



System i

ファイルおよびファイル・システム
スプール・ファイル

バージョン 6 リリース 1





System i

**ファイルおよびファイル・システム
スプール・ファイル**

バージョン 6 リリース 1

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、37 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 5761-SS1) のバージョン 6、リリース 1、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。

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： System i
Files and file systems
Spooled files
Version 6 Release 1

発 行： 日本アイ・ビー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

目次

| | |
|------------------------------|----|
| スプール・ファイル | 1 |
| スプール・ファイル用の PDF ファイル | 1 |
| スプーリングの概念 | 2 |
| 出力スプーリング | 2 |
| 出力待ち行列およびスプール・ファイル | 3 |
| デフォルトのシステム出力待ち行列 | 3 |
| スプーリング書き出しプログラム | 4 |
| スプール書き出しプログラムのコマンド | 4 |
| 入力スプーリング | 5 |
| ジョブ入力コマンド | 6 |
| インライン・データ・ファイル | 7 |
| インライン・データ・ファイルのオープンに関する考慮事項 | 8 |
| スプール・ファイルおよび出力待ち行列 | 8 |
| スプーリングの概要 | 9 |
| スプール・ファイル | 9 |
| 出力待ち行列 | 10 |
| 複数の出力待ち行列 | 11 |
| 出力待ち行列のリカバリー | 11 |
| スプール・ファイルのクリーンアップ | 12 |
| デフォルトの出力待ち行列 | 12 |
| 出力待ち行列上のスプール・ファイルの順序 | 13 |
| データ待ち行列のサポート | 14 |
| レコード・タイプ 01 のデータ待ち行列項目フォーマット | 17 |
| レコード・タイプ 02 のデータ待ち行列項目フォーマット | 18 |
| レコード・タイプ 03 のデータ待ち行列項目フォーマット | 20 |
| スプール・ファイル名 | 22 |
| スプール・ファイルのセキュリティー | 22 |
| 出力待ち行列のセキュリティー | 23 |

| | |
|-----------------------------------|----|
| QPRTJOB ジョブ | 24 |
| スプール・ファイル・サブシステム | 25 |
| スプール・ファイル・ライブラリー | 25 |
| 独立 ASP におけるスプール・ファイル | 26 |
| スプール・ファイルの管理 | 27 |
| スプール・ファイル・リストの表示 | 27 |
| スプール・ファイルの内容の表示 | 28 |
| スプール・ファイルに関連したメッセージの表示 | 28 |
| スプール・ファイルの保留 | 28 |
| スプール・ファイルの保留解除 | 29 |
| スプール・ファイルの移動 | 29 |
| スプール・ファイルの削除 | 30 |
| スプール・ファイルの PDF への変換 | 30 |
| スプール・ファイルの物理ファイルへのコピー | 30 |
| スプール・ファイルの別のユーザーまたはシステムへの送信 | 30 |
| スプール・ファイルの属性の変更 | 31 |
| スプール・ファイルの印刷の再開 | 31 |
| 1 つのスプール・ファイルの中断および別のスプール・ファイルの印刷 | 31 |
| スプール・ファイル通知メッセージを有効にする | 32 |
| スプール・ファイル数の制御 | 32 |
| 有効期限が切れたスプール・ファイルの削除 | 32 |
| スプール・ファイル・ストレージの再利用 | 33 |
| スプール・ファイルの保存およびリストア | 34 |
| スプール・ファイル・サイズによる印刷の制御 | 35 |

| | |
|--------------------|----|
| 付録. 特記事項 | 37 |
| プログラミング・インターフェース情報 | 38 |
| 商標 | 39 |
| 使用条件 | 39 |

スプール・ファイル

スプーリングとは、後から処理や印刷を実行するためにスプール・ファイルにデータを保存するシステム機能です。スプール・ファイルは、テープ・ファイルや他の装置ファイルと同じような働きをします。スプール・ファイルを使用すれば、プリンターなどの外部接続装置を対象にしたデータを管理できます。

スプール・ファイルには、印刷できるまで出力データが保持されます。スプール・ファイルには、プログラムまたは装置がデータを処理できるまで装置からデータが収集されます。プログラムは実際の装置から読み取ったり書き出したりするかのようにしてスプール・ファイルを使用します。

スプーリング機能は、システム・ユーザーが入出力操作をより効率的に管理するのに役立ちます。

ジョブの終了時には、ジョブ・ログをスプール・ファイル `QPJOBLOG` に書き込んで印刷できるようにすることができます。ジョブ・ログがスプール・ファイルに書き込まれた後、ジョブ・ログは削除されます。

関連概念

スプール・ファイルおよび出力待ち行列

スプール・ファイル用の PDF ファイル

この情報の PDF ファイルを表示および印刷することができます。

本書の PDF バージョンを表示またはダウンロードするには、スプール・ファイルを選択します。

以下の関連したトピックを表示またはダウンロードできます。


- 制御言語には、制御言語 (CL) とそのコマンドに関する説明があります。各コマンドが定義され、構文図、パラメーター、デフォルト値、およびキーワードが含まれています。
- 印刷の基本は、プリンターを理解し制御するための情報を提供しています。その内容は、印刷エレメントと概念、印刷装置ファイル・サポート、印刷スプーリング・サポート、プリンター接続、高機能印刷、およびパーソナル・コンピューターでの印刷などが含まれています。

PDF ファイルの保管

表示または印刷のために PDF をワークステーションに保管するには、以下のようになります。

1. ご使用のブラウザで該当の PDF リンクを右クリックする。
2. PDF をローカルに保存するオプションをクリックします。
3. PDF を保存したいディレクトリーに進む。
4. 「保存」をクリックする。

Adobe Reader のダウンロード

これらの PDF を表示または印刷するには、システムに Adobe® Reader がインストールされている必要があります。Adobe Web サイト (www.adobe.com/products/acrobat/readstep.html)  から無料のコピーをダウンロードできます。

スプーリングの概念

システムは、出力スプーリングと入力スプーリングをサポートします。出力スプーリングは、プリンター装置の場合に使用することができます。入力スプーリングは、データベース・ファイル入力に適用されます。

出力スプーリング

出力スプーリングは、プリンターとディスク装置の両方の場合に使用することができます。出力スプーリングは、ジョブ出力をプリンターやディスク出力装置に直接送らず、ディスク装置に送ります。出力スプーリングを用いると、出力を作成するジョブは、出力装置の速度および可用性に関係なく、処理を続けることができます。

さらに、出力スプーリングを使用することによって、サーバーではプリンターやディスク装置など、複数の出力装置による出力を効率よく作成することができます。これは、プリンターに向けられたジョブ出力をディスク装置に送ることによって行われます。このプロセスによって、出力装置の可用性および速度による潜在的なジョブの制限から解放されます。

出力スプーリングの主な要素は、次のとおりです。

- **装置記述:** プリンターの装置記述。
- **スプール・ファイル:** 出力装置で処理されるスプール出力レコードを含むファイル。
- **出力キュー:** スプール・ファイルの順序リスト。
- **書き出しプログラム:** ファイルを出力待ち行列から装置に送るプログラム
- **アプリケーション・プログラム:** スプーリング属性として `SPOOL(*YES)` が指定されている装置ファイルを使用して、スプール・ファイルを作成する高水準言語プログラム。
- **装置ファイル:** 出力フォーマットの記述と、サーバーがスプール・ファイルを処理する方法を記述する属性のリスト。

出力スプーリング機能は、サーバーによって実行され、出力を作成するプログラムによる特別な操作を必要とはしません。ある装置ファイルがプログラムによってオープンされると、オペレーティング・システムで、出力をスプール出力とすべきかどうかを決めます。スプーリングを指定する印刷装置ファイルがオープンされると、プログラムの出力が入っているスプール・ファイルが、サーバーの中の該当する出力待ち行列に入れられます。

スプール・ファイルは、プリンター・ファイルが開かれた時点、プリンター・ファイルが閉じられた時点、またはジョブが終了した時点で印刷用に使用できるようになります。印刷装置書き出しプログラムがスプーリング・サブシステムの中で始動されて、レコードをプリンターに送ります。スプール・ファイルは出力待ち行列から選択されます。

装置記述のスプーリング

各プリンターおよびディスク装置ごとに装置記述を作成して、サーバーに対してその装置を定義しなければなりません。プリンターの装置記述は印刷装置記述作成 (`CRTDEVPR`) コマンドを使用して作成し、ディスクの装置記述はディスク装置記述作成 (`CRTDEVDKT`) コマンドを使用して作成します。

スプール・ファイルのファイル指定変更

スプール・ファイルが最初に意図されていたものとは異なる出力装置に送信されるときに、ファイルの指定変更が生じます。ファイル指定変更には、異なる媒体を処理する場合 (ディスク装置に送られたプリンター出力など)、または処理するのは同一タイプの媒体でも装置タイプが異なる場合 (4224 印刷装置に送られた 5219 印刷装置出力など) があります。

スプール・ファイルの新しい出力装置によっては、ファイルはもともと指定されていた装置で処理される場合と同じように処理されます。しかし、装置の違いにより、出力が異なった仕方で様式化されることもしばしばあります。これらの場合には、サーバーでは、書き出しプログラムのメッセージ待ち行列に照会メッセージを送ってユーザーに状況を通知し、ユーザーが印刷の続行を望むかどうかを指定できるようにします。

出力待ち行列およびスプール・ファイル

バッチ・ジョブおよび対話式ジョブ処理の結果として、プリンターやディスク駆動機構などのような出力装置で処理されるスプール出力レコードを作成する場合があります。これらの出力レコードはその処理が行われるときまで、スプール・ファイルに保管されています。単一ジョブは多くのスプール・ファイルを持つことができます。

作成されると、スプール・ファイルは出力待ち行列に入れられます。各出力待ち行列にはスプール・ファイルの順序リストが含まれます。1つのジョブで、スプール・ファイルは1つまたは複数の出力待ち行列を持つことができます。個々の出力待ち行列上のすべてのスプール・ファイルは、装置、用紙タイプ、および1インチ当たりの行数など共通の出力属性を持つべきです。1つの出力待ち行列で共通の属性を使用すれば、介入の必要が減り、装置スループットの増加を図ることができます。

次にリストするのは、出力待ち行列作成 (CRTOUTQ) コマンドのパラメーターおよびその指定内容です。

- MAXPAGES: 1日の開始から終了までの間に印刷することができる最大スプール・ファイル・サイズをページ単位で指定します。
- AUTOSTRWTR: この出力待ち行列に対して自動的に開始される書き出しプログラムの数を指定します。
- DSPDTA: 特殊権限は持っていないが、出力待ち行列に対する *USE 権限を備えているユーザーが、自分自身以外のスプール・ファイルの内容を表示、コピー、または送信することができるかを決めます。DSPDTA に *OWNER を指定すると、ファイルの所有者か *SPLCTL 特殊権限を付与されたユーザーだけが、ファイルの表示、コピー、または送信を実行できます。
- JOBSEP: 出力印刷時に、各ジョブの出力間に印刷するジョブ区切りページ (存在する場合) の数です。
- DTAQ: この出力待ち行列に関連したデータ待ち行列。これを指定した場合、スプール・ファイルが待ち行列で作動可能状態になるたびに入力項目がデータ待ち行列に送信されます。
- OPRCTL: ジョブ制御権限を持っているユーザーが、出力待ち行列を制御する (例えばそのユーザーが出力待ち行列を保留する) ことができるかを決めます。
- SEQ: スプール・ファイルが出力待ち行列上でソートされる順序を制御します。
- AUTCHK: 出力待ち行列に対するどのタイプの権限によって、ユーザーは出力待ち行列上のスプール・ファイルを制御することができるか (例えば、そのユーザーが出力待ち行列上のスプール・ファイルを保留することができるか) を指定します。
- AUT: 共通権限。ユーザーが出力待ち行列そのものに対してどのような制御権を持つかを指定します。
- TEXT: テキスト記述。出力待ち行列を記述する 50 文字までのテキスト。

デフォルトのシステム出力待ち行列

CL コマンドのデフォルトでは、スプールされたすべての出力のデフォルト出力待ち行列としてシステム・プリンターのデフォルト出力待ち行列が使用されます。システム・プリンターは QPRTDEV サーバー値によって定義されます。

スプール・ファイルが装置ファイルを開くことによって作成され、そのファイルについて指定されている出力待ち行列が見つからない場合、システムはそのスプール・ファイルをライブラリー QGPL の中の出力待ち行列 QPRINT に入れようと試みます。何らかの理由により、スプール・ファイルが出力待ち行列 QPRINT に入れられない場合はエラー・メッセージが送られ、出力はスプールされません。

以下の出力待ち行列が提供されます。

- **QDKT**: デフォルトのディスク出力待ち行列
- **QPRINT**: デフォルトのプリンター出力待ち行列
- **QPRINTS**: 特殊用紙用のプリンター出力待ち行列
- **QPRINT2**: 2 部用紙用のプリンター出力待ち行列

スプーリング書き出しプログラム

書き出しプログラムとは、スプールされたファイルを出力待ち行列から取り出して、出力装置上に作成する i5/OS® プログラムです。個々の出力待ち行列に入れられたスプール・ファイルは、書き出しプログラムがその出力待ち行列に対して開始されるまで、システムの中に保管されたままになっています。

書き出しプログラムは、出力待ち行列からスプール・ファイルを、優先順位に基づいて一度に 1 つずつ取り出します。書き出しプログラムがスプール・ファイル进行处理するのは、出力待ち行列上のその項目が、そのファイルが作動可能 (RDY) 状態であることを示す場合だけです。個々のスプール・ファイルの状況は、出力待ち行列処理 (WRKOUTQ) コマンドを使用して表示することができます。

スプール・ファイルが作動可能状態である場合は、書き出しプログラムは、その出力待ち行列からその項目を取り出し、指定されたジョブ区切りまたはファイル区切り (あるいはその両方) の後にファイルの中の出力データを続けて印刷します。スプール・ファイルが作動可能状態でない場合、書き出しプログラムはその項目を出力待ち行列に残して、次の項目に進みます。通常、書き出しプログラムは、作動可能状態のすべてのファイルが出力待ち行列から取り出されるまで (ジョブ区切りおよびファイル区切りが先行する) スプール・ファイル进行处理し続けます。

書き出しプログラム開始コマンドの AUTOEND パラメーターは、書き出しプログラムで新しいスプール・ファイルが書き出し可能になるのを待つか、1 つのファイル进行处理後終了するか、または作動可能状態のスプール・ファイルがすべて出力待ち行列から取り出された後終了するかを判別します。

スプール書き出しプログラムのコマンド

スプーリング書き出しプログラムの制御に使用できるコマンドを以下に示します。

- **ディスク書き出しプログラム開始 (STRDKTWTR)**: スプール書き出しプログラムを指定されたディスク装置に対して開始して、その装置上でスプール・ファイル进行处理します。
- **印刷装置書出プログラム開始 (STRPRTWTR)**: スプール書き出しプログラムを指定されたプリンター装置に対して開始して、その装置上でスプール・ファイル进行处理します。
- **遠隔書き出しプログラム開始 (STRRMTWTR)**: スプール・ファイルを出力待ち行列からリモート・システムに送るスプール書き出しプログラムを開始します。
- **書き出しプログラム変更 (CHGWTR)**: 用紙タイプ、ファイル区切りページ数、または出力待ち行列属性など、一部の書き出しプログラム属性を変更します。
- **書き出しプログラム保留 (HLDWTR)**: レコードの終わり、スプール・ファイルの終わり、またはページの終わりで書き出しプログラムを停止します。
- **書き出しプログラム解放 (RLSWTR)**: 次の処理に備えて、前に保留した書き出しプログラムを解放します。
- **書き出しプログラム終了 (ENDWTR)**: スプール書き出しプログラムを終了し、関連する出力装置をサーバーが使用できるようにします。

注: いくつかの機能を定義することにより、追加のスプーリング・サポートを提供できます。これらの機能に関するソース例およびコマンド、ファイル、およびプログラムの説明文が、i5/OS の一部としてオプションで導入されている QUSRTOOL ライブラリーに入っています。

入カスプーリング

入カスプーリングは、情報を入力装置から取り、ジョブのスケジューリングを準備し、ジョブ待ち行列に項目を入れます。入カスプーリングを使用すると、通常はジョブの実行時間が短縮され、順次に行われるジョブ数が増加し、装置のスループットが向上します。

入カスプーリングの主な要素は、次のとおりです。

- **ジョブ・キュー:** 実行に備えてシステムに実行依頼される順序付けられたバッチ・ジョブのリストで、バッチ・ジョブはそこから選択されて実行されます。
- **読み取りプログラム:** ジョブを入力装置またはデータベース・ファイルから取り出し、ジョブ待ち行列に入れる機能です。

バッチ・ジョブが読み取りプログラムによって入力源から読み取られると、入力ストリームの中のコマンドはジョブに対する要求としてシステムの中に保管され、インライン・データはインライン・データ・ファイルとしてスプールされ、ジョブの入力がジョブ待ち行列に入れられます。ジョブ情報は、ジョブ入力サブシステムによる処理に備えてジョブ待ち行列から選択されるまで、読み取りプログラムによってシステム内に保管されたままになります。

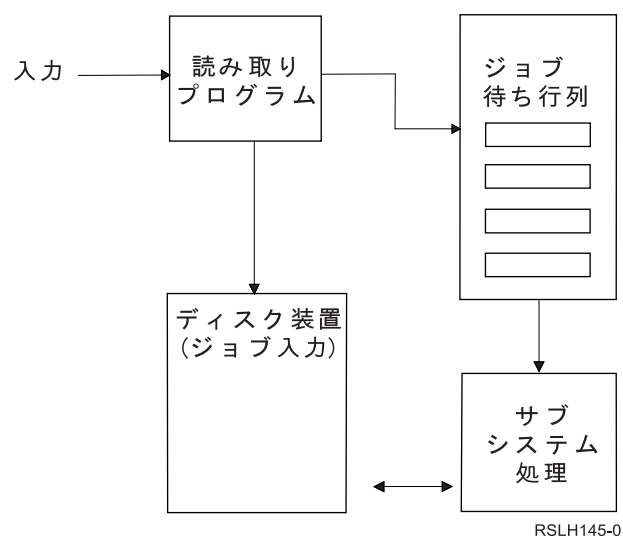


図1. 入カスプーリング要素の関係

読み取りプログラム機能を使用すれば、ディスク・ファイルまたはデータベース・ファイルから入力ストリームを読み取ることができます。

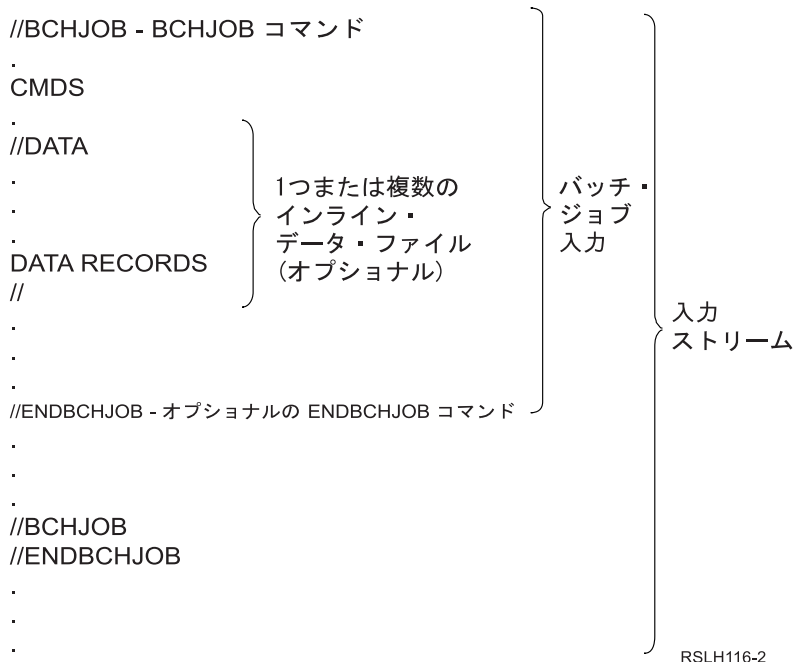


図2. 入力ストリームの典型的な編成

ジョブが入れられるジョブ待ち行列は、バッチ・ジョブ **BCHJOB** コマンド、データベース読み取りプログラム開始 **STRDBRDR** コマンド、またはジョブ記述の中の **JOBQ** パラメーターで指定されます。**BCHJOB** コマンドの **JOBQ** パラメーターの値は以下のとおりです。

- ***RDR**: ジョブ待ち行列は、**STRDBRDR** コマンドの **JOBQ** パラメーターから選択されます。
- ***JOBQ**: ジョブ待ち行列は、ジョブ記述の中の **JOBQ** パラメーターから選択されます。
- 特定のジョブ待ち行列: 指定された待ち行列が使用されます。

入力ストリームが小さいジョブの場合は、入力スプーリングを使用しないことによって、システム・パフォーマンスの向上を図ることができる場合があります。ジョブ実行依頼コマンド (**SBMJOB**) で入力ストリームを読み取り、該当するサブシステムの中のジョブ待ち行列にジョブを入れて、スプーリング・サブシステムおよび読み取りプログラムの操作をバイパスします。

ジョブが大きい入力ストリームの読み取りを必要とする場合は、入力スプーリング (ディスク読み取りプログラム開始 **STRDKTRDR** または **STRDBRDR** コマンド) を使用して、ジョブが実際に処理されるのとは独立してインポートできるようにしてください。

ジョブ入力コマンド

以下のコマンドを使用して、システムにジョブを実行依頼することができます。読み取りプログラム開始コマンドは、ジョブ入力のスプーリングに使用することができます。ジョブ実行依頼コマンドでは、スプーリングは使用しません。

- **バッチ・ジョブ (BCHJOB)**: バッチ入力ストリーム内のジョブの開始をマークし、ジョブの操作特性を定義します。
- **データ (DATA)**: インライン・データ・ファイルの開始をマークします。
- **バッチ・ジョブ終了 (ENDBCHJOB)**: バッチ入力ストリーム内のジョブの終了をマークします。
- **入力終了 (ENDINP)**: バッチ入力ストリームの終了をマークします。

- データベース・ジョブ実行依頼 (SBMDBJOB): 入力ストリームをデータベース・ファイルから読み取り、その入力ストリーム内のジョブを該当するジョブ待ち行列に入れます。
- ディスケット・ジョブ実行依頼 (SBMDKTJOB): ディスケットから入力ストリームを読み取り、その入力ストリーム内のジョブを該当するジョブ待ち行列に入れます。
- データベース読み取りプログラム開始 (STRDBRDR): 読み取りプログラムを開始して、データベース・ファイルから入力ストリームを読み取り、その入力ストリーム内のジョブを該当するジョブ待ち行列に入れます。
- ディスケット読み取りプログラム開始 (STRDKTRDR): 読み取りプログラムを開始して、ディスクから入力ストリームを読み取り、その入力ストリーム内のジョブを該当するジョブ待ち行列に入れます。

インライン・データ・ファイル

インライン・データ・ファイルは、読み取りプログラムまたはジョブ実行依頼コマンドによってジョブが読み取られる際に、バッチ・ジョブの一部として含まれるデータ・ファイルです。 SBMDBJOB または STRDBRDR を使用して CL バッチ・ストリーム (実行する CL コマンドのストリーム) を待ち行列に入れます。この CL バッチ・ストリームには、インライン・データ・ファイル (一時ファイル) に入れるデータを含めることができます。ジョブの終了時にインライン・データ・ファイルは削除されます。

インライン・データ・ファイルは、ジョブの中で、ファイルの開始を //DATA コマンドにより、ファイルの終了をデータ終了区切り文字によって、それぞれ区切られます。

データ終了区切り文字としては、ユーザー定義の文字ストリングまたはデフォルトの // を使用できます。// は 1 桁目および 2 桁目になければなりません。データの 1 桁目および 2 桁目に // が含まれている場合は、// *** END OF DATA のような固有の文字のセットを使用する必要があります。これを固有のデータ終了区切り文字として指定するには、//DATA コマンドの ENDCHAR パラメーターを次のようにコーディングする必要があります。

```
ENDCHAR('// *** END OF DATA')
```

注: インライン・データ・ファイルにアクセスすることができるのは、バッチ・ジョブの最初の経路指定ステップ中だけです。バッチ・ジョブにジョブ転送 (TFRJOB) コマンド、ジョブ経路再指定 (RRTJOB) コマンド、またはバッチ・ジョブ転送 (TFRBCHJOB) コマンドが入っている場合は、新しい経路指定ステップでインライン・データ・ファイルにアクセスすることはできません。

インライン・データ・ファイルは、名前の付いているものと名前のないもののどちらも可能です。名前のないインライン・データ・ファイルの場合は、QINLINE が //DATA コマンドの中でファイル名として指定されるか、名前は指定されないかどちらかです。名前付きインライン・データ・ファイルの場合は、ファイル名が指定されます。

名前付きインライン・データ・ファイルには以下の特性があります。

- ジョブの中で固有な名前を持ちます。他のインライン・データ・ファイルが同じ名前を持つことはありません。
- ジョブの中で複数回使用できます。
- オープンのたびに最初のレコードに位置合わせされます。

名前付きインライン・データ・ファイルを使用するには、プログラムでファイル名を指定するか、プログラムで指定されているファイル名を一時変更コマンドによってインライン・データ・ファイルの名前に変更する必要があります。ファイルは入力専用としてオープンされます。

名前のないインライン・データ・ファイルには以下の特性があります。

- 名前は QINLINE です。(1 つのバッチ・ジョブの中では、名前のないインライン・データ・ファイルには、すべて同一名が与えられます。)
- ジョブの中で使用できるのは 1 回だけです。
- 1 つのジョブの中に名前のないインライン・データ・ファイルが複数含まれるときは、それらのファイルはファイルのオープンと同じ順序で入力ストリームの中になければなりません。

名前のないインライン・データ・ファイルを使用するには、次のいずれかのようにします。

- プログラムで QINLINE を指定します。
- ファイル指定変更コマンドを使用することにより、プログラムの中で指定されているファイル名を QINLINE に変更します。

使用している高水準言語が 1 つのプログラム内で固有のファイル名を必要とする場合は、QINLINE をファイル名として使用できるのは 1 回だけです。名前のないインライン・データ・ファイルを複数個使用する必要がある場合は、ファイル指定変更コマンドをプログラムの中で使用して、その他の名前のないインライン・データ・ファイルに対して QINLINE を指定することができます。

注: コマンドを条件付きで実行し、名前のないインライン・データ・ファイルを複数処理する場合、名前のない間違ったインライン・データ・ファイルを使用すると、結果は予測できません。

インライン・データ・ファイルのオープンに関する考慮事項

インライン・データ・ファイルを開く際には、以下の要素に注意する必要があります。

- レコード長は入力レコードの長さを指定します。(レコード長はオプションです。)レコード長がデータの長さを超える場合、プログラムにメッセージが送られます。データにはブランクが埋め込まれます。レコード長がデータ長より短い場合、レコードは切り捨てられます。
- ファイルがプログラムの中で指定されると、システムはライブラリーの中でそのファイルを検索する前に、名前付きのインライン・データ・ファイルとしてそのファイルを検索します。したがって、名前付きのインライン・データ・ファイルが、インライン・データ・ファイルではないファイルと同じ名前の場合は、ファイル名がライブラリー名によって修飾されている場合でも、インライン・データ・ファイルが常に使用されます。
- 名前付きのインライン・データ・ファイルは、ファイル作成コマンドまたはファイル一時変更コマンドで SHARE(*YES) を指定することによって、同一ジョブの中においてプログラム間で共用することができます。例えば、INPUT という名前のファイルおよび SHARE(*YES) を指定するファイル一時変更コマンドが、INPUT という名前のインライン・データ・ファイルのあるバッチ・ジョブの中にある場合は、そのジョブの中でファイル名 INPUT を指定する実行中のプログラムがあれば、同じ名前付きインライン・データ・ファイルを共有することになります。名前のないインライン・データ・ファイルを同一ジョブの中のプログラム間で共用することはできません。
- インライン・データ・ファイルを使用する際は、正しいファイル・タイプが //DATA コマンドで指定されるようにしてください。たとえば、ファイルがソース・ファイルとして使用される場合は、//DATA コマンドでのファイル・タイプがソース・ファイルでなければなりません。
- インライン・データ・ファイルは入力専用としてオープンされます。

スプール・ファイルおよび出力待ち行列

スプーリング機能によって、スプール・ファイル (プリンター出力とも呼ばれる) は出力待ち行列に置かれます。これによって、印刷操作をより効率的に管理できるようになります。

スプーリングの概要

スプーリング機能は、システムによって実行され、出力を作成するプログラムによる特別な操作を必要としません。プログラムがプリンター・ファイルを開くと、オペレーティング・システムは、プリンター・ファイルの SPOOL パラメーターを調べて、出力をスプールするかどうかを決定します。

スプーリングを指定するプリンター・ファイルがオープンされると、プログラムの出力が入っているスプール・ファイル (印刷されるデータ) が、システムの中の該当する出力待ち行列に入れられます。スプール・ファイルは、プリンター・ファイルが開かれた時点、プリンター・ファイルが閉じられた時点、またはジョブが終了した時点で印刷用に使用できるようになります。スケジュール・パラメーターに特定の値を指定することによって、これを実行します。*IMMED は、プログラムが開かれるとすぐに、スプール・ファイルが書き出しプログラムに使用できるようにします。*FILEEND は、ファイルが閉じられるとすぐに、スプール・ファイルが書き出しプログラムに使用できるようにします。*JOBEND は、ジョブが完了するとすぐに、スプール・ファイルが書き出しプログラムに使用できるようにします。

このスプーリングのプロセスによって、プリンターの可用性および速度による潜在的なジョブの制限を回避することができます。つまり、システムは、プリンターが出力を印刷するよりかなり速く印刷出力を生成するアプリケーション・プログラムを処理できるようになります。

スプーリング (つまり、印刷を待機する出力待ち行列に出力を送信すること) によって、システムは、そのアプリケーション・プログラムの印刷が完了するのを待たずに、次のアプリケーション・プログラムの処理を開始できます。

スプーリングは、実行されているジョブの数が、使用できるプリンターの数を超えることがしばしばあるような複数ユーザー環境において特に重要です。スプーリングを使用することによって、ある出力待ち行列から別の出力待ち行列へ、またはあるプリンターから別のプリンターへ、出力を簡単にリダイレクトすることができます。

スプール・ファイル

スプーリングとは、後から処理や印刷を行うためにデータベース・ファイルにデータを保管するシステム機能です。保存され、最終的に印刷されるこのデータは、スプール・ファイル (またはプリンター出力ファイル) と呼ばれます。スプーリングを使用する場合、スプール・ファイルは、アプリケーション・プログラムやシステム・プログラムから、または Print キーを押すことによって作成されます。これらのファイルは、出力待ち行列と呼ばれる場所に入れられます。

印刷出力を生成するほとんどすべてのアプリケーション・プログラムが、i5/OS オペレーティング・システムが提供するスプーリング・サポートを利用します。プリンター・ファイルの SPOOL パラメーターの SPOOL(*YES) および SPOOL(*NO) の値によって、スプーリング・サポートを要求するかどうかが決まります。

表示画面のイメージを取り込む Print キーを使用すると、ほとんどの場合、スプール・ファイルが作成されます (ワークステーション装置記述で指定されたプリンター・ファイルで SPOOL = *YES が指定されている必要があります)。値が変更されていない限り、QSYSPRT プリンター・ファイルの SPOOL 属性のデフォルト値は *YES です。Print キーを押すと、システムは、QSYSPRT プリンター・ファイルの OUTQ パラメーターを調べて、どの出力待ち行列にスプール・ファイルを送信するかを判別します。

スプーリング (SPOOL = *YES) には、直接出力 (プリンター・ファイルにおける SPOOL = *NO) と比較して以下のようないくつかの利点があります。

- ユーザーのディスプレイ装置を引き続き作業に使用できる。
- プリンターが使用できる状態になるのを待つことなく、他のユーザーが印刷作業を要求できる。

- 特別な用紙が必要な場合、スプール・ファイルを特別な出力待ち行列に送信して、プリンターが使用中でない時点で印刷できる。
- ディスク操作はプリンターよりかなり高速であるため、システムが効率よく使用される。

出力待ち行列

出力待ち行列は、システムに定義されたオブジェクトであり、印刷されるまでスプール・ファイルが待機する場所を提供します。出力待ち行列はユーザーまたはシステムによって作成されます。

出力待ち行列作成 (CRTOUTQ) コマンドを使用して出力待ち行列を作成できます。プロンプト表示で、作成する出力待ち行列の名前を指定します。出力待ち行列は、ライブラリー・プロンプトによって示されるライブラリーに入れられます。必要な数だけの出力待ち行列を作成できます。

プリンターがシステムに対して構成されると、システムは、手動または自動構成によって、そのプリンターの出力待ち行列を QUSRSYS ライブラリーに作成します。システム作成の出力待ち行列は一般に装置出力待ち行列と呼ばれ、プリンターと同じ名前になります。例えば、「装置記述作成」(印刷装置) (CRTDEVPRT) コマンドを使用してプリンターを構成する場合、DEV D パラメーターでプリンター名 PRT01 を割り当てると、システムは、QUSRSYS ライブラリーに PRT01 という名前の出力待ち行列を作成します。

IBM 提供のシステムのデフォルト値が変更されていない場合は、システム値のデフォルト・プリンター (QPRTDEV) を表示することによって出力待ち行列を指定することができます。出力待ち行列は、システム・プリンターに表示される値と同じ名前になります。

アプリケーション・プログラムが実行されると、スプール・ファイルが作成されます。スプール・ファイルをすぐには印刷しない場合は、その時点でプリンターが割り当てられていない出力待ち行列にスプール・ファイルを送信することができます。例えば、使用できるプリンターが 1 つだけだとします。アプリケーション・プログラムの 1 つが 600 ページの印刷出力のジョブを作成します。すべてのユーザーが同じプリンターを使用するため、すべての人がその日の作業を終了するまでは、600 ページのジョブを印刷しないことにします。1 つの解決策として、2 つの別個の出力待ち行列を作成する方法があります。1 つの出力待ち行列は、600 ページの印刷出力を作成するアプリケーション・プログラムからスプール・ファイルを受け取ります。もう 1 つの出力待ち行列は、他のユーザーによって実行されるジョブのスプール・ファイルを受け取ります。

600 ページのジョブを作成するプログラムは、そのスプール・ファイルを特定の出力待ち行列に送信します。その出力待ち行列にはプリンターが割り当てられていません。したがって、600 ページのスプール・ファイルはプリンターが割り当てられるまで待つ必要があります。他方、もう一方の出力待ち行列内にあるスプール・ファイルは印刷することができます。複数の出力待ち行列は、据え置き印刷に使用することもできます。プリンターの出力待ち行列の現行制限を超える大きなスプール・ファイルを印刷するには、プリンターを、制限のない出力待ち行列に割り当てることができます。別の解決策として、指定する時間中に印刷する最大スプール・ファイル・サイズを設定する方法があります。例えば、100 ページの最大スプール・ファイル・サイズを、08:00:00 から 17:30:00 までの時間に設定することができます。この時間中には、100 ページ以下のスプール・ファイルのみが印刷されます。午後 5:30 以後は、どのスプール・ファイルも印刷されません。大きすぎるスプール・ファイルは、印刷ができるようになるまで、据え置き状況 (*DFR) に置かれます。据え置き印刷を構成する方法の詳細については、『スプール・ファイル・サイズによる印刷の制御』を参照してください。

複数の出力待ち行列

複数の出力待ち行列を作成したほうがよい理由として、以下のような場合があります。

- 特殊用紙での印刷
- 通常の作業時間後に印刷される出力
- 印刷されない出力

表示するかデータベース・ファイルにコピーすればよだけのスプール・ファイルを処理するために、出力待ち行列を作成することができます。不必要なスプール・ファイルを除去する際には注意が必要です。

- 特別な使用法

例えば、各プログラマーに別個の出力待ち行列を提供することができます。

- 特殊システム・ファイルの出力

以下のようなシステム提供ファイルでは、別個の待ち行列を作成したほうがよい場合があります。

- **QPJOBLOG:** すべてのジョブ・ログを別個の待ち行列に送信したほうがよい場合があります。
- **QPPGMDMP:** すべてのプログラム・ダンプを別個の待ち行列に送信して、必要に応じて、それらを検討または印刷したり、毎日消去したりするほうがよい場合があります。
- **QPSRVDMP:** すべてのサービス・ダンプを別個の待ち行列に送信して、必要に応じて、サービス担当員がそれらを検討できるようにしたほうがよい場合があります。

出力待ち行列のリカバリー

スプール・ファイルを作成したジョブが、そのジョブやシステムが異常停止したときに実行中の場合、これらのファイルは出力待ち行列に残ります。アクティブ・プログラムによって書き込まれた一定数のレコードは、ジョブが終了し、失われた場合にも、主ストレージに残る場合があります。これらのスプール・ファイルを検査して、これらのファイルを引き続き使用する前に、これらのファイルが完全であることを確認する必要があります。

ジョブによって作成されたすべてのスプール・ファイル (QPJOBLOG を除く) を、印刷装置書き出しプログラムによる通常の処理用に保持するかどうか、またはこれらのファイルを削除するかどうかを指定することができます。

異常終了が発生すると、スプール・ファイル QPJOBLOG は、システムの次の IPL の際に書き込まれません。

スプール・ファイルの印刷中に印刷装置書出プログラムが失敗すると、スプール・ファイルは出力待ち行列にそのまま残ります。

ユーザー作成の出力待ち行列のリカバリー

出力待ち行列が損傷を受けて使用できない状態になると、ユーザーは、システム・オペレーター・メッセージ待ち行列に送信されるメッセージによって通知を受けます。印刷装置書出プログラムやジョブが、スプール・ファイルを、損傷を受けた待ち行列に追加したり、そこから除去したりしようとする、システム機能からメッセージが出されます。

損傷を受けた出力待ち行列は、手動で削除することができます。そうしない場合は、次の IPL 中にシステムによって削除されます。

1 損傷を受けた出力待ち行列を削除すると、損傷を受けた出力待ち行列上のすべてのスプール・ファイルは、
1 ライブラリー QRCL 内の出力待ち行列 QSPRCLOUTQ に移動します。移動操作は QSPRC00001 システ
1 ム・ジョブによって実行されます。このシステム・ジョブは、すべてのスプール・ファイルが移動すると、
1 QSYSOPR メッセージ待ち行列に完了メッセージを発行します。

1 出力待ち行列が独立 ASP 内にある場合、スプール・ファイルはライブラリー QRCLxxxxx 内の出力待ち行
1 列 QSPRCLOUTQ に移動します。ここで、xxxxx はプライマリー独立 ASP の独立 ASP 番号です (例え
1 ば、プライマリー独立 ASP 番号が 33 の場合は QRCL00033)。移動操作は QSPRCxxxxx ジョブによっ
1 て実行されます (xxxxx はプライマリー独立 ASP の独立 ASP 番号)。すべてのスプール・ファイルが移動す
1 ると、メッセージが QSYSOPR メッセージ待ち行列に送信されます。

損傷を受けた出力待ち行列は、削除した後に再び作成することができます。次に、出力待ち行列
QSPRCLOUTQ 上のスプール・ファイルを、新しく作成した出力待ち行列に移動することができます。

システム作成の出力待ち行列のリカバリー

損傷を受けた出力待ち行列が、プリンターに関連したデフォルトの出力待ち行列の場合、システムは、出力
待ち行列が削除されると自動的にそれを再作成します。

このシステム作成の出力待ち行列の PUBLIC 権限は装置に指定されたものと同じであり、その他のパラメ
ーターはデフォルト値です。システムが出力待ち行列を再作成した後、その属性が正しいことを検査する
か、必要に応じてそれらを変更する必要があります。

1 プリンターに関連した損傷を受けた出力待ち行列が削除され、再び作成されると、損傷を受けた出力待ち行
1 列上のすべてのスプール・ファイルは、再作成された出力待ち行列に移動します。これは QSPRC00001 シ
1 ステム・ジョブによって実行されます。このシステム・ジョブは、すべてのスプール・ファイルが移動する
1 と、QSYSOPR メッセージ待ち行列に完了メッセージを発行します。

スプール・ファイルのクリーンアップ

1 システムのスプール・ファイル・リカバリーは、初期プログラム・ロード (IPL) の直後に開始します。ス
1 プール・ファイル・リカバリーはシステム・ジョブ QSPRC00001 の下に実行されます。破壊されたユーザ
1 ー作成出力待ち行列のスプール・ファイルは、ライブラリー QRCL の出力待ち行列 QSPRCLOUTQ に移
1 動します。破壊されたシステム作成出力待ち行列のスプール・ファイルは、再作成された出力待ち行列に移
1 動します。

1 スプール・ファイル・リカバリーは、独立 ASP グループをオンに変更する操作直後にも開始します。スプ
1 ール・ファイル・リカバリーはシステム・ジョブ QSPRCxxxxx の下に実行されます。ここで、xxxxx は、
1 プライマリー独立 ASP の独立 ASP 番号です。破壊されたユーザー作成出力待ち行列のスプール・ファイ
1 ルは、ライブラリー QRCLxxxxx の出力待ち行列 QSPRCLOUTQ に移動します。ここで、xxxxx は、プラ
1 イマリー独立 ASP の独立 ASP 番号です。

デフォルトの出力待ち行列

プリンターがシステムに対して構成されている場合、システムは、自動的にライブラリー QUSRSYS に、
プリンターのデフォルトの出力待ち行列を作成します。出力待ち行列には、'Default output queue for
printer xxxxxxxxxxx' というテキスト記述が与えられます (xxxxxxxxxxx は構成時にプリンターに割り当て
られた名前)。プリンター名は装置記述 (DEV D) パラメーターで指定されます。

出力待ち行列の AUT パラメーターには、プリンター装置記述の AUT パラメーターで指定された値と同
じ値が割り当てられます。他のすべてのパラメーターには、デフォルト値が割り当てられます。

CRTOUTQ コマンドで出力待ち行列を作成する際に使用するデフォルト値を変更するには、「コマンド省略時値変更」(CHGCMDDFT) コマンドを使用します。

プリンターのデフォルト出力待ち行列は、プリンター装置記述を作成したユーザーによって所有されます。自動構成の場合は、プリンターと出力待ち行列の両方ともシステム・プロファイル QPGMR によって所有されます。

システムは、スプールされたすべての出力のデフォルト出力待ち行列として、システム・プリンターのデフォルト出力待ち行列を使用するよう、デフォルト値が設定されて出荷されます。システム・プリンターは、デフォルト・プリンター (QPRTDEV) のシステム値によって定義されます。

スプール・ファイルがプリンター・ファイルを開くことによって作成され、そのファイルについて指定されている出力待ち行列が見つからない場合、システムはそのスプール・ファイルをライブラリー QGPL の中の出力待ち行列 QPRINT に入れようと試みます。何らかの理由により、スプール・ファイルが出力待ち行列 QPRINT に入れられない場合はエラー・メッセージが送られ、出力はスプールされません。

以下の出力待ち行列がシステムによって提供されます。

| 出力待ち行列 | 説明 |
|---------|--------------------|
| QPRINT | デフォルト・プリンター出力待ち行列 |
| QPRINTS | 特殊用紙用のプリンター出力待ち行列 |
| QPRINT2 | 2 部用紙用のプリンター出力待ち行列 |

出力待ち行列上のスプール・ファイルの順序

出力待ち行列上のスプール・ファイルの順序は、主に、スプール・ファイルの状況によって決まります。

書き出しプログラムによって処理されているスプール・ファイルは、印刷中 (PRT 状況)、書き出しプログラム (WTR 状況)、印刷の保留中 (PND 状況)、送信中 (SND 状況) の状況になる場合があります。PRT、WTR、PND、または SND の状況を持つスプール・ファイルは、出力待ち行列の上部に置かれます。ユーザーがスプール・ファイルを保留にしたが、書き出しプログラムがまだファイルの処理を終了していない場合、書き出しプログラムによって処理されているスプール・ファイルは、保留 (HLD) 状況になる場合があります。RDY の状況を持つ他のすべてのスプール・ファイルは、出力待ち行列上で、書き出しプログラムによって処理中のファイルの後にリストされます。この後に、据え置きスプール・ファイル (DFR 状況)、さらにその後に RDY や DFR 以外の状況を持つスプール・ファイルがリストされます。

スプール・ファイルの各グループ (RDY および非 RDY ファイル) は、さらに以下の項目によってソートされます。

1. スプール・ファイルの出力優先順位
2. 日時フィールド (タイム・スタンプ)
3. スプール・ファイルの SCHEDULE パラメーター値 SCHEDULE(*JOBEND) が指定されたファイルは一緒のグループにされ、同じジョブの SCHEDULE(*IMMED) または SCHEDULE(*FILEEND) が指定された他のスプール・ファイルの後に置かれます。
4. スプール・ファイルのスプール番号。

SEQ(*JOBNBR) が指定された出力待ち行列の場合、日時フィールドは、スプール・ファイルを作成したジョブがシステムに入った日時です。(ジョブがシステムに入ると、ジョブ通し番号と時刻の値もジョブに割り当てられます。) スプール・ファイルはこのようにして待ち行列上でソートされます。

先入れ先出し (*FIFO) 出力待ち行列では、日時は、次のような場合、システムの現在の日時に変更されません。

- 装置ファイルを開くことによってスプール・ファイルが作成される。
- スプール・ファイルを作成したジョブの出力優先順位が変更される。
- スプール・ファイルの状況が非 RDY から RDY に変更される。

注: 状況が RDY から WTR または WTR から RDY に変更された理由が、書き出しプログラムが取り消されたからである場合、日時は変更されません。また、状況が RDY から DFR、または DFR から RDY に変更された場合にも、日時は変更されません。

- スプール・ファイルは、SEQ(*FIFO) が指定された別の出力待ち行列に移動します。

出力待ち行列に SEQ(*FIFO) を指定せずに、SEQ(*JOBNBR) を指定する場合、スプール・ファイルの自動ソートによって、結果は異なります。例えば、あるスプール・ファイルが保留にされており、SEQ(*JOBNBR) が指定されている出力待ち行列ですぐに保留が解除される場合、このスプール・ファイルは、それが開始された場所で終了します。しかし、同じスプール・ファイルが保留にされており、SEQ(*FIFO) が指定されている出力待ち行列ですぐに保留が解除される場合、このスプール・ファイルは、同じ優先順位と RDY の状況を持つスプール・ファイルの最後に配置されます。

データ待ち行列のサポート

スプール・ファイルでは、2 つの異なるタイプのデータ待ち行列がサポートされます。

• 出力待ち行列でのデータ待ち行列のサポート

オプションとして、「出力待ち行列作成」(CRTOUTQ) コマンドまたは「出力待ち行列変更」(CHGOUTQ) コマンドを使用して、データ待ち行列を出力待ち行列に関連付けるサポートを利用できます。スプール・ファイルが出力待ち行列で作動可能 (RDY) 状況になると、項目がデータ待ち行列に記録されます。ユーザー・プログラムで、Receive Data Queue (QRCVDTAQ) API を使用してデータ待ち行列から情報を受け取ることによって、出力待ち行列でスプール・ファイルを使用できる時期を判断できます。詳しくは、「プログラミング」トピックの Receive Data Queue (QRCVDTAQ) API を参照してください。

出力待ち行列のスプール・ファイルが RDY 状況になるたびに、項目がデータ待ち行列に送信されます。スプール・ファイルは、出力待ち行列から取り除かれるまでに、複数の状況に変わることができません (例えば、作動可能 (RDY) から保留 (HLD)、保留解除 (RLS)、再び作動可能 (RDY) へなど)。スプール・ファイルが RDY 状況になるたびに、これらの状況変更の結果として、スプール・ファイルのデータ待ち行列に項目が記録されることになります。

スプール・ファイルは以下の場合に RDY 状況になります。

- 最初に出力待ち行列でスプールされる場合。
- スプール・ファイルが開かれる際に、スケジュール・パラメーター値が *IMMED の場合。
- ジョブが完了する際に、スプール・ファイル・スケジュール・パラメーター値が *JOBEND の場合。
- スプール・ファイルの保留が解除される場合。
- スプール・ファイルが別の出力待ち行列からこの出力待ち行列に移動する場合。
- スプール・ファイルの印刷中に書き出しプログラムが直ちに終了する場合 (スプール・ファイルの状況は WTR から RDY にリセットされます)。

データ待ち行列は、128 バイト以上の最大メッセージ長 (MAXLEN) パラメーター値で作成する必要があります。順序 (SEQ) パラメーター値は *FIFO または *LIFO でなければなりません。CRTDTAQ コマンドのフォーマットは次のとおりです。

```
CRTDTAQ DTAQ (<library name>/<data queue name>) MAXLEN(128) SEQ(*LIFO)
```

「出力待ち行列作成」(CRTOUTQ) コマンドおよび「出力待ち行列変更」(CHGOUTQ) コマンドには、データ名の指定に使用する、データ待ち行列 (DTAQ) パラメーターがあります。指定したデータ待ち行列が存在しない場合、または出力待ち行列を作成または変更するユーザーにデータ待ち行列の使用権限がない場合、これらのコマンドを使用するとエラーが発生します。

データ待ち行列を出力待ち行列に関連付けると、出力待ち行列に置かれたスプール・ファイルが作動可能状況になる場合は、データ待ち行列に項目が置かれるようになります。スプール・ファイルを生成するユーザーがデータ待ち行列に対してどのような権限を持つかに関わりなく、データ待ち行列項目が追加されます。

データ待ち行列の項目はレコード・タイプ 01 のフォーマットを持ちます。レコード・タイプ 01 に含まれる情報の説明については、『レコード・タイプ 01 のデータ待ち行列項目フォーマット』を参照してください。

• 環境変数データ待ち行列のサポート

「環境変数追加」(ADDENVVAR) コマンドまたは「環境変数変更」(CHGENVVAR) コマンドを使用して、データ待ち行列をジョブまたはシステムに関連付けることができます。スプール・ファイルを作成すると、項目がデータ待ち行列に記録されます。ユーザー・プログラムで Receive Data Queue (QRCVDTAQ) API を使用してデータ待ち行列から情報を受け取ることによって、スプール・ファイルがジョブまたはシステムによっていつ作成されたかを判別できます。QPRTJOB の下に保管され、リモート・コマンド・システム・ジョブなどのジョブによって作成されたスプール・ファイルの ID を判別する必要がある場合は、環境変数データ待ち行列サポートを使用します。

CL コマンドの ADDENVVAR を使用して、環境変数 QIBM_NOTIFY_CRTPSPLF に完全修飾データ待ち行列名を指定することによって、データ待ち行列をジョブまたはシステムに関連付けることができます。

コマンドは次のように使用します。

```
ADDENVVAR ENVVAR(QIBM_NOTIFY_CRTPSPLF)
          VALUE('*DTAQ <library name>/<data queue name>')
          LEVEL(*JOB | *sys)
```

データ待ち行列は、144 バイト以上のレコード長で作成する必要があります。また、データ待ち行列は *USE という共通権限を持つ必要があります。あるいは、ユーザーが、QSPL ユーザー・プロファイルの *USE という専用権限をデータ待ち行列に与える必要があります。収容ライブラリーが *EXECUTE という共通権限を持つようにするか、QSPL ユーザー・プロファイルの *EXECUTE という専用権限をライブラリーに与える必要があります。CRTDTAQ コマンドのフォーマットは次のとおりです。

```
CRTDTAQ DTAQ (<library name>/<data queue name>) MAXLEN(144) AUT(*USE)
```

| データ待ち行列情報の日時を協定世界時 (UTC) にする場合は、値の異なる同じ環境変数を使用できま
| す。上記の ADDENVVAR コマンドで *DTAQ を *DTA2 に置き換えます。この値により、データ待ち
| 行列に、タイプ 02 のデータ待ち行列項目の代わりに、タイプ 03 の項目が置かれるようになります。
| コマンド・フォーマットは次のとおりです。

```
| ADDENVVAR ENVVAR(QIBM_NOTIFY_CRTPSPLF)
|          VALUE('*DTA2 <library name>/<data queue name>')
|          LEVEL(*JOB | *sys)
```

- | タイプ 03 のデータ待ち行列項目の長さは 200 バイトです。次のコマンド・フォーマットを使用して、
- | タイプ 03 のデータ待ち行列を作成します。
- | CRTDTAQ DTAQ (<library name>/<data queue name>) MAXLEN(200) AUT(*USE)

データ待ち行列をジョブまたはシステムに関連付けると、ジョブまたはシステムによって作成されたスプール・ファイルは、自動的に項目がデータ待ち行列に入れられるようになります。この操作が実行されるようにするには、ユーザーまたはユーザー・プロファイル QSPL にデータ待ち行列に対する権限が必要です。

注: ジョブ・レベルで指定した環境変数は、システム・レベルで指定した同一の環境変数より優先されません。

エラー条件

指定したデータ待ち行列が存在しない場合、または出力待ち行列を作成または変更するユーザーにデータ待ち行列の使用権限がない場合にエラーが発生します。

データ待ち行列を出力待ち行列に関連付けると、出力待ち行列に置かれたスプール・ファイルが作動可能状況になる場合は、データ待ち行列に項目が置かれるようになります。スプール・ファイルを生成するユーザーがデータ待ち行列に対してどのような権限を持つかに関わりなく、データ待ち行列項目が追加されます。

存在しない、または長さが無効なデータ待ち行列にシステムが項目を追加しようとすると、システムの処理は継続されますが、QSYSOPR メッセージ待ち行列に通知メッセージが送信されます。このメッセージはデータ待ち行列に問題があることを示しており、そのデータ待ち行列の名前が示されます。このメッセージは、出力待ち行列のデータ待ち行列に特定の問題が初めて発生したときに送信されます。メッセージは 24 時間おきに送信されます。

例えば、メッセージ X が午前 10:00 に受信されると、このメッセージは QSYSOPR メッセージ待ち行列に記録されます。メッセージ X が再び午前 10:30、午前 11:00、午後 1:00、午後 1:30 に受信されても、このメッセージは記録されません。これから分かるように、メッセージは、一日中、継続して受信されても、次の日の午前 10:00 過ぎまでは記録されません。

メッセージ X が午前 10:00 に記録された後に、メッセージ Y が午後 2:00 に受信される場合、メッセージ Y は記録されます。メッセージ X が再び午後 2:30 に受信されると、メッセージ X は、この日の以前の時間に記録されている場合でも再び記録されます。

その意図は、繰り返される同一メッセージを一日中ログ記録することではなく、特定の出力待ち行列のデータ待ち行列に関連したエラー・メッセージが変更されるたびに、そのことをユーザーに知らせることです。

追加の考慮事項

出力待ち行列にスプール・ファイルが存在するかどうかに関係なく、出力待ち行列のデータ待ち行列を変更することができます。レコード・タイプ 01 のデータ待ち行列項目の場合、データ待ち行列上に項目があるのは変更の後に RDY 状況になるスプール・ファイルだけです。出力待ち行列で既に作動可能状況にあるスプール・ファイルについては、新しいデータ待ち行列に項目がありません。

データ待ち行列を管理するのはユーザーの責任です。これらの責任には、データ待ち行列の作成、クリア、および削除が含まれます。

IPL 中にすべての出力待ち行列をクリアする場合、関連するデータ待ち行列はクリアされません。損傷を受けたシステム出力待ち行列が見つかったら、関連するデータ待ち行列名なしに再作成されます。損傷を受けたデータ待ち行列は再作成されません。

レコード・タイプ 01 のデータ待ち行列項目フォーマット

レコード・タイプ 01 のデータ待ち行列項目フォーマットの表は、スプール・ファイルが出力待ち行列で作動可能な状況に変わるときの、01 のデータ待ち行列項目のフォーマットを示しています。

表 1. レコード・タイプ 01 のデータ待ち行列項目フォーマット

| 10 進オフセット | 16 進オフセット | タイプ | 説明 |
|-----------|-----------|-----------|---|
| 0 | 0 | CHAR(10) | 関数 データ待ち行列項目を作成した関数を示します。スプール・ファイルの値は *SPOOL です。 |
| 10 | A | CHAR(2) | レコード・タイプ 関数内のレコード・タイプを示します。有効な値は次のとおりです。 01 READY 状況のスプール・ファイルは出力待ち行列に置かれています。 |
| 12 | C | CHAR(26) | 修飾ジョブ名 出力待ち行列に置かれたスプール・ファイルを作成したジョブの修飾ジョブ名を示します。 CHAR(10) ジョブ名 CHAR(10) ユーザー名 CHAR(6) ジョブ番号 |
| 38 | 26 | CHAR(10) | スプール・ファイル名 出力待ち行列に置かれたスプール・ファイルの名前を示します。 |
| 48 | 30 | BINARY(4) | スプール・ファイル番号 出力待ち行列に入れられたスプール・ファイルの固有の番号を示します。 |
| 52 | 34 | CHAR(20) | 修飾出力待ち行列の名前 スプール・ファイルが入れられた出力待ち行列の修飾名を示します。 CHAR(10) 出力待ち行列の名前 CHAR(10) 出力待ち行列のライブラリー |
| 72 | 48 | CHAR(8) | ジョブ・システムの名前 スプール・ファイルが生成されたシステムの名前を示します。 |

表1. レコード・タイプ 01 のデータ待ち行列項目フォーマット (続き)

| 10 進オフセット | 16 進オフセット | タイプ | 説明 |
|-----------|-----------|----------|---|
| 80 | 50 | CHAR(7) | スプール・ファイル作成日付 CYYMMDD フォーマットでスプール・ファイルが作成された日付を示します (ローカル・システム時刻)。 |
| 87 | 57 | CHAR(1) | Reserved |
| 88 | 58 | CHAR(6) | スプール・ファイル作成時刻 HHMMSS フォーマットでスプール・ファイルが作成された時刻を示します (ローカル・システム時刻)。 |
| 94 | 5E | CHAR(7) | スプール・ファイル作成日付 (UTC) CYYMMDD フォーマットでスプール・ファイルが作成された日付を示します。 |
| 101 | 65 | CHAR(1) | Reserved |
| 102 | 66 | CHAR(6) | スプール・ファイル作成時刻 (UTC) HHMMSS フォーマットでスプール・ファイルが作成された時刻を示します。 |
| 108 | 6C | CHAR(20) | Reserved |

レコード・タイプ 02 のデータ待ち行列項目フォーマット

レコード・タイプ 02 のデータ待ち行列項目フォーマットの表は、スプール・ファイルを作成する際の DTAQ 項目のフォーマットを示しています。

表2. レコード・タイプ 02 のデータ待ち行列項目フォーマット

| 10 進オフセット | 16 進オフセット | タイプ | 説明 |
|-----------|-----------|----------|--|
| 0 | 0 | CHAR(10) | 関数 データ待ち行列項目を作成した関数を示します。スプール・ファイルの値は *SPOOL です。 |
| 10 | A | CHAR(2) | レコード・タイプ 関数内のレコード・タイプを示します。有効な値は次のとおりです。 02 スプール・ファイルが作成されており、出力待ち行列に置かれています。 |

表2. レコード・タイプ 02 のデータ待ち行列項目フォーマット (続き)

| 10 進オフセット | 16 進オフセット | タイプ | 説明 |
|-----------|-----------|-----------|---|
| 12 | C | CHAR(26) | 修飾ジョブ名 出力待ち行列に置かれたスプール・ファイル を所有するジョブの修飾ジョブ名を示します。 CHAR(10) ジョブ名 CHAR(10) ユーザー名 CHAR(6) ジョブ番号 |
| 38 | 26 | CHAR(10) | スプール・ファイル名 出力待ち行列に置かれたスプール・ファイルの 名前を示します。 |
| 48 | 30 | BINARY(4) | スプール・ファイル番号 出力待ち行列に入れられたスプール・ファイル の固有の番号を示します。 |
| 52 | 34 | CHAR(20) | 修飾出力待ち行列の名前 スプール・ファイルが入れられた出力待ち行列 の修飾名を示します。 CHAR(10) 出力待ち行列の名前 CHAR(10) 出力待ち行列のライブラリー |
| 72 | 48 | CHAR(26) | 修飾ジョブ名の作成 スプール・ファイルを作成したジョブの修飾ジ ョブ名を示します。 CHAR(10) ジョブ名 CHAR(10) ユーザー名 CHAR(6) ジョブ番号 |
| 98 | 62 | CHAR(10) | ユーザー・データ 作成されたスプール・ファイルのユーザー指定 データを示します。 |
| 108 | 6C | BINARY(4) | 補助記憶域プール スプール・ファイルが作成された補助記憶域プ ールを示します。 |

表2. レコード・タイプ 02 のデータ待ち行列項目フォーマット (続き)

| 10 進オフセット | 16 進オフセット | タイプ | 説明 |
|-----------|-----------|----------|--|
| 112 | 70 | CHAR(8) | スレッド ID スプール・ファイルを作成したジョブのスレッドを示します。 |
| 120 | 78 | CHAR(10) | システムの名前 スプール・ファイルが生成されたシステムの名前を示します。 |
| 130 | 82 | CHAR(7) | 作成日付 CYYMMDD フォーマットでスプール・ファイルが作成された日付を示します (ローカル・システム時刻)。 |
| 137 | 89 | CHAR(6) | 作成時刻 HHMMSS フォーマットでスプール・ファイルが作成された時刻を示します (ローカル・システム時刻)。 |
| 143 | 8F | CHAR(1) | Reserved |

レコード・タイプ 03 のデータ待ち行列項目フォーマット

レコード・タイプ 03 のデータ待ち行列項目フォーマットの表は、スプール・ファイルを作成する際の DTA2 項目のフォーマットを示しています。

表3. レコード・タイプ 03 のデータ待ち行列項目フォーマット

| 10 進オフセット | 16 進オフセット | タイプ | 説明 |
|-----------|-----------|----------|---|
| 0 | 0 | CHAR(10) | 関数 データ待ち行列項目を作成した関数を示します。スプール・ファイルの値は *SPOOL です。 |
| 10 | A | CHAR(2) | レコード・タイプ 関数内のレコード・タイプを示します。有効な値は次のとおりです。 03 スプール・ファイルが作成されており、出力待ち行列に置かれています。 |
| 12 | C | CHAR(26) | 修飾ジョブ名 出力待ち行列に置かれたスプール・ファイルを所有するジョブの修飾ジョブ名を示します。 CHAR(10) ジョブ名 CHAR(10) ユーザー名 CHAR(6) ジョブ番号 |

表 3. レコード・タイプ 03 のデータ待ち行列項目フォーマット (続き)

| 10 進オフセット | 16 進オフセット | タイプ | 説明 |
|-----------|-----------|-----------|---|
| 38 | 26 | CHAR(10) | スプール・ファイル名 出力待ち行列に置かれたスプール・ファイルの名前を示します。 |
| 48 | 30 | BINARY(4) | スプール・ファイル番号 出力待ち行列に入れられたスプール・ファイルの固有の番号を示します。 |
| 52 | 34 | CHAR(20) | 修飾出力待ち行列の名前 スプール・ファイルが入れられた出力待ち行列の修飾名を示します。 CHAR(10) 出力待ち行列の名前 CHAR(10) 出力待ち行列のライブラリー |
| 72 | 48 | CHAR(26) | 修飾ジョブ名の作成 スプール・ファイルを作成したジョブの修飾ジョブ名を示します。 CHAR(10) ジョブ名 CHAR(10) ユーザー名 CHAR(6) ジョブ番号 |
| 98 | 62 | CHAR(10) | ユーザー・データ 作成されたスプール・ファイルのユーザー指定データを示します。 |
| 108 | 6C | BINARY(4) | 補助記憶域プール スプール・ファイルが作成された補助記憶域プールを示します。 |
| 112 | 70 | CHAR(8) | スレッド ID スプール・ファイルを作成したジョブのスレッドを示します。 |
| 120 | 78 | CHAR(10) | システムの名前 スプール・ファイルが生成されたシステムの名前を示します。 |
| 130 | 82 | CHAR(7) | 作成日付 (UTC) CYMMDD フォーマットでスプール・ファイルが作成された日付 (UTC) を示します。 |

表 3. レコード・タイプ 03 のデータ待ち行列項目フォーマット (続き)

| 10 進オフセット | 16 進オフセット | タイプ | 説明 |
|-----------|-----------|----------|---|
| 137 | 89 | CHAR(6) | 作成時刻 (UTC) HHMMSS フォーマットでスプール・ファイルが作成された時刻を示します。 |
| 143 | 8F | CHAR(57) | Reserved |

スプール・ファイル名

スプール・ファイルを作成する場合、スプール・ファイル名は、通常、このファイルの作成に使用したプリンター・ファイルと同じ名前になります。例えば、Print キーを押すと、スプール・ファイルには QSYSPRT という名前が付けられます。これは、Print キー操作によって使用されるプリンター・ファイルが QSYSPRT だからです。

スプール・ファイルが別の名前を持つようにするには、以下のいくつかの方法があります。

- 「印刷装置ファイル一時変更」(OVRPRTF) コマンドを使用し、SPLFNAME パラメーターで名前を指定する。例えば、次のコマンドをタイプ入力すると、

```
OVRPRTF QSYSPRT SPLFNAME(REPORT1)
```

スプール・ファイルの名前は QSYSPRT の代わりに REPORT1 になります。

- OVRPRTF コマンドを使用し、TOFILE パラメーターで異なるプリンター・ファイルを指定する。例えば、次のコマンドをタイプ入力すると、

```
OVRPRTF QSYSPRT TOFILE(PRTF2)
```

スプール・ファイルには PRTF2 という名前が付き (OVRPRTF コマンドの TOFILE パラメーターで指定されたプリンター・ファイルの名前)。

- IBM® アプリケーションの中には、そのファイルの作成に使用されたプリンター・ファイルと異なる名前を持つスプール・ファイルを作成するものがあります。このような状況でユーザーは、スプール・ファイルの名前を制御することはできません。

スプール・ファイルのセキュリティー

スプールのセキュリティーは、主に、スプール・ファイルの入った出力待ち行列によって制御されます。

一般に、ユーザーがスプール・ファイルを制御する (例えば、スプール・ファイルの保留または解放など) 権限を与えられるには以下の 4 とおりの方法があります。

- ユーザーが、ユーザー・プロファイルでスプール制御権 (SPCAUT(*SPLCTL)) を割り当てられる。

この権限は、ユーザーが *EXECUTE 権限を持つすべてのライブラリーの出力待ち行列内のすべてのスプール・ファイルの制御権をユーザーに与えます。この権限は、適切なユーザーに対してのみ与えられる必要があります。

- ユーザーが、ユーザー・プロファイルでジョブ制御権限 (SPCAUT(*JOBCTL)) を割り当てられ、出力待ち行列がオペレーターによって制御され (OPRCTL(*YES))、ユーザーが、出力待ち行列が入っているライブラリーに対して *EXECUTE 権限を付与されている。
- ユーザーが、出力待ち行列の必要なオブジェクト権限を付与されている。必要なオブジェクト権限は、CRTOUTQ コマンドの AUTCHK パラメーターによって指定されます。*OWNER の値は、出力待ち行列の所有者のみが、出力待ち行列のすべてのスプール・ファイルを制御する権限を与えられることを示

します。 *DTAAUT の値は、出力待ち行列に対する *CHANGE 権限を持つユーザーが、出力待ち行列のすべてのスプール・ファイルを制御する権限を与えられることを示します。

注: *DTAAUT に必要な特定権限は *READ、*ADD、および *DLT データ権限です。

- ユーザーが、常に、そのユーザーによって作成されたスプール・ファイルの制御を許可される。

スプール・ファイル・コピー (CPYSPLF)、スプール・ファイル表示 (DSPSPLF)、およびネットワーク・スプール・ファイル送信 (SNDNETSPLF) コマンドの場合、既にリストした 4 とおりの方法に加えて、ユーザーに権限を与えることのできる追加の方法があります。

出力待ち行列が作成されたときに DSPDTA(*YES) が指定された場合、出力待ち行列に対する *USE 権限を付与されたユーザーには、スプール・ファイルのコピー、表示、送信、または移動の許可が与えられます。必要な特定権限は *READ データ権限です。

ユーザーが、上記の 4 とおりの方法のいずれかによってファイルを制御する権限を与られている場合、出力待ち行列を作成するときに DSPDTA(*NO) を使用しても、ユーザーがファイルを表示、コピー、または送信することを制限することはできません。 DSPDTA 権限がチェックされるのは、それ以外ではファイルに対する権限がユーザーに与えられていない場合のみです。

DSPDTA(*OWNER) は DSPDTA(*NO) より制限が厳しくなります。 DSPDTA(*OWNER) を指定して出力待ち行列を作成する場合、スプール・ファイルの所有者 (ファイルを作成したユーザー) または SPCAUT(*SPLCTL) を持ったユーザーのみが、その待ち行列のファイルを表示、コピー、または送信することができます。オペレーターによって制御された (OPRCTL(*YES)) 出力待ち行列に対する SPCAUT(*JOBCTL) を付与されたユーザーでも、所有していないスプール・ファイルの表示、コピー、移動、または送信は行うことができません。

個別コマンドの権限の要件の詳細については、『セキュリティ』のトピックを参照してください。

スプール・ファイルを出力待ち行列に入れるには、以下のいずれかの権限が必要です。

- ユーザー・プロファイルにおけるスプール制御権 (SPCAUT(*SPLCTL))。ユーザーには、出力待ち行列が入っているライブラリーに対する *EXECUTE 権限も必要です。

この権限は、システム上のすべてのスプール・ファイルの制御権をユーザーに与え、適切なユーザーにのみ与えられる必要があります。スプール制御権がある場合は、システム上の任意のスプール・ファイルを削除、移動、保留、および解放することができます。任意のスプール・ファイルの属性を変更することもできます。

- ユーザー・プロファイルにおけるジョブ制御権限 (SPCAUT(*JOBCTL))。および出力待ち行列がオペレーターに制御されている (OPRCTL(*YES))。ユーザーには、出力待ち行列が入っているライブラリーに対する *EXECUTE 権限も必要です。
- 出力待ち行列に対する *READ 権限。 CRTOUTQ コマンドで AUT(*USE) を指定することによって、この権限をパブリックに与えることができます。

出力待ち行列のセキュリティ

出力待ち行列は、出力待ち行列作成 (CRTOUTQ) コマンドの AUT パラメーターの値によって決まるセキュリティのレベルで作成されます。

その出力待ち行列上のスプール・ファイルを処理するには、その出力待ち行列に対する適切な権限を持っている必要があります (AUT パラメーターで指定されます)。例えば、スプール・ファイルの保留または保留

解除では、1つのレベルの権限が必要になる場合がありますが、そのスプール・ファイルの内容の読み取りでは、より高いレベルの権限が必要になる場合があります。

スプール・ファイルおよび出力待ち行列のセキュリティの詳細については、『セキュリティ』のトピックを参照してください。

QPRTJOB ジョブ

QPRTJOB ジョブは、現在のジョブのユーザー名が、現在実行されているユーザー・プロファイルのものと同じでない場合に、スプール・ファイルが関連付けられるジョブです。

ユーザーが、システム・ジョブの代わりにスプール・ファイルの所有権を獲得するように、システム・ジョブを変更してユーザーのプロファイルの下に実行されるようにすることができます。例えば、「ネットワーク・スプール・ファイル送信」(SNDNETSPLF) コマンドを使用してスプール・ファイルを異なるシステム上のユーザー TINA に送信する場合、ジョブ 999999/TINA/QPRTJOB のファイルがスプールされます。システム・ジョブの代わりにこのユーザーのジョブのファイルのスプールすることによって、ユーザー TINA がスプール・ファイルを所有できるようになります。さらに、彼女が「スプール・ファイル処理」(WRKSPLF) コマンドを実行すると、彼女に送信されたスプール・ファイルが表示されます。

注: スプール・ファイルを所有するユーザーを指定するには、SPLFOWN パラメーターを使用します。

QPRTJOB ジョブはシステムによって自動的に作成されます。システム上では、ユーザーあたり複数の QPRTJOB が存在することができます。QPRTJOB は、9999 個のスプール・ファイルのデフォルト値を持ちます。「プリンター出力ファイルの最大数」(QMAXSPLF) システム値の数を変更することによって、この数を 999,999 の最大値に拡張することができます。「プリンター出力ファイルの最大数」(QMAXSPLF) システム値の詳細については、『実行管理機能』のトピックを参照してください。ユーザーの QPRTJOB がいっぱいになると、システムによって、自動的にユーザー用に新しいジョブが作成されます。SNDNETSPLF コマンドによって送信されるスプール・ファイルを受け取るユーザーごとに、別個の QPRTJOB が作成されます。SNDNETSPLF コマンドを使用してユーザー TINA と KEVIN にスプール・ファイルを送信すると、受信システム上には、999999/KEVIN/QPRTJOB と 999999/TINA/QPRTJOB という名前のジョブが存在することになります。

QPRTJOB ジョブは、さまざまなシステム機能によって作成および使用されます。例えば、

- 「TCP/IP スプール・ファイル送信」(SNDTCPSPLF) コマンドまたは SNDNETSPLF コマンドを使用して、異なる System i™ プラットフォーム上の別のユーザーにスプール・ファイルを送信する。
- スプール・ファイルを、VM または MVS™ から、VM/MVS ブリッジを経由して System i プラットフォームへ送信する。
- TCP/IP またはライン・プリンター・デーモン (LPD) のプロセスを使用してスプール・ファイルを受信する。
- 「Create Spooled File (スプール・ファイルを作成)」(QSPCRTSP) スプール API を使用して、別のユーザーにスプール・ファイルを作成する。
- 「プロファイルの設定」(QWTSETP) セキュリティ API を使用して、異なるユーザーにユーザー・プロファイルを設定してから、新しいスプール・ファイルを作成する。

実行中の他のアプリケーションは、QSPCRTSP および QWTSETP API を使用して、システム上に付加的な QPRTJOB ジョブを作成することができます。

- SPLFOWN が *CURGRPPRF に設定されている場合、UNIX® SETGID API を使用して、異なるユーザー・プロファイル、現在のユーザー・プロファイル、またはグループ・ユーザー・プロファイルにスプール・ファイルを作成する。

- UNIX SETUID API を使用して、ユーザー・プロファイルを異なるユーザーに設定してから、そのユーザーに新しいスプール・ファイルを作成する。

QPRTJOB ジョブは、非アクティブの状態が 24 時間を超えるまでは、引き続き再使用されます。非アクティブの状態にあるというのは、ジョブのすべてのスプール・ファイルが削除されており、24 時間を超えてそのユーザーが新しいジョブを 1 つも受信しないことを意味します。リカバリーはシステム・ジョブ QSPLMAINT によって実行されます。

スプール・ファイル・サブシステム

スプール・ファイル・サブシステム QSPL は、印刷装置書出プログラムの処理に使用され、印刷装置書出プログラムがアクティブのときにはアクティブでなければなりません。スプール・ファイル・サブシステムおよび個別の印刷装置書出プログラムは、他のサブシステムで実行されるジョブから制御することができません。

- 1 「印刷装置書き出しプログラム開始」(STRPRTWTR) コマンドは、書き出しプログラム・ジョブをスプール・ファイル・サブシステムのジョブ待ち行列にサブミットします。

書き出しプログラム・ジョブへの要求は QSPL ジョブ待ち行列に置かれ、次の場合に、実行のために QSPL ジョブ待ち行列の次の項目が選択されます。

- アクティブ・ジョブの数が MAXJOBS の QSPL サブシステム属性より小さい。
- QSPL ジョブ待ち行列のアクティブ・ジョブの数がジョブ待ち行列の MAXACT 属性より小さい。

スプール・ファイル・ライブラリー

スプール・ファイル・ライブラリー (QSPL または QSPLxxxx、ここで xxxx は、基本ユーザー ASP またはプライマリー独立 ASP の番号です) には、インライン・データ・ファイルおよびスプール・ファイルのデータの保管に使用されるデータベース・ファイルが含まれます。ライブラリー QSPL または QSPLxxxx の各ファイルは複数のメンバーを持つことができます。各メンバーには、インライン・データ・ファイルまたはスプール・ファイルのすべてのデータが含まれます。

スプール・ファイルを印刷または削除すると、スプール・ライブラリー内の関連データベース・メンバーはレコードがクリアされますが、除去されることはありません。このため、このメンバーを、別のインライン・データ・ファイルやスプール・ファイルに使用することができます。ライブラリー QSPL または QSPLxxxx で使用できるデータベース・メンバーが存在しない場合は、メンバーが自動的に作成されます。

空のスプール・ファイル・メンバーを新しいスプール・ファイルの作成に使用できるようにすると、システムのランタイム・パフォーマンスが向上します。ただし、多数の空のスプール・ファイルは大量のストレージを使用し、システムの異常時 IPL パフォーマンスを低下させる場合があります。例えば、各スプール・ファイル・メンバーは 24 KB のストレージを占有する場合があります。

DLTSPLF または CLROUTQ コマンドで古いスプール・ファイルを定期的に削除することによって、QSPL または QSPLxxxx ライブラリーを小さくしておくことが最善です。この手順によって、新しいデータベース・メンバーを収容するためにスプーリング・ライブラリーのサイズを増やすことなく、データベース・メンバーを再使用することができるようになります。

QSPL または QSPLxxxx ライブラリーのデータを表示することによって、データがクリアされないようにし、ストレージ・スペースの浪費を防ぐことができます。QSPL または QSPLxxxx ライブラリー内のデータベース・ファイルの確認に使用されるコマンドまたはプログラムは、データベース・ファイルおよびメンバーを割り振ることができる必要があります。印刷が完了した後に、書き出しプログラムは、割り振られたメンバーを除去しようとしても、メンバーをクリアすることはできません。メンバーはクリア

することができないため、このメンバーは、別のインライン・データ・ファイルまたはスプール・ファイルに使用することができず、「未使用プリンター出力記憶域の自動クリーンアップ」(QRCLSPLSTG) システム値を設定したり、RCLSPLSTG コマンドを実行したりして除去することもできません。

データベース・ファイルを QSPL または QSPLxxxx ライブラリーに保存する場合は、このファイルの 1 つのメンバー内のデータを表示する場合よりも多くの問題が発生する可能性があります。これは、データベース・ファイルを保存する場合、すべてのメンバーの割り振りに非常に長い時間がかかるからです。これらのファイルを復元すると現在と将来のスプール・ファイル・データが破壊されるため、このようなファイルを保存する理由はありません。

QSPL または QSPLxxxx ライブラリーのタイプおよび権限は変更すべきではありません。QSPL または QSPLxxxx 内のファイルに対する権限も変更すべきではありません。QSPL または QSPLxxxx ライブラリーとこの中のファイルは、システム・スプーリング機能でアクセスできるような特別の仕方で作成されています。ライブラリーやファイルを変更すると、一部のシステム・スプーリング機能が正常に動作しなくなり、スプール・ファイルのセキュリティ機構の健全性が破壊される場合があります。

独立 ASP におけるスプール・ファイル

スプール・ファイルは、独立ディスク・プール (独立補助記憶域プールまたは独立 ASP と呼ばれる) に置かれた出力待ち行列に保管することができます。

スプール・ファイルの作成者は、選択する出力待ち行列が、必要とする独立 ASP 上に存在するようにする必要があります。プリンター・ファイル、ジョブ属性、ジョブ記述、またはユーザー・プロファイルなどの複数の方法によって、これを管理することができます。

スプール・ファイルの作成者は、スプール・ファイルの作成中に、ネーム・スペース (ジョブの解決先であるライブラリー・セット) の変更が発生しないようにする必要があります。印刷装置書出プログラムが独立 ASP を使用し、そのスプール・ファイルを処理するようにするには、印刷装置書出プログラムは、その独立 ASP をネーム・スペースの一部とするジョブから開始する必要があります (ジョブは INLASPGRP が独立 ASP に設定されて実行されたか、ユーザーが SETASPGRP 独立 ASP を実行してそのネーム・スペースで独立 ASP を取得しました)。

- | ネーム・スペースが実際に変更され、スプール・ファイルが作成されている独立 ASP がオフに変更される
- | 場合 (これはネーム・スペースの変更がなされ、独立 ASP 上の予約が失われた場合に発生する)、配置とク
- | ローズのエラーが発生する場合があります。これによって、スプール内部情報においてデータが不正確にな
- | る場合があります。独立 ASP を再びオンに変更すると、このような不正確性を修正することができます。
- | この状態のリカバリーはバックグラウンド・ジョブで実行されるため、QSPRCxxxx システム・ジョブが操
- | 作を完了できるようになるまでは、ユーザーには、それらのスプール・ファイルの矛盾が見える場合があり
- | ます。独立 ASP がオフに変更されない場合、スプール・ファイルの作成は問題なく継続できます。

QSPMNxxxx ジョブは、以下の作業に責任を持ちます。

- 削除したスプール・ファイルの未使用データベース・メンバーをクリアする
- 「未使用プリンター出力記憶域の自動クリーンアップ」(QRCLSPLSTG) システム値で指定された日数以内に再使用されなかったデータベース・メンバーを自動的に除去する

- | QSPRCxxxx ジョブは、以下の作業に責任を持ちます。

- | • 損傷を受けた出力待ち行列がユーザーによって削除された場合に、取り残されたスプール・ファイル
- | を、プライマリー ASP にある QRCLxxxx ライブラリーの出力待ち行列 QSPRCLOUTQ に移動する。

注: オンに変更される ASP グループごとに QSPMNxxxxx と QSPRCxxxxx のシステム・ジョブが 1 つずつ存在します。

印刷装置書出プログラムまたはジョブが異常終了する場合は、スプール・ファイルまたは出力待ち行列が使用できなくなるか、一部の操作が許可されない不安定な状態になることがあります。そのような場合は、「スプール再利用の開始」(STRSPLRCL) コマンドを使用して、リカバリー不能状態になったスプール・ファイルと出力待ち行列を修復することができます。STRSPLRCL コマンドがスプール・ファイルと出力待ち行列の修復に失敗する場合は、独立 ASP をオフに変更してから、再びオンに戻す必要があります。

独立 ASP に入れられたスプール・ファイルは、ジョブが終了し、ジョブのスプール・ファイルがシステムや基本ユーザー ASP に存在しなくなると、自動的にジョブから切り離されます。重複したスプール・ファイルや重複したジョブ・エラー・メッセージを防ぐために、すべてのアプリケーションが、特定の日付と時刻が含まれている、スプール・ファイルの IDENTITY 値 JOBSYSNAME および CRTDATE を利用するようする必要があります。フェイルオーバーの際に独立 ASP がシステム A からシステム B に移動する場合、スプール・ファイルは元のジョブを使用できなくなる (スプール・ファイルはジョブから切り離されている) ことに注意してください。スプール・ファイルが切り離されていると、システム A で実行されたジョブと同じ ID で開始する別のジョブからのオペレーティング・システム保護は存在しません。

システムや基本ユーザー・ディスク・プール (*SYSBAS) 上に既に存在するスプール・ファイルが独立 ASP に含まれている場合は、この ASP をオンに変更することはできません。詳細については、『ディスク管理における印刷の考慮事項』のトピック・コレクションを参照してください。

独立 ASP の処理の詳細については、『独立ディスク・プールの使用』のトピックを参照してください。

以下の CL コマンドは、S/36 サポート、操作援助機能サポート、およびライブラリー・ネーム・スペースに関連した独立 ASP の制約において、ファイルをスプールしました。

- 「ジョブ変更」(CHGJOB) コマンド
- 「書き出しプログラム変更」(CHGWTR) コマンド
- 「スプール・ファイル・コピー」(CPYSPLF) コマンド
- 「ジョブ保留」(HLDJOB) コマンド
- 「スプール記憶域再利用」(RCLSPLSTG) コマンド
- 「ジョブ解放」(RLSJOB) コマンド
- 「ジョブ処理」(WRKJOB) コマンド
- 「スプール・ファイル処理」(WRKSPLF) コマンド

スプール・ファイルの管理

スプール・ファイルの管理には、スプール・ファイルの保留、スプール・ファイルの保留解除、スプール・ファイルの移動などのタスクが含まれます。

スプール・ファイル・リストの表示

スプール・ファイル (プリンター出力) のリストを表示するには、以下のいずれかの方法を使用します。

System i ナビゲーター

1. 「基本操作」を展開します。
2. 「プリンター出力」をクリックします。

デフォルト設定では、現行ユーザーに関連するすべてのプリンター出力が表示されます。「**プリンター出力**」を右クリックしてから、「**このビューのカスタマイズ**」 → 「**組み込み**」とクリックして、他のプリンター出力を表示することもできます。

文字ベース・インターフェース

「スプール・ファイル処理」(WRKSPLF) コマンドを使用します。

スプール・ファイルの内容の表示

スプール・ファイル (プリンター出力) の内容を表示するには、以下のいずれかの方法を使用します。

System i ナビゲーター

1. 「**基本操作**」を展開します。
2. 「**プリンター出力**」をクリックします。
3. 出力するプリンター出力ファイルを右クリックします。
4. 「**開く**」をクリックします。

文字ベース・インターフェース

「スプール・ファイル処理」(WRKSPLF) コマンドを使用し、そこでオプション 5 (表示) を使用します。

注:

1. System i ナビゲーター・インターフェースには、ASCII スプール・ファイルを表示する追加機能があります。
2. 文字ベース・インターフェースには、*LINE および *IPDS スプール・ファイルを表示できる追加機能があります。

スプール・ファイルに関連したメッセージの表示

スプール・ファイル (プリンター出力) に関連したメッセージを表示するには、以下のいずれかの方法を使用します。

System i ナビゲーター

1. 「**基本操作**」を展開します。
2. 「**プリンター出力**」をクリックします。
3. メッセージのあるプリンター出力ファイルを右クリックします。
4. 「**応答**」をクリックします。

文字ベース・インターフェース

「スプール・ファイル処理」(WRKSPLF) コマンドを使用し、そこでオプション 7 (メッセージ) を使用します。

スプール・ファイルの保留

選択したスプール・ファイル (プリンター出力) が一時的に印刷されないようにするには、以下のいずれかの方法を使用します。

System i ナビゲーター

1. 「基本操作」を展開します。
2. 「プリンター出力」をクリックします。
3. 保留にするプリンター出力ファイルを右クリックします。
4. 「保留」をクリックします。
5. 保留オプションを指定して「OK」をクリックします。

文字ベース・インターフェース

「スプール・ファイル処理」(WRKSPLF) コマンドを使用し、そこでオプション 3 (保留) を使用します。

注: 文字ベース・インターフェースには、ユーザー、印刷装置、用紙タイプ、ユーザー・データ、または ASP が同じであるスプール・ファイルすべてを、1 回の操作で保留できる追加機能があります。

スプール・ファイルの保留解除

保留されているスプール・ファイル (プリンター出力) の保留を解除するには、以下のいずれかの方法を使用します。

System i ナビゲーター

1. 「基本操作」を展開します。
2. 「プリンター出力」をクリックします。
3. 保留にするプリンター出力ファイルを右クリックします。
4. 「保留解除」をクリックします。

文字ベース・インターフェース

「スプール・ファイル処理」(WRKSPLF) コマンドを使用し、そこでオプション 6 (保留解除) を使用します。

注: 文字ベース・インターフェースには、ユーザー、印刷装置、用紙タイプ、ユーザー・データ、または ASP が同じであるスプール・ファイルすべてを、1 回の操作で保留解除できる追加機能があります。

スプール・ファイルの移動

スプール・ファイル (プリンター出力) を、ある出力待ち行列から別の出力待ち行列に移動するには、以下のいずれかの方法を使用します。

System i ナビゲーター

1. 「基本操作」を展開します。
2. 「プリンター出力」をクリックします。
3. 移動するプリンター出力ファイルを右クリックします。
4. 「移動」をクリックします。
5. プリンター出力を移動するプリンターまたは出力待ち行列の名前を指定し、「OK」をクリックします。

文字ベース・インターフェース

「スプール・ファイル処理」(WRKSPLF) コマンドを使用し、そこでオプション 2 (変更) を使用します。

注: 文字ベース・インターフェースには、ユーザー、印刷装置、用紙タイプ、ユーザー・データ、または ASP が同じであるスプール・ファイルすべての属性を、1 回の操作で変更できる追加機能があります。

スプール・ファイルの削除

スプール・ファイル (プリンター出力) を削除するには、以下のいずれかの方法を使用します。

System i ナビゲーター

1. 「基本操作」を展開します。
2. 「プリンター出力」をクリックします。
3. 削除するプリンター出力ファイルを右クリックします。
4. 「削除」をクリックします。
5. 「削除」をクリックして確認します。

文字ベース・インターフェース

「スプール・ファイル処理」(WRKSPLF) コマンドを使用し、そこでオプション 4 (削除) を使用します。

注: 文字ベース・インターフェースには、ユーザー、印刷装置、用紙タイプ、ユーザー・データ、または ASP が同じであるスプール・ファイルすべてを、1 回の操作で削除できる追加機能があります。

スプール・ファイルの PDF への変換

スプール・ファイルを PDF ファイルに変換するには、以下の手順に従います。

1. 変換するプリンター出力ファイルを右クリックします。
2. 「PDF への変換」をクリックします。
3. 「プリンター出力の PDF への変換」オプションを指定し、「OK」をクリックします。

スプール・ファイルの物理ファイルへのコピー

スプール・ファイルを物理ファイルにコピーするには、「スプール・ファイル・コピー」(CPYSPLF) コマンドを使用します。

元のスプール・ファイルはコピー操作によって影響を受けず、引き続き印刷装置書出プログラムによって印刷できます。スプール・ファイルを物理ファイルにコピーする場合、多くの装置属性はコピーできません。

スプール・ファイルの別のユーザーまたはシステムへの送信

スプール・ファイル (プリンター出力) を、TCP/IP を実行しているリモート・システムに送信する、またはシステム・ネットワーク体系配布サービス (SNADS) ネットワーク上の別のユーザーに送信するには、以下のいずれかの方法を使用します。

System i ナビゲーター

1. 「基本操作」を展開します。
2. 「プリンター出力」をクリックします。
3. 送信するプリンター出力ファイルを右クリックします。
4. 「送信」をクリックします。

5. 「**TCP/IP 経由で送信**」をクリックしてプリンター出力を TCP/IP を実行しているリモート・システムに送信するか、「**SNA 経由で送信**」をクリックしてプリンター出力を SNADS ネットワーク上の別のユーザーに送信します。
6. 送信オプションを指定して、「**OK**」をクリックします。

文字ベース・インターフェース

- | 「スプール・ファイル処理」(WRKSPLF) コマンドを使用し、そこでオプション 1 (送信) を使用して、プリンター出力を SNADS ネットワーク上の別のユーザーに送信します。スプール・ファイルを TCP/IP を使用する別のユーザーに送信するには、「TCP/IP スプール・ファイル送信」(SNDTCPSPLF) コマンドを使用します。

スプール・ファイルの属性の変更

スプール・ファイル (プリンター出力) の属性を変更するには、以下のいずれかの方法を使用します。

System i ナビゲーター

1. 「**基本操作**」を展開します。
2. 「**プリンター出力**」をクリックします。
3. 変更するプリンター出力ファイルを右クリックします。
4. 「**プロパティ**」をクリックします。
5. 変更する属性またはプロパティを指定して、「**OK**」をクリックします。

文字ベース・インターフェース

「スプール・ファイル処理」(WRKSPLF) コマンドを使用し、そこでオプション 2 (変更) を使用します。

注: 文字ベース・インターフェースには、ユーザー、印刷装置、用紙タイプ、ユーザー・データ、または ASP が同じであるスプール・ファイルすべての属性を、1 回の操作で変更できる追加機能があります。

スプール・ファイルの印刷の再開

特定ページのスプール・ファイルの印刷を再開するには、以下の手順に従います。

System i ナビゲーター

1. 「**基本操作**」 → 「**プリンター出力**」を展開します。
2. 印刷を再開するプリンター出力ファイルを右クリックし、「**プロパティ**」を選択します。
3. 「**プロパティ**」ウィンドウの「**一般**」タブで、「**印刷の再開**」をクリックします。
4. 次に表示されるウィンドウでページ情報を指定し、「**OK**」をクリックします。

1 つのスプール・ファイルの中断および別のスプール・ファイルの印刷

1 つのスプール・ファイル (プリンター出力) の印刷を一時的に停止し、別のスプール・ファイル (プリンター出力) の即時印刷を開始するには、以下の手順に従います。

System i ナビゲーター

1. 「**基本操作**」を展開します。
2. 「**プリンター出力**」をクリックします。

- 次に印刷を開始するプリンター出力ファイルを右クリックします。
- 「次を印刷」をクリックします。このプリンター出力が出力待ち行列の上部に移動します。
- 現在印刷中のプリンター出力ファイルを右クリックします。
- 「保留」をクリックします。
- 「ページの終わり」で保留するよう指定して、「OK」をクリックします。現行ページの終わりで、このプリンター出力の印刷が停止します。出力待ち行列の次のプリンター出力の印刷が開始します。
- 停止したプリンター出力ファイルを右クリックします。
- 「次を印刷」をクリックします。このプリンター出力が出力待ち行列の上部に移動し、最後に印刷されたページに続くページから、この出力の印刷が再開します。

スプール・ファイル通知メッセージを有効にする

スプール・ファイル (プリンター出力) が印刷を完了するか、印刷装置書出プログラムによって保留されるときに通知を受け取るようにするには、スプール・ファイル通知機能を使用できるようにする必要があります。

System i ナビゲーター

- 「ユーザーおよびグループ」を展開します。
- 「すべてのユーザー」をクリックします。
- 変更するユーザー名をダブルクリックします。
- 「ジョブ」をクリックします。
- 「セッションの表示」をクリックします。
- 「スプール・ファイル所有者にメッセージを送信」を選択します。

スプール・ファイル数の制御

システム上のスプール・ファイルの数は制限する必要があります。ジョブが完了しても、スプール・ファイルと内部ジョブ制御情報は、スプール・ファイルが印刷または取り消されるまで保持されます。システム上のジョブ数とシステムによって認識されるスプール・ファイル数によって、IPL と内部検索の実行に必要な時間数が増加し、必要な一時ストレージ量も増加します。

定期的に必要でなくなったスプール・ファイルを識別し、削除してください。スプール・ファイルのリストを表示する方法の詳細については、『スプール・ファイルのリストの表示』を参照してください。

- 生成されるジョブ・ログ数は、「ジョブ記述作成」(CRTJOBDD) コマンドまたは「ジョブ変更」(CHGJOB) コマンドの LOG および LOGOUTPUT パラメーター、または QLOGOUTPUT システム値を使用して制御できます。詳細については、ジョブ・ログの保留を参照してください。
- QMAXSPLF システム値を使用して、ジョブによって生成されるスプール・ファイルの最大数を指定できます。

有効期限が切れたスプール・ファイルの削除

「印刷装置ファイル変更」(CHGPRTF) コマンド、「印刷装置ファイル作成」(CRTPRTF) コマンド、「スプール・ファイル属性変更」(CHGSPLFA) コマンド、または「印刷装置ファイル一時変更」(OVRPRTF) コマンドの EXPDATE または DAYS パラメーターを使用して、スプール・ファイルを、「満了スプール・ファイル削除」(DLTEXPSPLF) コマンドを使用した削除に適したものにすることができます。

例えば、次のコマンドによって、DLTEXPSPLF コマンドを、システム上の期限が切れたすべてのスプール・ファイルを毎日削除するようにさせる、ジョブ・スケジュール項目が作成されます。

```
ADDJOBSCDE JOB(DLTEXPSPLF) CMD(DLTEXPSPLF ASPDEV(*ALL)) FRQ(*WEEKLY) SCDDATE(*NONE) SCDDAY(*ALL)
SCDTIME(010000) JOBQ(QSYS/QSYSNOMAX) TEXT('DELETE EXPIRED SPOOLED FILES SCHEDULE ENTRY')
```

スプール・ファイル・ストレージの再利用

「スプール記憶域再利用」(RCLSPLSTG) コマンドまたは「未使用プリンター出力記憶域の自動クリーンアップ」(QRCLSPLSTG) システム値を使用してスプール・ファイル・ストレージを再利用することができます。これらは、QSPL または QSPLxxxx ライブラリーからスプールされたデータベース・メンバーを除去する唯一の許可された方法です。他の方法では、重大な問題が発生する場合があります。

スプール・ファイル・ストレージの詳細については、『スプール・ファイル・ライブラリー』を参照してください。

「未使用プリンター出力記憶域の自動クリーンアップ」(QRCLSPLSTG) システム値

「未使用プリンター出力記憶域の自動クリーンアップ」(QRCLSPLSTG) システム値を使用して、スプール・パフォーマンスと補助ストレージ間の望ましいバランスを調整します。このシステム値を使用して、システム補助記憶域プール (ASP)、基本ユーザー ASP、および独立 ASP 上の未使用プリンター出力ストレージをクリーンアップすることができます。詳細については、『Storage system values: Automatically clean up unused printer output storage in the System values』のトピックを参照してください。

注: 「未使用プリンター出力記憶域の自動クリーンアップ」(QRCLSPLSTG) を 0 日に設定すると、システム・パフォーマンスは低下します。

いずれかのアプリケーション・プログラムにエラーが発生し、ユーザーに価値のない数千のスプール・ファイルが生成されたと仮定します。この場合、これらのスプール・ファイルはシステム上の多くのストレージ・スペースを使用します。スプール・ストレージを再利用するには、以下の作業を完了します。

1. 「未使用プリンター出力記憶域の自動クリーンアップ」(QRCLSPLSTG) システム値を 1 に変更します。
2. アプリケーション・プログラムが作成した不必要なすべてのスプール・ファイルを削除します。不必要なすべてのスプール・ファイルを削除した時刻を記録します。
3. 24 時間後、空のスプール・ファイル・メンバーが再利用されていない場合、システムは、空のスプール・ファイルによって使用されていた補助ストレージを再利用します。
4. 「未使用プリンター出力記憶域の自動クリーンアップ」(QRCLSPLSTG) システム値を、その以前の値に変更します。

「スプール記憶域再利用」(RCLSPLSTG) コマンド

あるいは、DAYS パラメーターを *NONE に設定した「スプール記憶域再利用」(RCLSPLSTG) コマンドを使用して、すべての空のスプール・ファイル・メンバーをすぐに再利用することもできます。このコマンドを使用して、システム補助記憶域プール (ASP)、基本ユーザー ASP、および独立 ASP 上の未使用プリンター出力ストレージをクリーンアップすることができます。

注:

1. DAYS パラメーターを *NONE に設定して RCLSPLSTG コマンドを実行すると、未使用のデータベース・メンバーはすぐに削除されます。これは、スプール・ファイルを作成したときに使用できる未使用メンバーのプールが存在しないことを意味します。

2. 出力待ち行列またはスプール・データベース・ファイルにロック競合が発生し、ボトルネックや重大なパフォーマンス上の問題が発生する場合があります。

いずれかのアプリケーション・プログラムにエラーが発生し、ユーザーに価値のない数千のスプール・ファイルが生成されたと仮定します。この場合、これらのスプール・ファイルはシステム上の多くのストレージ・スペースを使用します。スプール・ストレージを再利用するには、以下の作業を完了します。

1. アプリケーション・プログラムが作成した不必要なすべてのスプール・ファイルを削除します。
2. DAYS パラメーターを *NONE に設定し、RCLSPLSTG コマンドを実行します。システムは、不必要なスプール・ファイルによって使用されていたすべての補助スプール・ストレージをすぐに再利用します。

システム ASP ストレージ

ユーザー補助記憶域プール (ASP) または独立 ASP に直接スプール・ファイルを移動または作成することによって、スプール・ファイルによって占有されるストレージの量を減らすことができます。そのためには、ユーザー ASP または独立 ASP にあるライブラリーに出力待ち行列を作成する際に、SPLFASP パラメーターで *OUTQASP を指定します。

この出力待ち行列に置くすべてのスプール・ファイルは、スプール・ファイル・データを、ライブラリー QSPLxxxx にあるユーザー ASP または独立 ASP に保管します (xxxx はユーザー ASP または独立 ASP 番号)。

注: ユーザー ASP 上のファイルでは、ジョブへのリンクは引き続きシステム ASP 上に存在します。システム ASP が失われると、ユーザー ASP のすべてのスプール・ファイルが失われます。ユーザー ASP が失われると、そのユーザー ASP のスプール・ファイルのみが失われます。

スプール・ファイルの保存およびリストア

「ライブラリー保管」(SAVLIB)、「オブジェクトの保管」(SAVOBJ)、「ライブラリー復元」(RSTLIB)、および「オブジェクト復元」(RSTOBJ) の各 CL コマンドで SPLFDTA パラメーターを使用して、スプール・ファイルの印刷精度、属性、ID を失わずにスプール・ファイルを保存およびリストアすることができます。

保存したスプール・ファイルをリストアする際に 100% の印刷精度を維持するには、以下のようにする必要があります。

- スプール・ファイルを作成したときにそれらのファイルが存在したライブラリー内のすべての外部リソースを保存およびリストアする。
- ユーザー・プロファイル (スプール・ファイル所有者) が存在し、スプール・ファイルの印刷に必要なすべての外部リソース (統合ファイル・システム・ディレクトリー、True Type フォントの統合ファイル・システム・ファイル、フォント・リソース・オブジェクト、オーバーレイ、書式定義、ページ定義、ページ・セグメント、および組み込みオブジェクト統合ファイル・システム・ファイルなど) に対する適切な権限を持っていることを確認する。
- リソース割り振り表 (RAT) のフォントおよびリンク済みフォントが、スプール・ファイルを作成した時点と同じであることを確認する。これは、True Type のリンク済みフォントを使用するスプール・ファイルの場合にのみ必要です。
- スプール・ファイルによって使用されるすべての True Type フォントがシステム上の適切なディレクトリー内にあることを確認する。
- 環境変数 QIBM_AFP_RESOURCES_PATH が、組み込みオブジェクトを使用するが、ディレクトリー修飾されていないスプール・ファイルの適切なパスに設定されていることを確認する。

- スプール・ファイルを作成したときにそれらが存在したのと同じディレクトリーに、スプール・ファイルによって使用されるすべての組み込みオブジェクトをリストアする。

スプール・ファイルの保存またはリストアが処理中の場合、別の保存またはリストア操作はブロックされま
す。ブロックされた操作は、そのスプール・ファイルの保存またはリストアに失敗する場合があります。こ
れが発生した場合、診断メッセージが発行されます。

スプール・ファイルがリストアされると、元のジョブがまだシステムに存在する場合、このファイルは元の
ジョブに再接続されます。元のジョブが存在しない場合、スプール・ファイルは、切り離された状態でリス
トアされます。スプール・ファイルが切り離された状態でリストアされる場合は、完全修飾ジョブ名、スプ
ール・ファイル名、およびスプール・ファイル番号が同じであるスプール・ファイルが複数存在する可能性
があります。このような状態の場合、リストアしたスプール・ファイルへアクセスする際に、ユーザーまた
はアプリケーションは、ジョブ・システム名またはスプール・ファイル作成日付を含める必要があります。
これによって、オペレーティング・システムが正しいスプール・ファイルを選択できるようになります。

リストアするスプール・ファイルの順序は、主に、ファイルの状況などの属性によって決定され、必ずしも
リストアする順序にはなりません。詳細については、『出力待ち行列でのスプール・ファイルの順序』を参
照してください。

i5/OS の V5R4 以前に使用された、スプール・ファイルの保存およびリストアの手順については、『シス
テム管理におけるスプール・ファイルの保存』のトピック・コレクションを参照してください。

スプール・ファイル・サイズによる印刷の制御

「出力待ち行列作成」(CRTOUTQ) コマンドまたは「出力待ち行列変更」(CHGOUTQ) コマンドの
MAXPAGES パラメーターを使用して、スプール・ファイルの印刷をサイズによって制御できます。

例えば、午前 8 時から午後 4 時までの間、出力待ち行列 MYOUTQ において、40 ページを超えるスプ
ール・ファイルの印刷を制限するとします。また、正午から午後 1 時までの間は、10 ページ以下のスプ
ール・ファイルの印刷を許可するとします。これらの制限は、次のコマンドによって実現できます。

```
CHGOUTQ OUTQ(MYOUTQ) MAXPAGES((40 0800 1600) (10 1200 1300))
```

付録. 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。

〒106-8711
東京都港区六本木 3-2-12
日本アイ・ビー・エム株式会社
法務・知的財産
知的財産権ライセンス渉外

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。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

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。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

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「Spooled files」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。

i5/OS

IBM

IBM (ロゴ)

MVS

System i

Adobe、Adobe ロゴ、PostScript、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における商標または登録商標です。

UNIX は The Open Group の米国およびその他の国における登録商標です。

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。



Printed in Japan