772
- System/36 アプリケーション互換性 (続き)
ディスプレイ・ファイル 763
入力データ 765
ヘルプからの戻り (HLPRTN) キーワード 771
メッセージ 767
ASSUME (想定) キーワード 764
ERASE (消去) キーワード 764
HELP (ヘルプ) キーワード 771
HLPRTN (ヘルプからの戻り) キーワード 771
KEEP (保持) キーワード 764
OVERLAY キーワード 764
PUTOVR (明示的な一時変更を伴う PUT) キーワード 769
PUTRETAIN (データ保持 PUT) キーワード 764
- ## T
- TI (変換リスト項目) タグ
概要 685
- TL (変換リスト) タグ
概要 686
許可されるタグの表 686
リスト、変換 687
- TOPINST (最上部指示) タグ
概要 687
- TT (真理値表) タグ
概要 688
許可されるタグの表 689
例、真理値表 689
- TTROW (真理値表行) タグ
概要 689
- ## U
- UDDS (ユーザー定義のデータ・ストリーム)
コマンド機能キーに対する影響 279
コマンドを要求するシステム操作 280
コマンド・アテンション・キーに対する影響 279
出力データ・ストリームの定義 279
出力バッファの定義 279
使用 277
使用可能な機能 278
制限 278
制限について 278
それを含むディスプレイ・ファイルのオープン 279
ディスプレイ装置の違いについて 277
入出力要求に対する影響 278
入力操作 280
ヘルプ情報 278

- UDDS (ユーザー定義のデータ・ストリーム) (続き)
 - ユーザー定義の要求後のフィールド・レベル要求の処理 278
 - 読み書き機能 279
 - 読み書き操作
 - CL (制御言語) 279
 - ILE COBOL 279
 - ILE RPG 279
 - レコード様式の出カレコード域 278
 - 3270 モデル 4 ディスプレイ装置サポート 279
 - 5250 ディスプレイ装置用のバッファースタイル 279
 - USRDFN キーワードとともに有効なキーワード 278
 - USRDFN (ユーザー定義) キーワード 277
- UIM
 - よく尋ねられる質問 400
- UIM ソース・ファイルの埋め込み 517
- UIM でのリスト定義 372
- UIM (ユーザー・インターフェース管理機能) 372
 - アクション変数定義 375, 376
 - アプリケーションを開く 365
 - アプリケーション・ウィンドウ、定義 393
 - アプリケーション・ハンドル定義 365
 - アプリケーション・プログラミング
 - 確認処理 333
 - データ表示パネル 347
 - メニュー・バー・パネル 363
 - リスト・パネル 327
- 印刷
 - 印刷パネル 398
 - 印刷ヘッド・パネル 398
 - 概念 396
 - 概要 396
 - 強調表示 398
 - 強調表示およびフォント 398
 - コード・ページおよび文字セット 369
 - トレーラー 398
 - フォントおよび強調表示 398
 - ブランク行による分離 398
 - プリンター・ファイル、共用およびオーバーライド 399
 - プリンター・ファイルのオーバーライドおよび共用 399
 - プリンター・ファイルの共用およびオーバーライド 399
 - プロローグ域 398
 - 分離のためのブランク行 398
 - ページ排出 399
 - ヘッダー域 399
- UIM (ユーザー・インターフェース管理機能) (続き)
 - 印刷 (続き)
 - 文字セットおよびコード・ページ 369
 - 2 バイト文字セット (DBCS) 400
 - DBCS (2 バイト文字セット) 400
 - ウィンドウ
 - アプリケーションの定義 393
 - コマンド行 394
 - コマンド行内の例 394
 - 使用 393
 - 除去 393
 - スタックを表示するための追加 393
 - 追加 393
 - 追加および除去のための API 393
 - 別のパネルへの追加 389
 - API を使用する除去 393
 - API を使用する追加および除去 393
 - カーソル位置に基づいて表示されるヘルプ・テキスト 380
 - 概要 303
 - 確認処理
 - アプリケーション・プログラミング 333
 - 自動的に 332
 - 確認パネルのソース 330
 - 確認リスト・パネル、作成 329
 - 各国語に関する考慮事項および DBCS 447
 - 機能キー
 - 書式設定 383
 - スクロール 378
 - 定義 383
 - VARUPD 値 384
 - 機能キーの書式設定 383
 - グラフィカル・ユーザー・インターフェース (GUI) 378
 - 経路指定ステップ 377
 - 考慮事項、使用前の 301
 - コマンド行
 - ウィンドウ 394
 - ウィンドウの例 394
 - 解釈 382
 - 制約事項 382
 - 長すぎる入力 383
 - ポップアップ・ウィンドウ 394
- 索引検索
 - ヘルプ・パネル・アクセス権の付与 423
- 作成
 - オブジェクト 305
 - 確認リスト・パネル 329
 - データ表示パネル 333
- UIM (ユーザー・インターフェース管理機能) (続き)
 - 作成 (続き)
 - メニュー・バー・パネル 351
 - メニュー・パネル 309
 - リスト・パネル 316
 - サポートされる機能 304
 - サポートされるパネル・エレメント 304
 - 自動確認処理 332
 - 詳細 365
 - 処置リスト定義 375
 - スクロール 378
 - エラー条件 379
 - 機能キー 378
 - 状況情報 378
 - 情報域 379
 - スクロール可能域の定義 378
 - スクロール可能な領域 378
 - データ項目グループ 380
 - テキスト域 380
 - 不完全リスト 379
 - メッセージ行 378
 - メニュー域 379
 - リスト域 379
 - 領域の定義 378
 - スクロール後の例 335
 - 生産性
 - アプリケーション・プログラマーの増大 301
 - 向上 301
 - 生産性を高めるために使用する場合 301
 - 制約事項
 - ダイアログ変数の使用 367
 - 5250 ラベル付きキー 378
 - 選択文字 377
 - 選択リスト定義 376
 - ソース
 - 確認パネルの例 330
 - データ表示パネルの例 337
 - メニューの例 311
 - メニュー・バー・パネルの例 354
 - リスト・パネルの例 319
 - その代わりに DDS を使用 301
 - ダイアログ変数
 - エラー・メッセージ 367
 - 初期値 366
 - 制約事項 367
 - 定義 365
 - 不正な文字 368
 - ダイアログ変数内の不正な文字 368
 - ダイアログ変数のエラー・メッセージ 367
 - ダイアログ変数の初期値 366

UIM (ユーザー・インターフェース管理機能) (続き)

- ダイアログ変数を使用する表示パネルのフィールド値 368
- データ入力パネル 350
- データ表示パネル
 - アプリケーション・プログラミング 347
 - 作成 333
 - ソース 337
 - 例 333
- 定義 3
 - アプリケーション・ウィンドウ 393
 - 機能キー 383
 - 機能キーおよび VARUPD 値 384
 - ダイアログ変数 365
 - パネル・グループ・オブジェクト 315
 - 文脈ヘルプ 380
 - メニュー・オブジェクト 308
- 出口フラグ 387
- パネル機能の管理 377
- パネルの書式設定
 - アプリケーション制御 386
 - いつ行うか 385
 - 概念 385
 - 定義 385
- パネルを表示する際の要求処理プログラム 395
- パネル・グループ
 - 定義 304
 - 要素 305
- パネル・グループ内のエレメント 305
- パネル・グループ・オブジェクト、定義 315
- パネル・フォーマッターの制限 387
- パネル・リストをまとめて閉じる 388
- 表示文字、不正な 368
- 不正な表示文字 368
- 不正なプリンター文字 368
- プリンター文字、不正な 368
- プルダウン・メニューとポップアップ・ウィンドウとの違い 391
- 文脈ヘルプ、定義 380
- 別のパネルへのウィンドウの追加 389
- 別のパネルへのポップアップ・ウィンドウの追加 389
- ヘルプ
 - 階層、索引検索 446
 - 索引検索 423
 - 索引検索階層 446
 - ユーザー定義パネル・グループ 423
- ヘルプ・ウィンドウ 134

UIM (ユーザー・インターフェース管理機能) (続き)

- ヘルプ・テキストが表示されるときのカーソル位置 380
- 変数プール・サービス 368
- ポップアップ・ウィンドウ
 - アプリケーションの定義 393
 - コマンド行 394
 - コマンド行内の例 394
 - 使用 393
 - 除去 393
 - スタックを表示するための追加 393
 - 追加 393
 - 追加および除去のための API 393
 - 別のパネルへの追加 389
 - API を使用する除去 393
 - API を使用する追加および除去 393
- ポップアップ・ウィンドウとプルダウン・メニューとの違い 391
- メッセージ行、スクロール 378
- メニュー域、スクロール 379
- メニュー定義 305
- メニュー・オブジェクト 308
- メニュー・バー 390
- メニュー・バー・パネル
 - アプリケーション・プログラミング 363
 - 作成 351
 - ソース 354
 - 例 351
- メニュー・パネル
 - 作成 309
 - 例 309
- 文字セットおよびコード・ページの交換 532
- ユーザー生産性の増大 301
- ユーザー定義パネル・グループ
 - ヘルプ・パネル・アクセス権の付与 423
- 要求処理プログラムを使用するパネルの表示 395
- リスト
 - 管理 371
 - 更新 373
 - 項目の除去 373
 - 項目の挿入 373
 - 初期化 372
 - 表示 372
 - 不完全な処理 373
- リスト画面
 - リスト項目の制御 374
- リスト画面上のリスト項目の制御 374
- リスト画面の応答時間 374
- リスト画面の対話応答時間 374

UIM (ユーザー・インターフェース管理機能) (続き)

- リストからの項目の除去 373
- リストからの項目の挿入 373
- リスト項目
 - ハンドル 373
 - ポインター 374
 - リスト画面上 374
- リストの更新 373
- リストの項目ポインター 374
- リストの初期化 372
- リストの表示 372
- リストの例の代替ビュー 317
- リスト・パネル
 - アプリケーション・プログラミング 327
 - 作成 316
 - ソース 319
 - まとめて閉じる 388
 - 例 316
- リスト・パネルをまとめて閉じる 388
- 例
 - メニューのソース 311
 - メニュー・バー・パネル 351
 - リストの代替ビュー 317
 - リスト・パネル 316
- 5250 ラベル付きキーの制約事項 378
- API サービス 307
- DBCS および各国語に関する考慮事項 447
- EXIT の要求時に複数のパネルをまとめて閉じる 387
- EXIT の要求時にまとめて閉じる 387
- VARUPD 値および機能キー 384
- 参照: パネル・グループ
- UL (順不同リスト) タグ
 - 概要 690
 - 許可されるタグの表 690
 - ヘルプ・モジュールでの使用 440
 - 例、順不同リスト 691
- UNLOCK (キーボードのロック解除) キーワード
 - 入力操作に対する影響 79
 - プログラムによるデータ処理中のキーボードのロック解除 77
 - INZRCD キーワードを指定した例 79
- USRDFN (ユーザー定義) キーワード
 - 参照: ユーザー定義のデータ・ストリーム (UDDS)
- USRDSMGT (ユーザー画面管理) キーワード
 - 応答標識の制限 774
 - サブファイルの制限 774
 - 使用 763
 - 制限された DDS キーワード 764, 774

USRRSTDSP (ユーザー復元画面) キーワード 139

V

VAR (変数定義) タグ

概要 692

VARRCD (変数レコード定義) タグ

概要 695

VARUPD 値および機能キー 384

VARUPD (変数更新) 属性

機能キー 384

ダイアログ・コマンド 702

KEYI タグ 594

PDFLDC タグ 662

W

WAIT (待機) コマンド 75

WAITFILE (ファイル待機) パラメーター

オープン操作 46

獲得操作 46

システムによるリソース割り振り 45

WAITRCD パラメーターのタイミング機能 75

WAITRCD (レコード待機) パラメーター
タイミング機能 75

WAIT(*NO) パラメーター 73

WDWBORDER (ウィンドウ枠) キーワード 133

WDWTITLE (ウィンドウ・タイトル) キーワード 135

WINDOW (ウィンドウ) キーワード 129
「Work with Members Using PDM」画面
5

WRDWRAP (語の折り返し) キーワード
35, 191

X

XH1 から XH4 (拡張ヘルプの見出し) タグ

概要 696

フォーマットの規則 696

ヘルプ・モジュールでの使用 440

例、サンプル見出し 697

XMP (例) タグ

概要 697

ヘルプ・モジュール 441

例、例のフォーマット設定 698

Z

Z 変数

定義 693

Z36ENV ダイアログ変数 693

ZCANCEL ダイアログ変数 693

ZDBCS ダイアログ変数 693

ZDSPSIZ ダイアログ変数 693

ZEXIT ダイアログ変数 693

ZJOB ダイアログ変数 693

ZJOBNBR ダイアログ変数 693

ZLMTCPB ダイアログ変数 693

ZMENU ダイアログ変数 693

ZMNULIB ダイアログ変数 693

ZSYSNAM ダイアログ変数 693

ZUSER ダイアログ変数 693

[特殊文字]

*DSPATR (表示属性) パラメーター 187

*FLD (フィールド) 特殊値 204

*JOBCCSID 特殊値 291

*PGM (プログラム) オブジェクト・タイ
プ

定義するメニュー・オブジェクト 271

*PGM メニューの定義 271



Printed in USA

SC88-4031-00



日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木3-2-12