

IBM

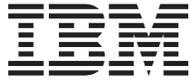
@server

iSeries

管理

バージョン 5 リリース 3





@server

iSeries

管理

バージョン 5 リリース 3

お願い

本書および本書で紹介する製品をご使用になる前に、113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、iSeries Access for Windows (プロダクト番号 5722-XE1) のバージョン 5、リリース 3、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 モデルで稼動するとは限りません。また CISC モデルでは稼動しません。

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： iSeries
Administration
Version 5 Release 3

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 2005.8

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 1998, 2005. All rights reserved.

© Copyright IBM Japan 2005

目次

iSeries Access for Windows の管理 . . .	1
V5R3 の新機能	2
このトピックの印刷	4
iSeries Access for Windows のネットワーク環境 . . .	4
Microsoft Windows Terminal Server	5
3 層環境での iSeries Access for Windows の使用	5
Microsoft Transaction Server (MTS) の使用 . . .	6
中間層からの iSeries サービスへのアクセス . . .	7
すべてのユーザーへの TCP/IP 構成の追加	9
すべてのユーザーに対する PC5250 ファイル場所	
の設定	9
複数のユーザーが使用する PC 用のユーザー・プ	
ロファイル	10
ODBC の管理	10
iSeries Access ODBC ドライバーの概要	11
iSeries Access ODBC ドライバーを使用するため	
のシステム・セットアップ	12
RDB ディレクトリーへのローカル・システム	
の追加	13
ODBC データ・ソースの指定	13
iSeries Access for Windows の ODBC のセキュリ	
ティー	14
危険性の高い ODBC セキュリティー・ストラ	
テジー	15
ODBC プログラム・セキュリティー・ストラ	
テジー	16
ODBC セキュリティーの関連情報	17
ODBC のトラブルシューティング	17
ODBC 診断およびパフォーマンス・ツール	17
iSeries Access ODBC のエラー・メッセージ	20
iSeries サーバー接続のトラブルシューティン	
グ	21
一般的な ODBC エラー	24
IBM サポートに提供する情報の収集	26
ホスト・サーバーの管理	27
OS/400 ホスト・サーバー	28
iSeries Access for Windows 機能別のホスト・	
サーバー	28
ファイル・サーバー	30
データベース・サーバー	31
データ待ち行列サーバー	36
ネットワーク・プリント・サーバー	36
セントラル・サーバー	37
リモート・コマンドおよび分散プログラム呼び	
出しサーバー	37
サインオン・サーバー	38
サーバー・ポートマッパー	38
拡張動的リモート SQL サーバー	
(QXDAEDRSQL)	38
DRDA/DDM サーバー	39
OS/400 ホスト・サーバーの使用	39
クライアント/サーバー通信の確立	40
iSeries サーバー上のサブシステム	45
iSeries サーバー上のシステム値	57
iSeries サーバー上のサーバー・ジョブの識別	
ホスト・サーバーでの EZ セットアップおよ	
び iSeries ナビゲーターの使用	63
サーバー出口プログラムの使用	63
出口プログラムの登録	64
出口プログラムの作成	67
出口プログラムのパラメーター	67
例: 出口プログラム	86
iSeries ネットサーバーの管理	101
ポリシーとアプリケーション管理機能を使用した制	
限の設定	101
iSeries Access for Windows ポリシーの概要	102
ポリシーのタイプとスコープ	103
ポリシーを使用するためのシステム・セットアッ	
プ	104
ポリシーのための iSeries TM サーバーの構成	104
ポリシーのためのクライアント PC の構成	104
ポリシー・ファイルの作成	105
iSeries Access for Windows ポリシー・リスト	106
機能別のポリシー	107
テンプレート別のポリシー	110
Secure Sockets Layer の管理	111
付録. 特記事項	113
商標	114
資料に関するご使用条件	115
コードに関する特記事項	115

iSeries Access for Windows の管理

このトピックでは、iSeries^(TM) Access for Windows^(R) がシステムにインストール済みで、その使用方法について熟知していることが前提になっています。クライアント/サーバー環境における iSeries Access for Windows の管理に関する情報を提供します。

2 ページの『V5R3 の新機能』

このリリースの管理トピックに含まれる新しい機能の概要を示します。

4 ページの『このトピックの印刷』

「iSeries Access for Windows の管理」の PDF バージョンの表示および印刷の方法を示します。

4 ページの『iSeries Access for Windows のネットワーク環境』

iSeries Access for Windows が稼働するいくつかのネットワーク環境について説明します。特に、3 層環境で iSeries Access for Windows を使用するか、または、Terminal Service を使用するリモート・ログオンのサポートを提供する Windows オペレーティング・システムに iSeries Access for Windows をインストールすることによって、クライアントが OS/400^(R) サービスを利用できるようにする方法について、詳しく解説します。さらに、同じ PC に複数のユーザーが割り当てられている場合の管理方法についても説明します。

10 ページの『ODBC の管理』

アプリケーションは、iSeries Access for Windows に組み込まれている ODBC ドライバーを利用して、ネットワーク内の DB2^(R) UDB for iSeries データベースに簡単にアクセスできます。このトピックでは、ODBC の概要、そのドライバーのセットアップ方法、およびトラブルシューティング・ガイドを提供します。

ODBC API を扱う際の情報と考慮事項については、「ODBC プログラミング」を参照してください。

27 ページの『ホスト・サーバーの管理』

このトピックでは、iSeries Access for Windows でよく使用されるいくつかのホスト・サーバーと、それらの効果的な管理方法と使用方法について説明します。

101 ページの『ポリシーとアプリケーション管理機能を使用した制限の設定』

iSeries Access for Windows では、さまざまな方法で制約事項とプロファイルを設定することができます。これらの方法には、Microsoft^(R) のポリシー・エディターを使用して設定できるポリシーと、iSeries ナビゲーターのアプリケーション管理機能があります。

➤ iSeries Access for Windows の概要と、ネットワーク内でのその使用方法については、トピック「iSeries Access for Windows スタートアップ・ガイド」を参照してください。iSeries Access for Windows のインストールと設定方法に関するヘルプ情報については「iSeries Access for Windows インストールおよびセットアップ」を参照してください。◀

iSeries Access for Windows の管理に必要な追加情報については、以下のトピックから選択してください。

- 111 ページの『Secure Sockets Layer の管理』
- 101 ページの『iSeries ネットサーバーの管理』

- iSeries Access for Windows プログラミング

注: 法的に重要な情報については、「コードに関する特記事項」を参照してください。

V5R3 の新機能

▶ V5R3 iSeries[™] Access for Windows[®] をインストールすると、OLE DB Provider に追加された新機能を介して、または新しい .NET データベース・プロバイダーを利用して、環境を管理することができます。また、新しいデータ・タイプのサポートおよび 10 進数の処理の精度の向上など、いくつかのデータベース機能拡張により、柔軟性が高まっています。

すべてのデータベースおよびデータ・アクセス機能は、ユニコードを使用でき、異なるエンコード・スキームおよび文字セットによるバリアを越えてデータを電子的に転送できます。さらに、すべてのデータベースおよびデータ・アクセス機能は、最新の ISO および ANSI SQL 規格に準拠しています。

V5R3 における多くの機能拡張は、iSeries Access for Windows が引き続き iSeries サーバーおよびそのデータベースを管理する上で最良の選択であることを保証します。

iSeries Access for Windows アドミニストレーターの新規機能には、以下のものがあります。

- **データ転送機能**

V5R3 では、データ圧縮を使用して転送を高速化でき、アプリケーションはユニコードの使用可能化および新しいユニコードのテキスト・ファイル・タイプを利用できます。数値データの 10 進数を高い精度で管理でき、新しい BINARY および VARBINARY SQL データ・タイプを使用できます。DB2[®] データベース・テーブルは UTF-8 および UTF-16 データをサポートするため、柔軟性が向上しています。

Microsoft[®] Excel では、標準の **日時セル** および **数値から文字への変換** に対する新しいサポートにより、希望するフォーマットのデータのサーバーへの転送およびサーバーからの転送を容易に管理できます。さらに、Excel アドインの **最近使った要求リスト** および **最終ディレクトリー** がサポートされ、管理が容易になります。

- **PC5250 印刷およびエミュレーション**

V5R3 iSeries Access for Windows PC5250 には、パーソナル・コミュニケーションズ 5250 のバージョン 5.7 に対する統合サポートが組み込まれています。バージョン 5.7 の主要な機能拡張のひとつは、ポップアップ・キーパッド、カラー・マッピング、および音声の視覚表示などのさらなるアクセシビリティ機能を管理できることです。LamAlef 双方向言語サポートおよび日本語 USB 106 キーボード・マッピングなどの特定のニーズがある場合、これらのサポートは有用で生産性を向上させるものです。他のより一般的な機能拡張には、マウス・マーキングおよび basic_ascii 印刷 PDF および PDT に対するサポートがあり、印刷機能および表示機能のオプションが増加します。

- **iSeries ナビゲーター**

iSeries ナビゲーターの新機能がいくつかあります。詳しくは、「iSeries ナビゲーター V5R3 の新機能」を参照してください。

- **受信リモート・コマンド**

V5R3 では、既知のユーザー ID のセキュリティー・コンテキストで実行されるリモート・コマンドのユーザー・プロファイル情報をロードできます。以前はユーザー・レジストリーまたは環境変数に対する必要な許可がなかったために失敗していた一部のコマンドも、現在は成功します。このオプションを、コマンドを実行するたびにリセットする必要がないように設定して、保管できます。詳細と例については、**ユーザーズ・ガイド (3ページ)** を参照してください。

- ODBC

V5R3 では、ODBC はアプリケーションのグローバル化のために BINARY および VARBINARY データ・タイプ、UTF-8 および UTF-16 データをサポートし、10 進数の精度が向上され、拡張された MTS サポートを提供します。

- データベース・プロバイダー

- **.NET Provider** - 新しい IBM.Data.DB2.iSeries Data Provider により、.NET Framework を使用するアプリケーションは .NET クラスおよびデータ・セットのフルセットを使用して DB2^(R) UDB for iSeries^(TM) データベースにアクセスできるようになります。これにより、既存の OLE DB Provider が補完され、最新の .NET テクノロジーを利用して、データの読み取りおよび検索を行い、変更を加えて、iSeries サーバーの機密保護機能のある環境で SQL サーバー・コマンドをデータ・オブジェクトに対して実行できます。詳しくは、「.NET Provider」を参照してください。詳しくは、「.NET プログラミング」を参照してください。
- **OLE DB Provider** - 既存のアプリケーションを扱うための IBMDBA400 のフルレンジの柔軟なサポートに対する機能拡張に加えて、新しい IBMDBASQL データ・プロバイダーを使用してコミットメント制御および MTS により SQL アプリケーションを開発および管理できます。新しいアプリケーションに、前進のみカーソルおよびブロック・フェッチに対するレコード・レベルのアクセスが必要である場合、新しい IBMDBARLA データ・プロバイダーはこの柔軟性を提供します。また、SQL データ圧縮およびパッケージ・サポート、BINARY および VARBINARY データ・タイプ、数値の精度の向上、NLSS 分類順序、および UTF-8 および UTF-16 データに対するサポートも追加されています。詳しくは、「OLE DB プログラミング」を参照してください。
- IBM.Data.DB2.iSeries プロバイダーについて詳しくは、*IBM^(R) DB2 UDB for iSeries .NET Provider Technical Reference* を参照してください。その他のプロバイダーについては、**OLE DB Technical Reference** を参照してください。これらの資料は、次のパスに従い、*Programmer's Toolkit* のトピックからアクセスできます。

「スタート -> プログラム -> IBM iSeries Access for Windows -> Programmer's Toolkit -> Programmer's Toolkit -> 共通インターフェース」

- 構成

V5R3 から、cwback および cwbenv コマンドがユニコード・エンコード方式を使用して情報を保管します。その結果、これらのユーティリティにより作成されたファイルは、旧バージョンの cwbrest または cwbenv を使用して復元できません。この問題に対応するために、V5R3 から新しいパラメーター */c* が cwback および cwbenv で使用可能になります。*/c* は、互換性 (Compatible) を表します。*/c* パラメーターの使用により、情報は ANSI コード・ページを使用して保管されます。これにより、旧バージョンの cwbrest および cwbenv により正常に復元されます。cwbrest および cwbenv の V5R3 バージョンは、ユニコードまたは ANSI として保管されたファイルを復元できます。

注: */c* が指定される場合は ANSI コード・ページが使用されるため、ANSI コード・ページで表現されない文字は失われます。

その他の情報

iSeries Access for Windows をインストールした後、iSeries Access for Windows フォルダーから「スタート -> プログラム -> IBM iSeries Access for Windows -> ユーザーズ・ガイド」を選択して、ユーザーズ・ガイドにアクセスします。

C/C++ Database API (最適化された SQL API) は拡張されません。将来のある時点で、このサポートは除去されます。データベース・アクセスには、他のテクノロジーを使用することをお勧めします。

Windows 98 (すべてのエディション) および ME オペレーティング・システムは、V5R3 iSeries Access for Windows ではサポートされません。 <<

新情報または変更の表示

技術的な変更が行われた個所を見つけやすいように、次のような表示を使用しています。

- >> イメージは、新しい情報または変更された情報の開始点を示すマークです。
- << イメージは、新しい情報または変更された情報の終了点を示すマークです。

>> このリリースの新機能または変更点に関するその他の情報は、「プログラム資料説明書」を参照してください。 <<

このトピックの印刷

PDF 版を表示またはダウンロードするには、「管理」(約 820 KB) を選択してください。

表示用または印刷用の PDF ファイルを Netscape Navigator からワークステーションに保存するには、次のようにします。

1. ブラウザーで PDF を開く (上記のリンクをクリック)。
2. Internet Explorer を使用している場合は、「名前を付けてターゲットを保存...」をクリックする。
Netscape Communicator を使用している場合は、「名前を付けてリンクを保存...」をクリックする。 <<
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 Acrobat Reader のダウンロード

PDF を表示または印刷するには、Adobe Acrobat Reader が必要です。これは、Adobe Web サイト

(www.adobe.co.jp/products/acrobat/readermain.html)  からコピーをダウンロードできます。 <<

iSeries Access for Windows のネットワーク環境

iSeries^(TM) Access for Windows^(R) エンド・ユーザーが iSeries のサービスにアクセスできるようにするための、いくつかの方法を提供します。一般に、これらの方法では、iSeries Access for Windows を使用する PC と iSeries サーバーとの間に直接接続が存在していることが前提となります。ただし、次の方法を使用すれば他のネットワーク環境も利用することができます。

- 5 ページの『Microsoft Windows Terminal Server』

Microsoft^(R) Windows Terminal Server は、1 つの Windows サーバーで複数のクライアント・セッションを同時に実行できる機能です。これを使用することにより、Windows 以外に、UNIX^(R)、Linux、DOS、OS/2^(R)、および他のネットワーク・ステーションなどの複数のクライアント・プラットフォームから接続できます。この機能を備えている Windows サーバーに iSeries Access for Windows をインストールすることにより、iSeries Access for Windows がインストールされていないワークステーションからでも iSeries のサービスにアクセスできるようになります。

- 5 ページの『3 層環境での iSeries Access for Windows の使用』

3 層環境の中間層に iSeries Access for Windows をインストールすることにより、さまざまな種類のクライアント・ワークステーションから iSeries のサービスにアクセスできるようになります。さらに、3 層環境には、拡張トランザクション管理などのような他の利点もいくつかあります。

iSeries Access for Windows は、複数のユーザーが使用する PC を管理するための手段も提供します。

- 9 ページの『すべてのユーザーへの TCP/IP 構成の追加』
CWBCFG コマンドを入力することによって、すべての PC ユーザーと iSeries サーバーの接続構成を行うことができます。
- 10 ページの『複数のユーザーが使用する PC 用のユーザー・プロファイル』
Windows オペレーティング・システムでは、ローミング・ユーザー・プロファイルおよび必須ユーザー・プロファイルを使用して、複数のユーザーを持つ PC を管理することができます。

Microsoft Windows Terminal Server

Microsoft^(R) Windows^(R) Terminal Server 1 つの Windows サーバーで複数のクライアント・セッションを同時に実行できる機能です。これを使用することにより、Windows 以外に、UNIX^(R)、Linux、DOS、OS/2^(R)、および他のネットワーク・ステーションなどの複数のクライアント・プラットフォームから接続できます。この機能を備えている Windows サーバーに iSeries^(TM) Access for Windows^(R) をインストールすることにより、iSeries Access for Windows がインストールされていないワークステーションからでも iSeries のサービスにアクセスできるようになります。

注： Terminal Services を実行し、Windows 2000 およびそれ以降のオペレーティング・システムを使用する場合は、「iSeries Access for Windows プロパティ」の「サービス」タブで、「検査時期」を「検査しない」に設定してください。

Microsoft Windows Terminal Server で iSeries Access for Windows を使用する場合は、インストール方法、サポート、および既知の問題については、APAR II11373  を参照してください。

Windows NT^(R) サーバー上の Terminal Services については、Microsoft の Windows NT Server 4.0 Terminal Server Edition Web サイト  を参照してください。

3 層環境での iSeries Access for Windows の使用

3 層環境の中間層に iSeries^(TM) Access for Windows^(R) をインストールすることにより、さまざまな種類のクライアント・ワークステーションから iSeries のサービスにアクセスできるようになります。それだけでなく、3 層環境には他の利点もあります。

- **多様なクライアントとサーバー・アプリケーションの間の統合性の向上:** 各種のクライアント上で実行されている複数のエンド・ユーザー・アプリケーションが、Windows サーバー上の複数のアプリケーションと、同時に通信することができます。Windows サーバー上の各アプリケーションも、複数のデータベースと同時に通信することができます。
- **Microsoft^(R) Transaction Server (MTS) を使用した拡張トランザクション管理:** 3 層環境では、相互に依存しなければそれぞれが正常に完了できないような、複雑なトランザクションなどを処理することができます。(つまり、あるトランザクションが完了するためには、他のすべてのトランザクションが正常に完了しなければなりません。)
- **Microsoft Internet Information Server (IIS) の使用による iSeries サーバーから Web ページへのデータのインポート:** IIS は Active Server Pages を使用して、Web ページを DB2 Universal Database^(TM) for iSeries からのデータで動的に更新します。

すべての 3 層環境では、コンポーネントとアプリケーションが 3 つの層に分割されます。3 つの層がそれぞれ異なる PC (端末) 上にあり、ネットワークを介して互いに通信する形であってもかまいません。通常、各層は次のような特性を備えています。

クライアント層

この層には、エンド・ユーザーがデータを操作するために使用できるインターフェースとアプリケーションが含まれています。これには、たとえば、ネットワーク・ステーションで実行されている Web ブラウザーや、リモート・コンポーネントを使用するカスタムビルト・アプリケーションなどが含まれます。この層では iSeries Access for Windows クライアントは使用されません。

中間層

この層にはビジネス・ロジックまたはアプリケーション・ロジックが含まれます。iSeries Access for Windows を使用する環境では、Microsoft Active Server Pages スクリプトまたはリモート・コンポーネントを実行する、Windows サーバーをこの層に含める必要があります。さらに、この層は、Microsoft の Internet Information Server (IIS) および Microsoft Transaction Server (MTS) を使用して、クライアント層とのトランザクションを管理します。iSeries Access for Windows は、ODBC ドライバーまたは IBM DASHDB OLE DB Provider を使用してクライアント上の MTS をサポートし、データベース層との通信を処理します。.NET、OLE DB、ActiveX Data Objects (ADO)、および Remote Data Service を使用して、中間層のコンポーネントからデータにアクセスできます。

中間層についての詳細は、次のトピックを参照してください。

- 『Microsoft Transaction Server (MTS) の使用』
- 7 ページの『中間層からの iSeries サービスへのアクセス』

データベース層

通常、この層には、DB2 Universal Database for iSeries が含まれます。ユーザー・アプリケーションは、ホスト・サーバー・プログラムまたはカスタムビルトの iSeries プログラムを利用して、このサービスおよびその他各種の iSeries サービスにアクセスできます。

Microsoft Transaction Server (MTS) の使用

iSeries^(TM) Access for Windows^(R) クライアントは、V5R1 またはそれ以降のサーバーに対して、iSeries Access ODBC ドライバーおよび IBM DASHDB OLE DB Provider を使用して、MTS バージョン 2.x およびそれ以降をサポートします。

MTS

MTS は、インターネット・サーバー・アプリケーションを開発、配置、および管理するための、Microsoft^(R) コンポーネント・ベースのプログラミング・モデルおよびランタイム環境です。多くの 3 層環境では、Active Server Pages (ASP) は、MTS コンポーネントを呼び出して、データベース、メインフレーム、アプリケーション、およびメッセージ待ち行列にアクセスします。MTS コンポーネントは、3 層環境の中間層で実行される iSeries Access for Windows と共に使用され、クライアント・アプリケーション、iSeries Access for Windows のコンポーネント、および関連データベースの間のトランザクションを管理します。

MTS は、Microsoft Distributed Transaction Coordinator (MSDTC) を使用して、複数のデータベース管理システム (Database Management System: DBMS) にまたがるトランザクションを管理し、互いに他の成功に依存しながらインプリメンテーションを行うトランザクションを取り扱うときの 2 フェーズ・コミットの保全性を確保します。

インプリメンテーションに関する注意

- MSDTC が iSeries Access ODBC ドライバーをロードできないときは、SQLSetConnectAttr (SQL_ATTR_ENLIST_IN_DTC) は失敗し、理由コード 2 (XaRmCreate failed) が返されます。iSeries Access for Windows PC5250 のエミュレーター・コンポーネントがインストール済みである場合は、MSDTC システム環境パスが設定されます。これを避けるには、MSDTC を実行する PC 上のシステム環境パスに、iSeries Access for Windows がインストールされているディレクトリー内の Shared ディレクトリーへのパスを組み込む必要があります。例：C:\Program Files\IBM\Client Access\Shared
- SSL か、iSeries ナビゲーターの「接続 → プロパティー」ダイアログで構成可能なその他の値を使用している場合は、iSeries ナビゲーターの中の iSeries 接続名は、MTS が管理するクライアント PC で指定されている接続名に一致している必要があります。MSDTC は、MTS が管理する iSeries Access for Windows ODBC クライアント PC と同じ接続名を使用して、DB2^(R) UDB for iSeries^(TM) データベースに接続します。MSDTC 接続の接続プロパティーを変更するには、システム・アカウント・レジストリーを変更する必要があります。

これを行うための方法の 1 つとして、受信リモート・コマンド (Incoming Remote Command) (IRC) と CWBENV ユーティリティーを組み合わせて使用する方法があります。

1. いずれかのクライアント PC で CWBENV を実行して、特定の環境に関する構成情報を抽出する。
2. 結果のファイルを MSDTC PC にコピーする。
3. iSeries Access for Windows リモート・コマンド・サービスを開始して、確実にローカル・システム環境で実行されるように構成する。
4. PC5250 セッションで RUNRMTCMD コマンドを使用し、CWBENV コマンドを MSDTC PC に送信して、環境をインポートする。

これらの機能について詳しくは、iSeries Access for Windows のプログラム・グループの中の「ユーザース・ガイド (3ページ)」を参照してください。

MTS について詳しくは、Microsoft MTS Web サイト  を参照してください。

中間層からの iSeries サービスへのアクセス

中間層のコンポーネントから iSeries^(TM) サーバーにアクセスできるようにするには、いくつかの方法があります。

注：中間層のコンポーネントにはユーザー・インターフェースがありません。したがって、iSeries Access がサインオン情報に対するプロンプトを出すと、3 層アプリケーションが停止しているように見えることがあります。この状況が起こらないようにするには、開発者は、iSeries サーバーに対して、必要な接続情報 (ユーザー ID およびパスワード) を指定するための新しいシステム・オブジェクトを使用する必要があります。このオブジェクトのプロンプト・モード値は「プロンプトなし (prompt never)」でなければなりません。

iSeries^(TM) Access for Windows^(R) の .NET Data Provider

最新 V5R3 iSeries Access for Windows データベース・オフリングは、iSeries サーバーの柔軟性を引き続き実証し、ユーザーが常に業界の先端テクノロジーを利用できるようにします。IBM^(R) DB2^(R) UDB for iSeries .NET Provider は、Microsoft^(R) の .NET Data Access Framework を使用してアプリケーションを作成するプログラマーが iSeries データベースに最高のパフォーマンスでアクセスできます。本書を通して、管理対象プロバイダーは、IBM DB2^(R) UDB for iSeries^(TM) .NET Provider および

IBM.Data.DB2.iSeries data provider と置き換え可能な用語として使用されます。言及される名称に関わらず、.NET データ・タイプおよび SQL 機能の完全セットを利用して、アプリケーションで容易に iSeries サーバー・データベースに安全に保管されているデータを扱うことができます。

詳しくは、「.NET プログラミング」を参照してください。

iSeries Access for Windows の OLE DB Provider

ほとんどのアプリケーションおよびコンポーネントは、ActiveX Data Object (ADO) を通じて、iSeries Access for Windows OLE DB Provider を使用します。この技法をインプリメントすることにより、次の 4 つの主要な利点が得られます。

- 開発者は、1 つのインターフェースおよびプログラミング手法にわずかな変更を加えるだけで、iSeries のプログラム、コマンド、SQL 照会、ストアード・プロシージャ、および物理ファイルと論理ファイルにアクセスできます。
- iSeries と PC データ・タイプ間の自動データ変換がサポートされます。
- レコード・レベルのファイル・アクセスのためのサポートが提供されるので、SQL に関連したオーバーヘッドを回避できます。
- アプリケーションのインプリメントや開発が比較的簡単になります。この方法は、一般に、3 層アプリケーションを開発するための最も簡単なテクノロジーです。

詳しくは、「OLE DB プログラミング」を参照してください。

iSeries Access for Windows の ODBC ドライバー

さらに、Microsoft OLE DB Provider for ODBC (MSDASQL) を使用することにより、ADO または Remote Data Services (RDS) を介して、iSeries Access ODBC ドライバーにアクセスすることができます。

ADO を介して ODBC にアクセスする方法については詳しくは、「ODBC ドライバーへのアクセスに使用するインターフェースの選択」を参照してください。

iSeries Access ODBC ドライバーについてさらに詳しくは、「ODBC プログラミング」を参照してください。

注: iSeries Access for Windows OLE DB Provider と、iSeries Access ODBC ドライバーの一部の機能を使用するには、MDAC バージョン 2.5 およびそれ以降が必要です。

ActiveX オートメーション・オブジェクト

iSeries Access for Windows クライアントには、新しい拡張 ActiveX オートメーション・オブジェクトのライブラリーが組み込まれており、開発者は、これを中間層の開発に利用することができます。これらのオブジェクトを使用して次のものにアクセスできます。

- iSeries データ待ち行列
- リモート・コマンドおよび分散プログラム呼び出し
- 管理オブジェクト
- iSeries システム・オブジェクト
- iSeries データベース表へのデータ転送アクセス

場合によっては、ADO より ActiveX オブジェクトを使用した方が柔軟性と機能性が高くなる場合がありますが、プログラミングは少々複雑になります。

注: iSeries Access for Windows クライアントには、Windows 95/NT クライアント (XD1 プロダクト) からのオートメーション・ライブラリーが含まれています。データベースも含めてこれらのオートメーション・オブジェクトは、3 層環境内では使用できません。

Express C/C++ APIs

iSeries Access for Windows API は、OS/400(R) ホスト・サーバーへの高速かつ低レベルのアクセス手段を提供します。しかし、このような API を使用するには、開発者は C/C++ に関する知識と経験を持つ必要があります。特に、開発者は、C の API とデータ型について精通している必要があり、コンポーネントを作成するときはスレッド・セーフティーについても考慮する必要があります。

すべてのユーザーへの TCP/IP 構成の追加

コマンド・プロンプトから、または「スタート → ファイル名を指定して実行」から、CWBCFG コマンドを入力することによって、PC で定義されているすべてのユーザーのための iSeries^(TM) サーバーの接続構成を行うことができます。このコマンドを使用することにより、Windows^(R) のデフォルト・ユーザー・プロファイルに構成情報が追加されます。このプロファイルは、追加のユーザー・プロファイルを作成する場合に使用されます。

また、CWBCFG を使用して、PC5250 エミュレーターがファイルをオープンまたは作成する際に使用する場所を追加または変更できます。CWBCFG は、PC のロケーション設定を変更できます。

CWBCFG については、オンラインの「iSeries^(TM) Access for Windows ユーザーズ・ガイド (3ページ)」を参照してください。

すべてのユーザーに対する PC5250 ファイル場所の設定



PC5250 エミュレーターがすべてのファイルをすべての定義済みユーザー用に検索および保管するデフォルトの場所は、PC のすべてのユーザーにより共有されていますが、一部のユーザーは書き込み権限を持っていません。デフォルトの場所は次のとおりです。

(iSeries^(TM) Access for Windows^(R) のインストール・フォルダー)¥emulator¥private

このデフォルトの場所は、すべての許可ユーザーにより、「iSeries^(TM) Access for Windows プロパティ」の「PC5250」タブから変更できます。すべてのユーザーに対して同時にこのデフォルトの場所を変更するには、管理者は、コマンド・プロンプトから CWBCFG コマンドを使用して、/pc5250path オプションを指定します。

注:

- CWBCFG が実行された後に作成されたすべてのユーザー・アカウントは、CWBCFG により設定されたデフォルトの場所を使用します。
- 管理者のみが CWBCFG を使用できます。
- CWBCFG は、いずれのファイルも元の場所から新規の場所に移動しません。ファイルは必要に応じて手動で移動する必要があります。

CWBCFG については、オンラインの「iSeries Access for Windows ユーザーズ・ガイド (3ページ)」を参照してください。

複数のユーザーが使用する PC 用のユーザー・プロファイル

複数の iSeries^(TM) Access for Windows^(R) ユーザーを持つ PC を管理することができます。このタイプの管理は、ローミング・プロファイル、および必須プロファイルの使用により、Windows オペレーティング・システムの機能として行うことができます。

注： ネットワーク内に複数ユーザーの管理方式をインプリメントする方法は、ご使用の Windows オペレーティング・システム用の Microsoft^(R) Resource Kit を参照してください。Resource kit は Microsoft から入手可能なもので、Microsoft Developers Kit に組み込まれています。

ローミング・ユーザー・プロファイル

ローミング・ユーザー・プロファイルは、Windows の PC 間で移動できるユーザー・プロファイルです。構成変更はユーザーと共に移動します。ローミング・ユーザー・プロファイルは通常、Windows サーバー上に常駐します。個々のローミング・ユーザーのユーザー・プロファイル設定でユーザー・プロファイル・パスにより指定されている Windows サーバーに、そのユーザー用のディレクトリーがあります。このディレクトリーには、レジストリー情報と、各ユーザー用の開始メニューおよびデスクトップ情報が含まれています。ローミング・ユーザー・プロファイルは、Windows NT^(R) ファミリーを実行している PC 間でのみ移動することができます。

必須ユーザー・プロファイル

必須ユーザー・プロファイルは、システム管理者がセットアップするユーザー・プロファイルで、PC ユーザーが Windows PC で使用するためのプロファイルです。通常、これらのユーザーは、それぞれの設定値は変更できません。必須ユーザー・プロファイルは、1 つの PC に存在することも、複数の PC 間を移動することも可能です。

ODBC の管理

Open Database Connectivity (ODBC) は、データベースへのアクセスに適用される Microsoft^(R) 規格の 1 つです。ODBC は、構造化照会言語 (SQL) を使用してデータベースにアクセスするための、明確に定義された一組のアプリケーション・プログラミング・インターフェース (API) を備えています。

11 ページの『iSeries Access ODBC ドライバーの概要』

このトピックでは、ODBC の概説と iSeries^(TM) Access for Windows^(R) での使用方法について説明します。

12 ページの『iSeries Access ODBC ドライバーを使用するためのシステム・セットアップ』

このトピックでは、ODBC ドライバーをサポートするように環境をセットアップするための手順を示します。ODBC ドライバーの構成方法に関するヘルプ情報については、iSeries Access for Windows プログラム・グループから ODBC アドミニストレーターを開始し、オンライン・ヘルプを参照してください。

14 ページの『iSeries Access for Windows の ODBC のセキュリティー』

このトピックには、ODBC を操作するときのいくつかのセキュリティーに関する考慮事項を重点的に解説し、セキュリティーに関するより詳細な参照情報が示されています。

iSeries ODBC Driver for Linux

このトピックでは、iSeries ODBC Driver for Linux をインストールし、それを使用して iSeries データベースにアクセスする方法について説明しています。

注：iSeries ODBC Driver for Linux は iSeries Access for Windows には含まれていません。これは Linux オペレーティング・システムでのみ使用される独立した製品です。

17 ページの『ODBC のトラブルシューティング』

このトピックは、iSeries Access for Windows と ODBC で一般的に発生しやすい問題を解決するために役立ちます。また、パフォーマンス上のボトルネックを解消するために利用できるいくつかのツールも示されています。テクニカル・サポートに連絡する前に、まずこのトピックの情報を調べてください。

ODBC サポートをアプリケーションに組み込む方法については iSeries Access for Windows の「ODBC プログラミング」を参照してください。この資料には、以下のサブトピックに関する情報が載せられています。

- ODBC API リスト
- ODBC API のインプリメンテーション
- プログラミング例
- ODBC のパフォーマンス

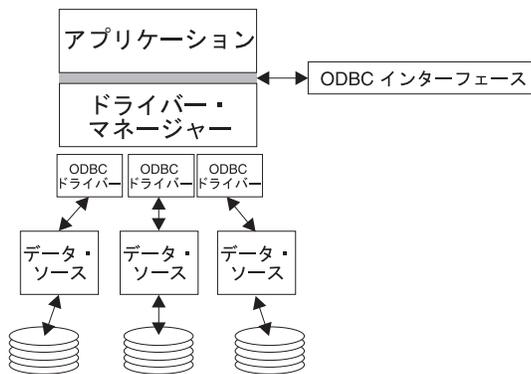
iSeries Access ODBC ドライバーの概要

iSeries^(TM) Access ODBC ドライバーは、構造化照会言語 (SQL) を使用してデータベースの情報にアクセスするためのアプリケーション・プログラミング・インターフェース (API) の集合です。iSeries Access ODBC ドライバーを使用することにより、アプリケーションは、同じソース・コードを使用して iSeries サーバー上の異なるデータベースにアクセスし、各アプリケーションにとって最も効率的な形式でデータを取り扱うことができます。ODBC は、複数の DBMS を取り扱う必要のある移植可能なアプリケーションやコンポーネントを作成するための、比較的単純なモデルをアプリケーション開発者に提供します。

ODBC アーキテクチャーには、アプリケーション、ドライバー・マネージャー、ODBC ドライバー、およびデータ・ソースが含まれています。iSeries Access は、32 ビットと 64 ビットの ODBC ドライバーを備えています。64 ビット・バージョンの Windows^(R) で実行すると、32 ビット ODBC ドライバーと一緒に 64 ビット ODBC ドライバーも自動的にインストールされます。64 ビット・バージョンの Windows で実行する ODBC アプリケーションは、いずれのビット・バージョン向けにコンパイルされたかに応じて、自動的に該当する ODBC ドライバーを使用します。たとえば、64 ビット・アプリケーションで使用できるのは 64 ビット・ドライバーだけです。

アプリケーションが ODBC を使用するようにするには、データ・ソースをセットアップする必要があります。データ・ソースのセットアップには、ODBC アドミニストレーターを使用できます。ODBC アドミニストレーターには 32 ビットと 64 ビットの 2 つのバージョンがあり、いずれも iSeries^(TM) Access for Windows フォルダーからアクセスできます。ODBC アドミニストレーターを使用する際には、ユーザー、システム、そしてファイルの 3 つのデータ・ソース・タイプをセットアップするオプションがあります。データ・ソースの構成方法について詳しくは、「iSeries Access for Windows プログラミング」の『64 ビット ODBC サポート』を参照してください。

ODBC のコンポーネント



RV3W364-1

アプリケーション 処理を行い、SQL ステートメントを実行するための ODBC 関数を呼び出します。

ドライバー・マネージャー ODBC 関数呼び出しを処理し、要求をドライバーに転送します。

ドライバー ODBC 関数呼び出しを処理し、SQL 要求を特定のデータ・ソースに実行依頼し、結果をアプリケーションに戻します。

データ・ソース データ・ソースを使用するには、データ・ソース名 (Data Source Name: DSN) を作成する必要があります。DSN には、DBMS にどのようにアクセスするかについての情報が入っています。次のいずれかの DSN を指定できます。

- **ユーザー DSN:** これはコンピューターにとってローカルなデータ・ソースで、それを作成したユーザーのみが使用できるものです。この情報はレジストリーに格納されます。
- **システム DSN:** これは、特定のユーザー専用のもではなく、コンピューターにとってローカルなデータ・ソースです。システムその他、必要な特権を持つどのユーザーも、システム DSN によりセットアップされたデータ・ソースを使用することができます。この情報はレジストリーに格納されます。

注: 64 ビットプロセッサを搭載した PC では、レジストリーのシステム部分は 32 ビットの部分と 64 ビットの部分に分かれています。32 ビットの ODBC アドミニストレーターを使用して構成されたシステム DSN は、32 ビット・アプリケーションでのみ使用可能です。同様に、64 ビットの ODBC アドミニストレーターを使用して構成されたシステム DSN は、64 ビット・アプリケーションでのみ使用可能です。

- **ファイル DSN:** これはファイル・ベースのデータ・ソースで、同じドライバーをインストールしているすべてのユーザーがこれを共有して、データベースにアクセスすることができます。この種のデータ・ソースは、特定のユーザー専用にしたり、コンピューターにとってローカルにしたりする必要はありません。

ODBC について詳しくは、Microsoft^(R) Web サイトを参照してください。

iSeries Access ODBC ドライバーを使用するためのシステム・セットアップ

iSeries^(TM) Access ODBC ドライバーは ODBC バージョン 3.5 準拠のドライバーです。このドライバーには Microsoft^(R) Data Access Components (MDAC) バージョン 1.5 あるいはそれ以降が必要です。

Microsoft ActiveX Data Objects (ADO) を使用するアプリケーションには、MDAC バージョン 2.1 またはそれ以降がインストール済みの環境が必要です。MDAC バージョン 2.1 およびそれ以降のランタイムには、ADO、Microsoft OLE DB Provider for ODBC、および iSeries^(TM) Access for Windows^(R) ODBC を使

用するアプリケーションが iSeries データにアクセスするための追加機能が提供されます。アプリケーションで接続プーリングまたは Microsoft Transaction Server (MTS) サポートが使用されている場合は、最新バージョンの MDAC をインストールすることをお勧めします。MDAC は、次の Microsoft Web サイトからダウンロードできます。www.microsoft.com/japan/msdn/data/download/mdac/27/27.asp 

ODBC ドライバーの構成については、「ODBC データ・ソース」のトピックを参照してください。「RDB ディレクトリーへのローカル・システムの追加」のトピックで説明されているステップに従って、構成を行います。

ODBC を介した独立 ASP の使用はオプションです。このサポートの構成について詳しくは、「独立 ASP (14ページ)」を参照してください。

特定データ・ソース用のオプションの構成方法に関するヘルプ情報については、iSeries Access for Windows プログラム・グループから ODBC アドミニストレーターを開始し、構成するデータ・ソースを選択し、オンライン・ヘルプを参照してください。

RDB ディレクトリーへのローカル・システムの追加

ODBC、OLE DB、または .NET Data Provider を使用するには、RDB ディレクトリーにローカル・システム名が入っていないなければなりません。

RDB ディレクトリーにローカル・システムを追加する方法：

1. コマンド・プロンプトから、リレーショナル・データベース・ディレクトリー項目の追加 (ADDRDBDIRE) CL コマンドを実行する。
2. ADDRDBDIRE 画面で値の入力を求められたら、Relational Database (リレーショナル・データベース) パラメーターとしてシステムの名前を入力する。
3. Remote Location (リモート・ロケーション) パラメーターとして *LOCAL と入力する。

ご使用のシステムのバージョンが V5R2 またはそれ以降であり、アプリケーションが独立 ASP 内のデータにアクセスする場合は、データベース (RDB) 名の設定にさらにステップが必要になることがあります。RDB 名は、システム ASP およびいずれかのユーザー ASP、またはシステム ASP に関連するリンクされた ASP グループで構成されるネーム・スペース名と一致しています。独立 ASP について詳しくは、「ディスク管理」を参照してください。

注： ODBC では、[catalog name].[schema name].identifier の形式の完全修飾名を使用できます (たとえば、identifier は、テーブル、ビュー、プロシージャなどの名前)。DB2^(R) UDB for iSeries^(TM) の SQL のインプリメンテーションでは、これは [RDB name].[collection name].identifier に該当します。

ODBC データ・ソースの指定

アプリケーションがデータにアクセスしてそれを処理できるようにするには、データ・ソースを指定する必要があります。

データ・ソースを指定する方法：

1. iSeries^(TM) Access for Windows^(R) プログラム・グループから ODBC アドミニストレーターを始動する。
2. データ・ソースのタイプに該当するタブを選択する。詳しくは、11 ページの『iSeries Access ODBC ドライバーの概要』を参照してください。

3. リストから既存のデータ・ソースを選択する。または、新しいデータ・ソースを作成するには「追加」を選択する。既存のデータ・ソースを使用する場合は、「構成」をクリックし、ステップ 5 に進みます。
4. データ・ソース用の iSeries Access ODBC ドライバーを選択し、「終了」をクリックする。
注：ドライバーのリスト内に Client Access ODBC Driver (32-bit) が含まれているのがわかるはずで
す。この名前は、前のリリースの Client Access によって作成されたデータ・ソースが継続して正しく
使用できるようにリストされています。いずれの名前も同一 ODBC ドライバーを示すことになりま
す。どちらの名前を使用してもかまいませんが、将来のリリースでは、Client Access ODBC Driver
(32-bit) の名前は除去されることになります。
5. 「iSeries Access for Windows ODBC セットアップ」ダイアログを使用して、必要なオプションを指定
する。制御機能の説明については、F1 キーまたは「ヘルプ」ボタンを使用して、データ・ソースのオ
ンライン・ヘルプを参照してください。

注： データ・ソース名は最大 32 文字で、最初の 1 文字は英字でなければなりません。また下記の文字は使用できません。

データ・ソース名に使用できない文字	
左大括弧 ([)	疑問符 (?)
右大括弧 (])	アスタリスク (*)
左中括弧 ({)	等号 (=)
右中括弧 (})	感嘆符 (!)
左括弧 (()	アットマーク (@)
右括弧 ())	セミコロン (;)

独立 ASP

ODBC によって**独立 ASP**を使用するためには、ODBC DSN を構成して、以下の手順に従います。

1. 「サーバー」タブを選択する。
2. 接続したい**独立 ASP** に対応する **RDB 名**を指定する。
3. 「デフォルトのデータベースを以下によってオーバーライドする」をクリックする。
4. 接続したい**独立 ASP** に対応する **RDB 名**を指定する。
5. **RDB 名**を指定しないと、ODBC 接続を行うためのユーザー・プロファイルのジョブ記述からデフォルトの **RDB 名**が決めます。デフォルトでは、このドライバーは、ユーザー・プロファイルの設定値を使用して ODBC 接続を行います。

独立 ASP について詳しくは、ディスク管理を参照してください。

iSeries Access for Windows の ODBC のセキュリティー

以下の情報は、iSeriesTM サーバーまたは iSeries Access for Windows^R の全体のセキュリティー計画に関する包括的なガイドを目的とするものではありません。単に、iSeries Access for Windows および ODBC のユーザーに影響を与えるセキュリティー計画上の問題点に関する概要を示すだけです。詳細情報について

は、「IBM^R 機密保護解説書」  を参照してください。

- 危険性の高い ODBC セキュリティー・ストラテジー

- ODBC プログラム・セキュリティー・ストラテジー
- ODBC セキュリティーに関するその他の情報リソース

危険性の高い ODBC セキュリティー・ストラテジー

システム管理者によっては、データ自体の保護対策をとらずに、データへのアクセスを管理することで保護対策を講じようとする場合があります。これは、ユーザーがデータのアクセスに使用するありとあらゆる方法を管理者が熟知していることが前提になり、きわめて危険な方法です。一般的な ODBC セキュリティー手法のうち、避けた方がよいものの一部を挙げておきます。

コマンド行セキュリティー

これは文字ベース・インターフェースまたは 5250 エミュレーション・ベースのアプリケーションには便利な場合があります。ただし、この方法では、5250 エミュレーション・セッションでユーザーがコマンドを入力できないようにすれば、ユーザーは、システム管理者が提供したプログラムとメニューを使用しなければデータにアクセスできないものと想定されています。したがって、コマンド行セキュリティーは絶対に安全とは言いきれません。iSeries^(TM) ポリシーおよびアプリケーション管理を使用することによってセキュリティーをより堅固なものにし、オブジェクト・レベルでの権限を採用して、さらにセキュリティーを厳しくします。

iSeries^(TM) Access for Windows^(R) のポリシーは、読み取り専用の特定データ・ソースへの ODBC アクセスを制限することができます。そして、iSeries ナビゲーターのアプリケーション管理機能は、ODBC アクセスを防止することができます。

追加情報については、「IBM^(R) 機密保護解説書」  を参照してください。

ユーザー出口プログラム

システム管理者は、ユーザー出口プログラムにより、IBM 提供のホスト・サーバー・プログラムを保護することができます。iSeries Access ODBC ドライバーは、データベース・ホスト・サーバーを使用します。出口点は QIBM_QZDA_INIT、QIBM_QZDA_NDBx、および QIBM_QZDA_SQLx です。ODBC ドライバー、および iSeries Access for Windows データ・アクセス・メソッド (OLE DB など) の中には、他のホスト・サーバーを使用するものもあります。

ジャーナル

ジャーナリングは、クライアント/サーバー・アプリケーションでコミットメント制御機能としてよく使われます。ジャーナルには、ジャーナル対象のファイルに加えられたすべての更新に関する詳細情報が含まれています。ジャーナル情報は、たとえば次のような特定情報を戻すようにフォーマットすることも、照会の対象にすることもできます。

- ファイルを更新したユーザーのプロファイル
- 更新されたレコード
- 更新のタイプ

ジャーナリングでは、ユーザー定義のジャーナル項目も使用できます。これをユーザー出口プログラムまたはトリガーで使用すると、比較的低いオーバーヘッドで、ユーザー定義の監査を保持する方法となります。

詳しくは、「バックアップおよび回復の手引き」  を参照してください。

データ・ソース名 (DSN) の制限

iSeries Access ODBC ドライバーは、データベースへのアクセスを読み取り専用にするための DSN 設定をサポートしています。iSeries Access ODBC ドライバーは、読み取り専用および読み取り呼び出し (read-call) のデータ・ソース設定をサポートしています。これらの設定は、安全とは言えませんが、誤って削除や更新を行うことを防止するためには役立ちます。

ODBC プログラム・セキュリティー・ストラテジー

以下の ODBC プログラム・セキュリティー・ストラテジーを検討してみてください。

データベースへのプログラム・アクセスの制限

システム管理者は、特定のファイル、または特定のプログラム (1 つまたはセット) へのアクセスを制限しなければならないことがよくあります。文字ベース・インターフェースを使用しているプログラマーは、プログラム借用権限を使用して制限を設定する場合があります。これと同じ方法を ODBC にも利用できません。

ODBC プログラマーは、ストアード・プロシージャーを使用して、プログラム借用権限をインプリメントすることができます。プログラマーは、ユーザーが Microsoft^(R) Access や Lotus^(R) 1-2-3^(R) などのデスクトップ・アプリケーションを使用してデータベースのファイルを操作することを、制限したい場合があります。つまり、データベースを更新できるのはプログラマーのアプリケーションのみに制限したいことがあります。この制限をインプリメントするには、オブジェクト・レベルのセキュリティーまたはユーザー出口プログラムによって、データベースへのユーザー・アクセスを制限する必要があります。そのためには、アプリケーションはストアード・プロシージャーにデータ要求を送り、ストアード・プロシージャーがデータベースを更新するようにプログラミングします。

ユーザー別の CPU 使用の制限

ODBC の登場により、iSeries^(TM) のデータへのアクセスが非常に簡単になりました。しかし、これには欠点が 1 つあります。それは、ユーザーが、自分では気付かずに、きわめて CPU 使用率の高い照会を作成してしまうことがあるということです。ODBC は対話式ジョブの優先順位で実行されるため、これはシステムのパフォーマンスに重大な影響を与えるおそれがあります。iSeries は照会管理プログラムをサポートしています。ODBC は、ストアード・プロシージャー呼び出しの中で、たとえば PC アプリケーションを介して照会管理プログラムを起動することができます。あるいは、ODBC API で、照会タイムアウト・パラメーターを使用して照会管理プログラムを起動することもできます。ユーザー出口プログラムにより、ODBC ジョブで照会管理プログラムを強制実行することもできます。制限時間は、CHGQRYA CL コマンドの QRYTIMLMT パラメーターに指定します。また、照会オプション・ファイル (QAQQINI) を使用してこの値を設定することもできます。

追加情報については「SQL リファレンス」に収められています。オンラインの DB2 Universal Database^(TM) for iSeries SQL 資料で、この資料の HTML オンライン・バージョンを読むか、または PDF バージョンを印刷してください。

また、27 ページの『ホスト・サーバーの管理』を参照してください。

監査ログ (セキュリティーのモニター)

セキュリティーをモニターするために使用できるログがいくつかあります。QHST (ヒストリー・ログ) には、システムに対して行われたセキュリティー変更に関連したメッセージが記録されています。セキュリティー関連の機能の詳細なモニターを行うには、QAUDJRN を使用できます。*SECURITY 値に次の機能が記録されます。

- オブジェクト権限に対する変更
- ユーザー・プロファイルの作成、変更、削除、表示、および復元操作
- オブジェクト所有権に対する変更
- 所有者のプロファイルを借用するプログラム変更 (CHGPGM)
- システム値およびネットワーク属性の変更
- サブシステム・ルーティングの変更
- DST により QSECOFR パスワードが出荷時の値にリセットされた場合
- DST セキュリティー担当者のパスワードをデフォルトにすることが要求された場合
- オブジェクトの監査属性の変更

追加情報については、「IBM^(R) 機密保護解説書」  を参照してください。

ODBC セキュリティーの関連情報

セキュリティに関する詳しい説明や、上記のストラテジーをインプリメントするためのサポートが必要な場合は、IBM^(R) Consultline (1-800-274-0015) をご利用ください。特定のトピックに関する詳細情報については、下記を参照してください。

- 27 ページの『ホスト・サーバーの管理』
- IBM 機密保護解説書 
- バックアップおよび回復の手引き 
- DB2 Universal Database^(TM) for iSeries^(TM) SQL 解説書
- 以下の指示に従って、**Client Access ODBC and OLE DB Security Issues Technical Reference** にアクセスします。▶▶
 - www.ibm.com/servers/eserver/iseries/support にアクセスする。
 - 「**Find it fast!** → **Search Technical databases**」に進む。
 - 検索条件として資料名 (Client Access ODBC and OLE DB Security Issues) を入力する。◀◀

ODBC のトラブルシューティング

次に示すトピックは、iSeries^(TM) Access for Windows^(R) ODBC エラーを検出し解決するための一般的なガイドラインを提供します。

- 『ODBC 診断およびパフォーマンス・ツール』
- 20 ページの『iSeries Access ODBC のエラー・メッセージ』
- iSeries サーバー接続のトラブルシューティング
- 一般的な ODBC エラー
- IBM^(R) サポートに関する情報の収集

ODBC 診断およびパフォーマンス・ツール

次の表は、クライアントおよびサーバーの両サイドのための ODBC 診断およびパフォーマンス・ツールを示しています。

クライアント・サイドのツール

ODBC トレース (SQL.LOG)	Microsoft ^(R) の ODBC アドミニストレーターは、アプリケーションからの ODBC API 呼び出しをトレースするための独自のトレース・ユーティリティを提供します。 詳しくは、「ODBC トレースの収集 (SQL.LOG)」を参照してください。
ODBC トレース・ユーティリティ	ODBC トレース (SQL.LOG) より強固な他の ODBC トレース・ユーティリティも使用できます。それらの市販されているユーティリティは、ODBC API 呼び出しの入り口点および出口点に関する詳細なトレース機能を提供できます。2 つのトレース・ユーティリティには、Trace Tools (Dr. DeeBee) および SST Trace Plus (Systems Software Technology) などがあります。
CWBPING	CWBPING を使用するには、コマンド・プロンプトで、 <code>cwbping</code> に続けてシステム名または IP アドレスを入力します。例: <code>cwbping testsys1</code> または <code>cwbping 127.127.127.1</code> CWBPING は、サーバーとそれらの状況のリストを表示して応答します。CWBPING を使用する場合のヘルプを表示するには、パラメーターを指定せずに CWBPING を実行します。CWBPING について詳しくは、「22 ページの『サーバー状況の検査』」を参照してください。
CWBCOTRC	CWBCOTRC を使用するには、¥Program Files¥IBM¥Client Access ディレクトリーの下で、コマンド・プロンプトから CWBCOTRC ON とタイプ入力します。トレースに戻ってから、アプリケーションを開始できます。 CWBCOTRC OFF とタイプ入力すると、トレースが停止します。CWBCOTRC は、サーバーとの間で送受信されるデータに関する情報を収集します。CWBCOTRC を使用する場合のヘルプを表示するには、パラメーターを指定せずに CWBCOTRC を実行します。
詳細トレース	詳細トレースは、使用されている iSeries ^(TM) Access for Windows ^(R) コンポーネントによってトレースされた情報を収集します。このトレースで見つけられる ODBC 情報には、ドライバーへの入り口点、事前開始ジョブについての情報、使用中のパッケージ名、および特殊なエラー条件が含まれます。詳しくは、「詳細トレースの収集」を参照してください。

サーバー・サイドのツール

通信トレース	通信トレース機能は、回線記述を持つ通信タイプ (トークンリングおよびイーサネット) をトレースし、フォーマットします。 これは、多くの問題を明確に分離するためのツールです。このツールは、パフォーマンスの低下がどこで発生しているかを診断するためにも便利です。1 つの要求を処理するための所要時間を測定するには、タイム・スタンプ・フィールドと目印フィールドを使用します。
--------	--

<p>ジョブ・トレース</p>	<p>ジョブ・トレースは、ホストに関するほとんどの問題およびパフォーマンスに関する多くの問題を明確にするために役立ちます。まず、トレースするジョブでサービス・ジョブを開始しておく必要があります。そのためには、ODBC ジョブの完全修飾ジョブ名を見つけます。5250 エミュレーション・セッションから、STRSRVJOB コマンドを使用して、この QZDASOINIT ジョブでサービス・ジョブを開始します。そして、必要な情報に応じて、次の 2 つのトレースのいずれかを選択します。</p> <p>ジョブ・トレース ホスト・サーバーが実行した内部呼び出しをトレースします。これを行うには、TRCJOB *ON コマンドを実行します。</p> <p>デバッグ・トレース アプリケーションのパフォーマンスを調べ、特定の問題の原因を究明するために使用されます。</p> <p>STRDBG コマンドは、活動状態のサービス・ジョブに対して実行されます。このコマンドは、照会最適化プログラムが行った決定を、デバッグ・セッションのジョブ・ログに記録します。たとえば、推定照会時間、使用されるアクセス・パス、およびカーソル・エラーを記録します。</p> <p>STRDBG を使用可能にする簡単な方法は、「診断」タブで「デバッグの開始 (STRDBG) コマンドを使用可能にする」オプションを選択することにより、ODBC アドミニストレーターを介してご使用の ODBC DSN を構成することです。または、以下のコマンドを実行できます。</p> <p style="text-align: center;">STRDBG UPDPROD(*YES)</p> <p>ODBC ジョブ・ログには、iSeries サーバーで発生したすべてのエラーを記録することができます。ジョブがデバッグ・モードになっているときは、ジョブ・ログにはパフォーマンス関連の情報も含まれます。</p>
<p>パフォーマンス・ツール</p>	<p>パフォーマンス・ツールキットが提供する報告書とユーティリティを使用することにより、アプリケーションのパフォーマンスに関する徹底的な分析を行うことができます。このツールキットは、CPU の使用状況、ディスク・アームの使用状況、メモリー・ページング、およびその他多くの情報を提供します。基本オペレーティング・システムにはパフォーマンス・データを収集する機能が含まれていますが、その結果を分析するには、ライセンス・プログラム Performance Tools for iSeries を別途入手する必要があります。</p> <p>Database Monitor および Visual Explain も、この目的のためのツールとして使用できます。詳しくは、iSeries ナビゲーターのオンライン・ヘルプを参照してください。</p>
<p>QZDASOINIT ジョブ・ログ</p>	<p>最善のサポートを受けるためには、QZDASOINIT ジョブ・ログを生成、検出、および検索するようにしてください。ジョブ・ログには、ODBC から戻されたエラーを判別し解決するために役立つメッセージが含まれていることがあります。</p> <p>ジョブ・ログにアクセスする簡単な方法は、「診断」タブで「切断時にジョブ・ログを印刷する」オプションを選択することにより、ODBC アドミニストレーターを介してご使用の ODBC DSN を構成することです。ジョブ・ログを検出するには、PC5250 エミュレーション・セッションをオープンし、WRKSPLF コマンドを実行します。ODBC 接続に使用された iSeries ユーザー・プロファイルを WRKSPLF コマンドのユーザー・パラメーターとして指定します。</p>

QAQQINI (照会オプション・ファイル)	Query オプション・ファイルのライブラリーを設定するには、ご使用の ODBC DSN を ODBC アドミニストレーター を介して構成し、 診断 タブを選択します。Query オプション・ファイル・ライブラリー・ボックスで使用したいライブラリーの名前を入力します。
------------------------	--

iSeries Access ODBC のエラー・メッセージ

エラーが発生すると、iSeries^(TM) Access ODBC ドライバーは、SQLSTATE (ODBC エラー・コード) とエラー・メッセージを戻します。このドライバーは、それ自体が検出したエラーと、DBMS が戻したエラーの両方から、この情報を入手します。

データ・ソース内で発生したエラーの場合は、iSeries Access ODBC ドライバーは、戻されたネイティブ・エラーを該当の SQLSTATE にマップします。iSeries Access ODBC ドライバーと Microsoft^(R) Driver Manager の両方が同じエラーを検出した場合は、それぞれ該当の SQLSTATE を生成します。iSeries Access ODBC ドライバーは、DBMS が戻したメッセージに基づいてエラー・メッセージを戻します。

iSeries Access ODBC ドライバーまたは Microsoft Driver Manager 内で発生したエラーの場合は、iSeries Access ODBC ドライバーは、SQLSTATE に関連付けられているテキストに基づいてエラー・メッセージを戻します。

エラー・メッセージの形式

エラー・メッセージの形式は次のとおりです。

[ベンダー][ODBC コンポーネント][データ・ソース]
エラー・メッセージ

大括弧 ([]) の中の接頭部は、エラーの発生源を示します。次の表は、iSeries Access ODBC ドライバーが戻すこれらの接頭部の値を示しています。

データ・ソース内でエラーが発生した場合は、接頭部 [ベンダー] および [ODBC コンポーネント] に、データ・ソースからエラーを受け取った ODBC コンポーネントのベンダーと名前が示されます。

エラーの発生源	値
ドライバー・マネージャー	[Microsoft] [ODBC Driver Manager] [N/A]
iSeries Access ODBC ドライバー	[IBM ^(R)] [iSeries Access ODBC driver] N/A
NLS メッセージ	[IBM] [iSeries Access ODBC driver] Column #: NLS エラー・メッセージ番号 NLS エラー・メッセージ・テキスト

エラーの発生源	値
通信層	[IBM] [iSeries Access ODBC driver] Communications link failure.Comm RC=xxxx - (message text) ここで、xxxx は 16 進形式ではなく 10 進形式のエラー番号です。 エラーの性質を示すメッセージ・テキストと、 エラー番号が表示されます。 注: エラー・メッセージ ID について詳しくは、「iSeries Access の戻りコード」 またはオンラインの「iSeries Access for Windows ユーザーズ・ガイド (3ペー ジ)」を参照してください。
DB2 ^(R) UDB for iSeries	[IBM] [iSeries Access ODBC driver] [DB2 UDB] サーバー・エラー・メッセージ

DB2^(R) UDB for iSeries^(TM) エラー・メッセージ・テキストを表示する

エラーの先頭の文字	使用する CL コマンド
SQL	DSPMSGD RANGE(SQLxxxx) MSGF(QSQLMSG)
IWS または PWS	DSPMSGD RANGE(ZZZxxxx) MSGF(QIWS/QIWSMSG)。ZZZ は IWS または PWS。

他の ODBC エラー・メッセージに関するヘルプについては、24 ページの『一般的な ODBC エラー』を参照してください。

NLS メッセージまたは通信エラー・メッセージは、オンラインの「iSeries^(TM) Access for Windows^(R) ユーザーズ・ガイド (3ページ)」や、サービス、エラー、およびトレースの各メッセージのヘルプ・トピックを使用して、検索し表示することができます。

iSeries サーバー接続のトラブルシューティング

各 ODBC 接続では、iSeries^(TM) サーバーで実行される 1 つのデータベース・サーバー・プログラムとの通信が行われます。このプログラムは、**ホスト・サーバー・プログラム**と呼ばれます。TCP/IP で使用されるデータベース・サーバー・プログラムの名前は、**QZDASOINIT** です。これは通常、サブシステム QUSRWRK 内にありますが、システム管理者が別の場所にセットアップすることも可能です。

通常の場合では、このプログラムは透過的に呼び出されるので、ユーザーは、適切なサブシステムおよび通信プロトコルが実行されていることを確認する以外には、何もする必要はありません。ホスト・サーバー・ジョブの管理に関する詳細は、27 ページの『ホスト・サーバーの管理』を参照してください。

接続障害を示す最も一般的なものは、通信リンク障害が発生したことを示す ODBC ドライバーからのエラー・メッセージです。

ODBC が iSeries サーバーに接続できないときは、トラブルシューティングのために次のような作業を行ってください。

- 22 ページの『サーバー状況の検査』
- 22 ページの『サブシステムが活動状態であることを確認する』

- 『事前開始ジョブが実行されていることを確認する』
- 23 ページの『TCP/IP に関するその他の考慮事項』

サーバー状況の検査: iSeries^(TM) Access for Windows^(R) 製品には、ホスト・サーバーの状況を検査するための特別のコマンドがあります。

```
CWBPING systemname
```

上記の systemname はシステムの名前です。

このコマンドは次のような検査結果を戻します。

```
To cancel the CWBPING request, press CTRL-C or CTRL=BREAK
I - Verifying connection to system MYSYSTEM...
I - Successfully connected to server application: Central Client
I - Successfully connected to server application: Network File
I - Successfully connected to server application: Network Print
I - Successfully connected to server application: Data Access
I - Successfully connected to server application: Data Queues
I - Successfully connected to server application: Remote Command
I - Successfully connected to server application: Security
I - Successfully connected to server application: DDM
I - Successfully connected to server application: Telnet
I - Successfully connected to server application: Management Central
I - Connection verified to system MYSYSTEM
```

注:

- ODBC が機能するためには、データベース・サーバーとセキュリティー・サーバーが作動可能でなければなりません。
- 接続で SSL が使用されるように構成されていることを示すメッセージが表示された場合は、その接続は 32 ビット・アプリケーションでしか使用できません。64 ビット用 iSeries Access ODBC ドライバーまたは 64 ビット用 iSeries Access OLE DB provider による接続を利用しようとすると失敗します。64 ビット・アプリケーションを使用して iSeries サーバーに正常に接続できるようにするために大事なことは、はじめに、接続を SSL を使用しないように構成することです。

サブシステムが活動状態であることを確認する: TCP/IP 接続の ODBC ジョブ (QZDASOINIT) は QUSRWRK サブシステムの中で実行されます。このサブシステムが実行されていることを確認してください。QSERVER サブシステムは手動で始動しなければならない場合があります。そのためには次のコマンドを実行するだけです。

```
STRSBS QSERVER
```

サブシステムが IPL 時に自動的に始動されるようにするには、STRSBS QSERVER コマンドを組み込むように IPL 始動プロシージャ (デフォルトは QSYS/QSTRUP) を変更します。

サブシステム QSERVER の他に、サブシステム QSYSWRK および QUSRWRK も実行されていることが必要です。

事前開始ジョブが実行されていることを確認する: ジョブの初期化と開始の時点でのパフォーマンスを高めるために、出荷時の設定では、QSERVER/QUSRWRK サブシステムは事前開始ジョブを使用するように構成してあります。このサブシステムに事前開始ジョブが構成されている場合は、そのジョブが活動状態になっていなければ接続できません。TCP/IP 接続用に使用される事前開始ジョブは次のとおりです。

- QZDASOINIT - サーバー・プログラム

事前開始ジョブが実行されていることを確認するには、以下のいずれかを使用します。

```
WRKACTJOB SBS(QUSRWRK)
```

```
WRKACTJOB SBS('user-defined-subsystem)
```

適切な事前開始ジョブが活動状態になっていることが必要です。

Job	User	Type	-----Status-----	
QZDASOINIT	QUSER	PJ	ACTIVE	(socket connection)
QZDASRVSD	QUSER	PJ	ACTIVE	(socket connection)

接続がすでに活動状態になっていない限り、事前開始ジョブは WRKACTJOB に表示されません。WRKACTJOB パネルで「F14 - 組み込み」を使用する必要があります。

TCP/IP に関するその他の考慮事項: 次のコマンドを使用して、TCP/IP が開始されていることを確認します。

```
NETSTAT *CNN
```

注: iSeries[™] ナビゲーターで TCP/IP が開始済みであることを確認するには、ご使用のサーバーが TCP/IP で構成済みでなければなりません。そうであれば、以下の操作を行ってください。

1. iSeries ナビゲーターで、ご使用のサーバーを選択する。 → 「ネットワーク」
2. 「TCP/IP 構成」を右マウス・ボタンでクリックし、「ユーティリティ」を選択する。
3. 「Ping」を選択する。
4. ホスト名または TCP/IP アドレスを指定して、「Ping する (Ping Now)」をクリックする。

必要なプロトコルがまだ実行されていない場合は、STRTCP コマンドを使用してそのプロトコルを開始します。

NETSTAT *CNN コマンドから戻された情報をブラウザして、必要なデーモンがすべて実行されていることを確認します。

リモート・アドレス	リモート・ポート	ローカル・ポート	アイドル時間	状態
*	*	as-cent >	000:09:31	接続待機
*	*	as-signon	000:09:41	接続待機
*	*	as-svrmap	002:57:45	接続待機
*	*	as-data >	002:57:45	接続待機

必要があれば、STRHOSTSVR SERVER(*ALL) コマンドを使用してデーモンを始動します。

- QZDASRVSD (ODBC ソケット・デーモン) が実行されていることを確認する。
 - as-database は listen 状態になっていることが必要です。
 - WRKJOB QZDASRVSD を使用して、このデーモンのジョブ・ログにエラー・メッセージが記録されていないかどうか調べます。
- ソケット・デーモン QZSOMAPD が QSYSWRK サブシステム内で実行されていることを確認する。
 - as-svrmap は listen 状態になっていることが必要です (この状態は NETSTAT *CNN により示されます)。
 - WRKJOB QZSOMAPD を使用して、このデーモンのジョブ・ログにエラー・メッセージが記録されていないかどうか調べます。

PC は、サーバー・マップ・ソケットに接続することにより、データベース・サーバーが使用するソケットを見つけます。PC は、as-database により使用されているソケットを検索します。次に、PC は、ファ

イル・サーバー・デーモン QZDASRVSD がモニターしている適切なソケットに接続します。サーバー・デーモンは、クライアントの接続を、QUSRWRK 内の QZDASOINIT 事前開始ジョブの 1 つに結合させます。ユーザー・プロファイルとパスワードの検査が終わり、ユーザー・プロファイルがその事前開始ジョブにスワッピングされてしまうと、そのジョブは QZDASOINIT ジョブと同様に実行されます。これが、この PC から行われた最初の接続である場合は、別のもう 2 つのサーバーが使用されます。それは、ライセンスを交付するセントラル・サーバーと、ユーザー ID とパスワードの検査を行うサインオン・サーバーです。

TCP/IP が開始済みであるかどうかを検査する方法については、「TCP/IP の一般的な問題」を参照してください。

一般的な ODBC エラー

次に示すトピックは、iSeries^(TM) Access for Windows^(R) ODBC エラーを検出し解決するための一般的なガイドラインを提供します。

- SQL エラー
- ストアード・プロシージャ・エラー
- ODBC の誤った出力および予測不能エラー

SQL エラー:

- SQL0104 - トークン &1 が正しくない。有効なトークンは &2
- SQL0113 - 名前 &1 は許されない。
- SQL0114 - リレーショナル・データベース &1 が現行サーバー &2 と同じでない。
- SQL0204 - MYSYSCONF が見つからない。
- SQL0208 - ORDER BY 列が結果テーブルにない。
- SQL0900 - アプリケーション・プロセスが接続状態にない。
- SQL0901 - SQL システム・エラー
- SQL5001 - 列修飾子またはテーブル &2 が未定義である。
- SQL5016 - オブジェクト名 &1 が命名規則として正しくない。
- SQL7008 &2 の &1 は操作では無効である。理由コードは 3 です。

注: SQL エラーについて詳しくは、「SQL メッセージおよびコード」を参照してください。

ストアード・プロシージャ・エラー: 一般的なストアード・プロシージャ・エラーには、次のものがあります。

- 『SQL0444 - &B の外部プログラム &A が見つからない (DB2 UDB for iSeries SQL)』
- 25 ページの『OUTPUT パラメーターおよび INPUT_OUTPUT パラメーターについてデータが戻されない』
- 25 ページの『SQL0501 - カーソル CRSR000x がオープンしていない』

SQL0444 - &B の外部プログラム &A が見つからない (DB2 UDB for iSeries SQL): SQL0444 が生成されるのは、execute (実行) または execute direct (直接実行) で、データベース・サーバーがプロシージャ宣言は検出できたが、プログラム・オブジェクトを検出できなかった場合です。外部プログラムは、システム・カタログ・テーブル内で指定された位置になければなりません。この位置は、プロシージャが呼び出されたときではなく、プロシージャが (CREATE PROCEDURE を使用して) 定義された時点で有効であった命名規則とデフォルト・コレクションによって定義されるという点に注意してください。ストアード・

プロシージャの外部プログラム名用に定義されている位置を調べるには、QSYS2.SYSPROCS に対する照会を実行し、「EXTERNAL_NAME」名前フィールドの値を確認してください。

OUTPUT パラメーターおよび INPUT_OUTPUT パラメーターについてデータが戻されない: この問題の原因としては、次のいずれかが考えられます。

- ODBC **SQLBindParameter** API で、**fParamType** に誤って **SQL_PARAM_INPUT** が指定されている。
- **CREATE PROCEDURE** の代わりに **DECLARE PROCEDURE** が使用されているが、拡張動的サポートが使用不可にされている。
- プログラマーが、**CREATE** または **DECLARE PROCEDURE** のパラメーターに誤って **IN** を宣言した。
- ストアード・プロシージャ・プログラムが誤ってこのパラメーターを戻した。

SQL0501 - カーソル CURSOR000x がオープンしていない: ILE プログラム内の 組み込み SQL を使用しているときにデータを戻すには、コンパイル・オプションに、デフォルトの *NEW を使用せずに、ACTGRP(*CALLER) を指定する必要があります。

プログラムが、終了処理ではなく、戻り処理を実行することを確認してください。

ストアード・プロシージャ・プログラムが、戻り処理でなく終了処理を実行する場合は、「**SQL カーソルをクローズ (Close SQL Cursor)**」オプションを *ENDACTGRP に設定する必要があります。「**SQL カーソルをクローズ (Close SQL Cursor)**」オプションが *ENDMOD に設定されている場合は、カーソルはデータが検索される前にクローズされます。

CREATE PROCEDURE に正しい結果セット数が指定されていることも確認してください。これは、配列結果セットを使用している場合に特に重要です。

ODBC の誤った出力および予測不能エラー: iSeriesTM Access ODBC ドライバーとデータベース・サーバー・プログラムのコード・レベルが一致していることを確認します。また、オーダーした PTF について、またはサービス・バックの readme.txt ファイルの中で、相互に必要な PTF 要件をチェックしてください。それでも問題が再発する場合は、ODBC データ・ソースの中でプリフェッチ・オプションを使用不可にしてあることを確認します。アプリケーションで **SQLExtendedFetch** または **SQLFetchScroll** ODBC API のどちらかを使用している場合、またはこれらを使用しているかどうかははっきりしない場合は、プリフェッチ・オプションは使用しないでください。

ストアード・プロシージャの結果セット・カーソル は、前進のみの読み取り専用であるという点に注意してください。

ASCII 文字でなく 2 進または 16 進データの場合

「**Translation (変換)**」パラメーターのデフォルト値は、2 進データ (CCSID 65535) をテキストに変換しないように設定されています。CCSID は、データ変換に使用される変換テーブルを識別するために、ファイル、テーブル、および場合によってはフィールド (列) にまで付加されます。65535 の CCSID は、多くの場合、ビットマップ・グラフィックスなどのように、言語から独立した生データ (2 進または 16 進) を識別します。生データを損傷しないようにするためには、2 進データ (CCSID 65535) をテキストに変換することを選択しないでください。

変換パラメーターを「2 進データ (CCSID 65535) をテキストに変換する (Convert binary data (CCSID 65535) to text)」に設定すると、データに付加されている CCSID がジョブの CCSID に更新されます。データが真に 2 進データである場合は、このパラメーター設定はデータの損傷の原因になることがあります。

IBM サポートに提供する情報の収集

IBM サポートに対して問題記録をオープンする際には、IBM^(R) サポート・スタッフが最良のサービスを提供できるようにするために、必要な情報を用意しておいてください。この情報を収集するには次のようにしてください。

OS/400 ^(R) のバージョンと累積 PTF レベルを記録します。	<ol style="list-style-type: none"> 1. 端末エミュレーション・コマンド行に、PTF 表示コマンドを入力する。 <p style="text-align: center;">DSPPTF</p> 2. VxRxMx の形式の OS/400 リリース情報を記録する。 3. IPL ソースが ##MACH#B であることを確認する。 4. F5 を押して PTF の詳細を表示する。 5. リスト内の最初の PTF ID を記録する。この ID の形式は Tzxyyy で、xx は年、yyy はユリウス日付 (年間通算日)、そして z は L または C です。
ODBC ドライバーのバージョンを記録します。	<ol style="list-style-type: none"> 1. タスクバーから、「スタート → プログラム → IBM iSeries^(TM) Access for Windows^(R) → ODBC 管理 (ODBC Administration)」と選択する。 <p>注：64 ビット・ドライバーを使用する 64 ビット・マシンでは、「ODBC 管理 (64 ビット)(ODBC Administration (64-bit))」を選択してください。</p> 2. 「ドライバー」タブを選択する。 3. iSeries Access ODBC ドライバーのバージョンを記録する。
ODBC ドライバー・マネージャーのバージョンを記録します。	<ol style="list-style-type: none"> 1. タスクバーから、「スタート → プログラム → IBM iSeries Access for Windows → ODBC 管理 (ODBC Administration)」と選択する。 <p>注：64 ビット・ドライバーを使用する 64 ビット・マシンでは、「ODBC 管理 (64 ビット)(ODBC Administration (64-bit))」を選択してください。</p> 2. 「バージョン情報」タブを選択する。 3. ドライバー・マネージャーのバージョンを記録する。
トレースを収集します。	<p>サポートのために最も要求される可能性のあるトレースは ODBC トレース (SQL.LOG)、CWBCOTRC または通信トレース、および詳細トレースです。トレースについての詳細は、「17 ページの『ODBC 診断およびパフォーマンス・ツール』」を参照してください。</p>
追加情報を記録します。	<p>PC アプリケーション、エラーの説明、およびどのような ODBC ドライバーを使用しているか (32 ビットまたは 64 ビット)。</p>

ホスト・サーバーの管理

このトピックでは、iSeries^(TM) サーバーで実行されるサーバー機能について簡単に説明すると共に、iSeries^(TM) Access for Windows^(R) 製品で使用されるホスト・サーバーに固有の技術情報を提供します。これらは、iSeries Access for Windows が使用するサーバーのすべてを網羅しているわけではなく、また、このトピックでホスト (iSeries) システム上のすべてのサーバーを紹介するわけではありません。

OS/400^(R) ホスト・サーバー

ホスト・サーバーは、クライアントの PC またはデバイスからの要求を取り扱います。要求には、アプリケーションの実行、データベースの照会、文書の印刷の他、バックアップまたはリカバリー手順の実行までも含まれます。iSeries は、ファイル、データベース、アプリケーション、メール、印刷、ファクシミリ、および無線通信などを網羅する多数のタスクを同時に行うことのできるフル・ファンクション型サーバーです。このようなタスクを複数の異なるサーバーにより取り扱う場合は、サーバーの管理と調整が複雑になります。すべてのサーバーをまとめて 1 つの統合システムにすれば、大幅にネットワーク管理の総費用と複雑さを軽減することができます。

これらのサーバーは iSeries Access for Windows で使用されるものですが、他のクライアント製品でも使用できるように設計されています。このトピックでは、iSeries Access for Windows でこれらのサーバーをどのように使用するかに重点を置いて説明します。

OS/400 ホスト・サーバー・オプションの追加または削除

ここで説明する OS/400 サーバーはすべて最適化されたサーバーであり、OS/400 の基本オプションに含まれています。iSeries Access for Windows の iSeries ナビゲーター機能を使用するには、ホスト・サーバー・オプションをインストールします。

iSeries Access for Windows または iSeries NetServer 製品を使用していないために OS/400 Host Server オプションを削除したいときは、これらのサーバーが使用しているサブシステムをすべて終了してから、このオプションを削除してください。終了するサブシステムは、QBASE または QCMN サブシステム (APPC サポート付きのホスト・サーバーの場合)、QSYSWRK および QUSRWRK サブシステム (ソケット・サポート付きのホスト・サーバーの場合)、および QSERVER サブシステム (データベースおよびファイル・サーバーの場合) です。これらのサブシステムが活動状態のときにこのオプションを削除しようとすると、問題が発生することがあります。

28 ページの『OS/400 ホスト・サーバー』

このトピックでは、iSeries Access for Windows クライアントでよく使われる多くのホスト・サーバーと、それぞれに関連したオブジェクトについて説明します。サーバーは、タイプ別に見ることも、iSeries Access for Windows でのそれぞれの機能別に見ることもできます。

39 ページの『OS/400 ホスト・サーバーの使用』

このトピックでは、クライアント/サーバー通信プロセスと、それを管理する方法について説明します。さらに、関連の iSeries システム値およびサブシステムのリストを示し、iSeries でサーバー・ジョブを識別、表示、および管理する方法についても説明します。

63 ページの『サーバー出口プログラムの使用』

このトピックでは、出口プログラムを作成し登録する方法について説明します。出口プログラムのパラメーターとプログラミング例も示されています。

OS/400 ホスト・サーバー

この情報は、iSeries^(TM) Access for Windows で使用されるサーバーのみを対象としています。ホスト (iSeries) システム上のすべてのサーバーを対象としているものではありません。iSeries Access for Windows ホスト・サーバーには、次のものがあります。

『iSeries Access for Windows 機能別のホスト・サーバー』

ホスト・サーバーを、iSeries Access for Windows に関連する機能別に分類すると、次のようになります。

30 ページの『ファイル・サーバー』

ファイル・サーバーを利用するクライアントは、iSeries サーバー上にあるファイルやプログラムなどの情報の保管とアクセスを行うことができます。

31 ページの『データベース・サーバー』

これは、データ転送、ODBC、iSeries ナビゲーター・データベース、SQL API (DB API)、および iSeries Access for Windows プロバイダー (OLE DB および .NET Data Provider) のためのものです。

36 ページの『データ待ち行列サーバー』

iSeries サーバー上のデータ待ち行列にアクセスできるようにします。

36 ページの『ネットワーク・プリント・サーバー』

リモート・プリント・サポートおよび追加のプリント管理機能を提供します。

37 ページの『セントラル・サーバー』

ライセンス管理およびその他のクライアント管理機能などのサービスを提供します。

37 ページの『リモート・コマンドおよび分散プログラム呼び出しサーバー』

PC アプリケーションが、OS/400^(R) でコマンドを実行してプログラムを呼び出し、その結果をクライアントに戻すことができるようにします。

38 ページの『サインオン・サーバー』

ソケット・サポート付きのホスト・サーバーにパスワード管理機能を提供します。

38 ページの『サーバー・ポートマッパー』

接続を要求しているクライアントに現行のサーバー・ポートの番号を提供します。



38 ページの『拡張動的リモート SQL サーバー (QXDAEDRSQL)』

リモート SQL アクセスおよびその他のデータベース機能をサポートします。

39 ページの『DRDA/DDM サーバー』

DB2^(R) UDB for iSeries に搭載されている機能にアクセスできます。 <<

iSeries Access for Windows 機能別のホスト・サーバー

次の表は、iSeries^(TM) Access for Windows^(R) のいくつかの機能で使用されるサーバーのサブセットです。



クライアントの機能	使用される OS/400 ^(R) サーバー
.NET Data Provider	<ul style="list-style-type: none"> • 31 ページの『データベース・サーバー』 • 38 ページの『サインオン・サーバー』 • 37 ページの『セントラル・サーバー』 • 38 ページの『拡張動的リモート SQL サーバー (QXDAEDRSQL)』
IBM ^(R) Toolbox for Java ^(TM)	<ul style="list-style-type: none"> • 38 ページの『サインオン・サーバー』 • 37 ページの『セントラル・サーバー』 • 30 ページの『ファイル・サーバー』 • 31 ページの『データベース・サーバー』 • 39 ページの『DRDA/DDM サーバー』 • 36 ページの『データ待ち行列サーバー』 • 37 ページの『リモート・コマンドおよび分散プログラム呼び出しサーバー』 • 36 ページの『ネットワーク・プリント・サーバー』
データ転送	<ul style="list-style-type: none"> • 38 ページの『サインオン・サーバー』 • 37 ページの『セントラル・サーバー』 • 31 ページの『データベース・サーバー』
ODBC ドライバー	<ul style="list-style-type: none"> • 38 ページの『サインオン・サーバー』 • 31 ページの『データベース・サーバー』
iSeries ナビゲーターからの統合ファイル・システムへのアクセス	30 ページの『ファイル・サーバー』
データ待ち行列 API	36 ページの『データ待ち行列サーバー』
OLE DB Provider	<ul style="list-style-type: none"> • 36 ページの『データ待ち行列サーバー』 • 31 ページの『データベース・サーバー』 • 37 ページの『リモート・コマンドおよび分散プログラム呼び出しサーバー』 • 38 ページの『サインオン・サーバー』
拡張動的リモート SQL サーバー (QXDAEDRSQL)	<ul style="list-style-type: none"> • 38 ページの『サインオン・サーバー』 • 37 ページの『セントラル・サーバー』 • 38 ページの『拡張動的リモート SQL サーバー (QXDAEDRSQL)』
ライセンス管理	37 ページの『セントラル・サーバー』
<p>ライセンスを必要とするアプリケーション (データ転送および 5250 エミュレーション) が始動されるときに行われます。</p>	

クライアントの機能	使用される OS/400 ^(R) サーバー
変換マップの検索 初期接続の時点で、クライアントに必要な変換マップが含まれていない場合のみ行われます。	37 ページの『セントラル・サーバー』
リモート・コマンド機能	37 ページの『リモート・コマンドおよび分散プログラム呼び出しサーバー』
分散プログラム呼び出し	37 ページの『リモート・コマンドおよび分散プログラム呼び出しサーバー』
妥当性検査のためのパスワードの送信および有効期限切れパスワードの変更 (TCP/IP)	38 ページの『サインオン・サーバー』
ネットワーク・プリント GUI およびプログラミング・インターフェース	36 ページの『ネットワーク・プリント・サーバー』



詳しくは、「iSeries Access for Windows Servers and Ports Required」APAR II12227  を参照してください。

ファイル・サーバー

ファイル・サーバーを利用するクライアントは、iSeries^(TM) サーバー上にあるファイルやプログラムなどの情報の保管とアクセスを行うことができます。OS/400^(R) ファイル・サーバーは、iSeries サーバー上の統合ファイル・システムとのインターフェースとしての役割を果たします。クライアントは、統合ファイル・システム・ユーザー・インターフェースおよび API ではなく、それぞれ各自のインターフェースを使用してファイル・システムにアクセスします。

統合ファイル・システムは OS/400 プログラムの一部です。このファイル・システムは、パーソナル・コンピュータおよび UNIX^(R) オペレーティング・システムに似たストリーム入出力およびストレージ管理をサポートします。さらに、このファイル・システムは、iSeries サーバーに保管されているすべての情報を統合します。

統合ファイル・システムの主要機能は次のとおりです。

- ストリーム・ファイルへの情報の保管サポート。ストリーム・ファイルは、長い連続したデータのストリングを含むファイルです。このようなデータ・ストリングには、たとえば、文書のテキストやピクチャーの画素などがあります。iSeries のフォルダーに保管される文書は、ストリーム・ファイルです。ストリーム・ファイルのその他の例としては、PC ファイルや、UNIX システム内のファイルなどがあります。ストリーム・ファイル・サポートは、クライアント/サーバー・アプリケーションの中で効率的に使用できるように設計されています。
- オブジェクトを木の枝のような形で編成できるようにする階層ディレクトリー構造。オブジェクトにアクセスするには、ディレクトリーからオブジェクトへのパスを指定します。
- ユーザーおよびアプリケーションが、ストリーム・ファイル、データベース・ファイル、ドキュメント、および、iSeries サーバーに保管されているその他のオブジェクトにアクセスするために使用する共通インターフェース。

iSeries サーバーは、類似したインターフェースをもついくつかの異なるファイル・システムをサポートすることができます。ユーザーおよびアプリケーションは、ファイル・システムを使用して、ストレージ内の論理単位として編成されている特定セグメントにアクセスできます。論理単位には、ファイル、ディレクトリー、ライブラリー、およびオブジェクトがあります。

iSeries ファイル・システムのリストについては、「ファイル・システムの扱い」を参照してください。

統合ファイル・システムについて詳しくは、「統合ファイル・システム」を参照してください。

OS/400 ファイル・サーバーを利用するクライアントは、クライアント製品が提供するサポートに応じて、すべての iSeries ファイル・システムか、または QDLS のみにアクセスできます。

このサーバーには、次の表に示すプログラムが含まれています。

ファイル・サーバー・オブジェクト

プログラム名	ライブラリー	オブジェクト・タイプ	説明
QPWFSEVSO	QSYS	*PGM	サーバー・プログラム
QPWFSEVS2	QSYS	*PGM	サーバー・プログラム
QPWFSEVSD	QSYS	*PGM	デーモン・プログラム
QPWFSEVSV	QSYS	*JOB	サーバー・ジョブに使用されるジョブ記述
QPWFSEVSR	QSYS	*CLS	すべてのファイル・サーバー・ジョブおよびデータベース・サーバー・ジョブに使用されるクラス
QPWFSEVSS	QSYS	*PGM	SSL サーバー・プログラム

データベース・サーバー

クライアントは、データベース・サーバーを利用して、DB2^(R) UDB for iSeries^(TM) が備えている機能にアクセスすることができます。このサーバーは次の機能を提供します。

- リモート SQL アクセスに関するサポート
- ODBC、ADO、OLE DB、および .NET Data Provider を介したデータへのアクセス
- データベース機能 (ファイルの作成と削除およびファイル・メンバーの追加と削除など)
- システム上に存在するデータベース・ファイルに関する情報を取得するための検索機能 (SQL カタログ機能など)

さらに、データベース・サーバーと一緒に 分散リレーショナル・データベース体系^(TM) (DRDA^(R)) を使用できます。DRDA は、OLE DB または .NET Data Provider と連動しません。DRDA で次のものを使用する場合の情報については、以下のリンクを使用してください。

- SQL パッケージ
- 35 ページの『DRDA を使用するときの規則と制限』

DRDA について詳しくは、「分散データベース・プログラミング」を参照してください。

このサーバーには、次の表に示すプログラムが含まれています。

データベース・サーバーのプログラム

プログラム名	ライブラリー	説明
QZDASOINIT	QSYS	サーバー・プログラム
QZDASON2	QSYS	ソケット・セットアップ・プログラム
QZDASRVSD	QSYS	デーモン・プログラム
QZDASSINIT	QSYS	SSL サーバー・プログラム
注: QZDANDB および QZDACRTP *PGM オブジェクトは、*SRVPGM オブジェクト QZDASRV とともに、データベース・サーバーにより使用されます。		

SQL パッケージ: SQL パッケージは、アプリケーション・プログラム内の SQL ステートメントをリレーショナル・データベースにバインドします。SQL パッケージを使用すると、アプリケーションは SQL 要求に関する情報を再利用できるので、動的 SQL サポートを使用するアプリケーションのパフォーマンスが向上します。データベース・サーバーは、動的 SQL 要求を使用するアプリケーション・プログラムです。データベース・サーバーでは、使用頻度の高い SQL ステートメントのパッケージの使用をサポートし、特定のバインド情報を再利用できるようにします。

詳しくは、次のトピックを参照してください。

- 『SQL パッケージ名』
- 34 ページの『SQL パッケージのクリーンアップ』

SQL パッケージ名: データベース・サーバーは、DRDA^(R) を使用する他のリレーショナル・データベースへのゲートウェイとして使用できます。データベース・サーバーは、ターゲットのリレーショナル・データベース上に、1 つまたは複数の SQL パッケージを自動的に作成します。パッケージ名は、現在サーバーで使用されている属性に従って生成されます。

リレーショナル・データベースが iSeries^(TM) サーバーではない場合のパッケージ名。リレーショナル・データベース (RDB) が iSeries サーバーではない場合は、パッケージは、アプリケーション・サーバー上の QSQL400 と呼ばれるコレクションの中に作成されます。RDB が iSeries サーバーである場合は、パッケージはライブラリー QGPL の中に作成されます。アプリケーション・サーバーが iSeries サーバーではない場合は、パッケージ名は QZDabcde です。abcde は、使用されている特定のパーサー・オプションに対応する値です。次の表は、パッケージ名に関するオプションを示しています。

パッケージ名フィールドのオプション

フィールド	フィールドの説明	オプション
a	日付形式	<ul style="list-style-type: none"> • ISO、JIS • EUR • JUL
b	時刻形式	<ul style="list-style-type: none"> • JIS • EUR、ISO

フィールド	フィールドの説明	オプション
c	コミットメント制御/小数点区切り文字	<ul style="list-style-type: none"> • *CS/ピリオド • *CS/コンマ • *CHG/ピリオド • *CHG/コンマ • *RR/ピリオド • *RR/コンマ
d	ストリング区切り文字	<ul style="list-style-type: none"> • アポストロフィ • 引用符
e	パッケージに含めることのできる最大ステートメント数	<ul style="list-style-type: none"> • 0 から 64 • 1 から 256 • 2 から 512 • 3 から 1024

リレーショナル・データベースが iSeries サーバーである場合のパッケージ名

アプリケーション・サーバーが iSeries サーバーである場合は、パッケージ名は **QZDAabcdef** です。
abcdef は、使用されている特定のパーサー・オプションに対応する値です。

パッケージ名フィールドのオプション

フィールド	フィールドの説明	オプション
a	日付形式	<ul style="list-style-type: none"> • ISO、JIS • EUR • JUL • MDY • DMY • YMD
b	時刻形式と命名規則	<ul style="list-style-type: none"> • ISO、JIS、および SQL 命名規則 • USA および SQL 命名規則 • EUR および SQL 命名規則 • HMS および SQL 命名 • ISO、JIS およびシステム命名規則 • USA およびシステム命名規則 • EUR およびシステム命名規則 • HMS およびシステム命名規則

フィールド	フィールドの説明	オプション
c	コミット・レベルと小数点	<ul style="list-style-type: none"> • *CS/ピリオド • *CS/コンマ • *ALL/ピリオド • *ALL/コンマ • *CHG/ピリオド • *CHG/コンマ • *NONE/ピリオド • *NONE/コンマ
d	ストリング区切り文字	<ul style="list-style-type: none"> • アポストロフィ • 引用符
e	パッケージ内のセクション数	<ul style="list-style-type: none"> • 0 から 64 • 1 から 256 • 2 から 512 • 3 から 1024
f	日付と時刻の区切り文字	<ul style="list-style-type: none"> • 文字の高位ビット : • '1100'b - ISO 日付形式の 1 つ • '1101'b - 日付区切り文字にコンマを使用 • '1110'b - 日付区切り文字にピリオドを使用 • '1111'b - 日付区切り文字にコロンを使用 • 文字の低位ビット : • '0001'b - ISO 時刻形式の 1 つ • '0010'b - 時刻区切り文字にコンマを使用 • '0011'b - 時刻区切り文字にピリオドを使用 • '0100'b - 時刻区切り文字にスラッシュを使用 • '0101'b - 時刻区切り文字にダッシュを使用 • '0110'b - 時刻区切り文字にブランクを使用

SQL パッケージのクリーンアップ: DRDA^(R) 機能用に使用されるパッケージは、必要に応じて、システム上で自動的に作成されます。これらのパッケージを定期的にクリーンアップします。パッケージを削除するには、SQL パッケージの削除 (DLTSQLPKG) コマンドを使用します。

削除するのは、使用頻度の低いパッケージのみに限るようにしてください。パッケージは、必要があれば再作成されますが、同じパッケージが二度目に作成された場合は、そのパッケージのパフォーマンスは大幅に低下します。

ステートメントの命名規則: 次の表は、データベース・サーバーで決められている命名規則の要約を示しています。

ステートメントの命名規則

ステートメント	動的 SQL	拡張動的 SQL パッケージの使用
ローカル	ステートメント名は iSeries ^(TM) iSeries 命名規則に従っていなければなりません。ただし、推奨される形式は STMTxxxx です。 カーソル名は iSeries 命名規則に従っていなければなりません。	ステートメント名は iSeries 命名規則に従っていなければなりません。ただし、推奨される形式は STMTxxxx です。 カーソル名は iSeries 命名規則に従っていなければなりません。
DRDA ^(R)	ステートメント名の形式は STMTxxxx でなければなりません。 カーソル名の形式は次のとおりです。 スクロール不能カーソルの場合は CRSRyyyy で、スクロール可能カーソルの場合は SCRSRyyyy。yyyy は xxxx と同じです。	ステートメント名は、Sxxxx の形式でなければなりません。 カーソル名の形式は、スクロール不能カーソルの場合は Cyy でなければなりません。yy は xxxx と同じで、1 から 15 の範囲内の値です。

注:

- ローカル・システムでは、ステートメント名に関する命名規則は適用されないため、クライアント・アプリケーションは、QSQRCEC システム API を使用して、作成されたステートメントを iSeries アプリケーションとの間で共用することができます。
- サーバーは、STMTxxxx の形式のステートメント名の先頭に空白を 1 つ付加します。したがって、ホスト・アプリケーションは、STMTxxxx 形式を使用するクライアント・アプリケーションとの間でステートメントを共用するためには、先行空白を付加する必要があります。ステートメント名が STMTxxxx の形式ではない場合は、サーバーは先行空白を付加しません。

DRDA を使用するときの規則と制限: 分散リレーショナル・データベース体系^(TM) (DRDA^(R)) は、DRDA をサポートする他のデータベースへのアクセスを許可するアーキテクチャーです。DRDA については、「分散データベース・プログラミング」を参照してください。

データベース・サーバーを、DRDA を使用する他の RDB へのゲートウェイとして使用する場合は、いくつかの機能上の制約条件に従う必要があります。

次の表は、データベース・サーバーからリモート・システムに接続しているときに制限を受ける機能を示しています。

DRDA 機能の制限

機能	制限
パッケージの作成	サポートされない機能
パッケージのクリア	
パッケージの削除	
準備	DRDA を使用しているときは拡張作成オプションは使用できません。

機能	制限
拡張動的パッケージ・サポート	<ul style="list-style-type: none"> DRDA を使用する場合、ステートメント名は 'STMTxxxx' の形式である必要があります。ここで、xxxx はセクション番号です。 DRDA を使用する場合、カーソル名は 'CURSORxxxx' または 'SCCURSORxxxx' の形式である必要があります。ここで、xxxx はセクション番号です。
パラメーター・マーカの記述	iSeries ^(TM) サーバーに接続しているときに限り使用できます。この機能は、DRDA を使用しているときはサポートされません。
コミット保留	iSeries サーバーに接続している場合に限り有効です。
コミット・レベル *NONE	サポートされません。
コミット・レベル *CHANGE	ターゲット RDB が iSeries である場合に限りサポートされます。その他の RDB の場合は *CS または *ALL コミット・レベルが必要です。

データ待ち行列サーバー

データ待ち行列は、iSeries^(TM) アプリケーション・プログラムが通信に使用するオブジェクトの 1 つです。アプリケーションは、データ待ち行列を使用して、ジョブ間でデータの受け渡しをすることができます。複数の iSeries ジョブで 1 つのデータ待ち行列を使用して、データの送信または受信を行うことができます。

PC アプリケーションでは、iSeries Access for Windows^(R) が提供する API を使用して、iSeries アプリケーションの場合と同様に簡単に、iSeries データ待ち行列を操作することができます。can. これにより、iSeries アプリケーションの通信機能の範囲が、リモート PC で実行されているプロセスにまで広がります。

このサーバーには、次の表に示すプログラムが含まれています。

ソケット・サポートと共に使用するために提供されているデータ待ち行列サーバー・プログラム

プログラム名	ライブラリー	説明
QZHQSSRV	QSYS	サーバー・プログラム
QZHQSRVD	QSYS	デーモン・プログラム

ネットワーク・プリント・サーバー

OS/400^(R) ネットワーク・プリント・サーバーにより、iSeries^(TM) サーバー上のプリント・リソースに対するクライアントの制御を強化することができます。このプリント・サーバーは、プリント・サービスの要求に基づき、次の機能を各クライアントに提供します。

スプール・ファイル

作成、シーク、オープン、読み取り、書き込み、クローズ、保留、リリース、削除、移動、送信、出口プログラム呼び出し、属性変更、メッセージ検索、メッセージ応答、属性検索、およびリスト

書き込み機能ジョブ

開始、終了、およびリスト

プリンター

属性検索およびリスト

出力待ち行列

保留、リリース、ページ、リスト、および属性検索

ライブラリー

リスト

プリンター・ファイル

属性検索、属性変更、およびリスト

ネットワーク・プリント・サーバー

属性変更および属性検索

このサーバーには、次の表に示すプログラムが含まれています。

ネットワーク・プリント・サーバー

プログラム名	ライブラリー	説明
QNPSEVRV	QSYS	サーバー・プログラム
QNPSEVRD	QSYS	デーモン・プログラム

セントラル・サーバー

セントラル・サーバーは、クライアントに次のサービスを提供します。

• ライセンス管理

データ転送 または PC5250 から最初に要求が出された時点で、該当の iSeriesTM Access for Windows[®] ユーザーのライセンスが予約されます。サーバーは、解放遅延タイムアウト時間が満了するまで活動状態を維持します。ライセンスは、解放されるかまたはサーバー・ジョブが終了するまで保持されます。どのようなライセンスが予約されているかを確認するには、iSeries ナビゲーターを使用して、iSeries システムのプロパティを表示します。

• 変換マップの検索

セントラル・サーバーは、変換マップを必要とするクライアントのために、変換マップを検索します。変換マップは、通常、ASCII から EBCDIC へ、また EBCDIC から ASCII への変換に使用されます。コード化文字セット ID (CCSID) を指定する必要があります。マップを要求するには、クライアントは、正しいソース CCSID、ターゲット CCSID、および変換するコード・ポイントのテーブルを指定します。これに対して、サーバーは、クライアントが使用するための正しいマッピングを戻します。

このサーバーには、次の表に示すプログラムが含まれています。

セントラル・サーバーのプログラム

プログラム名	ライブラリー	説明
QZSCSRV	QSYS	サーバー・プログラム
QZSCSRVD	QSYS	デーモン・プログラム

リモート・コマンドおよび分散プログラム呼び出しサーバー

ユーザーおよびアプリケーションは、リモート・コマンドおよび分散プログラム呼び出しサーバーのサポートを使用して、iSeriesTM CL コマンドを発行し、プログラムを呼び出すことができます。このサポート

を使用すると、同じジョブの中で複数のコマンドを実行することができます。また、この機能は、機能を制限されている (ユーザー・プロファイルの中で、LMTCPB =*YES が指定されている)、iSeries ユーザーに対して厳密なセキュリティー検査も行います。

アプリケーションは、分散プログラム呼び出しサポートによって、iSeries プログラムを呼び出し、パラメーター (入力および出力) を渡すことができます。プログラムが iSeries サーバー上で実行されてしまうと、出力パラメーター値がクライアント・アプリケーションに戻されます。このプロセスにより、アプリケーションは、必要な通信および変換について配慮せずに、簡単に iSeries のリソースにアクセスできます。

このサーバーには、次の表に示すプログラムが含まれています。

リモート・コマンドおよび分散プログラム呼び出しサーバーのプログラム

プログラム名	ライブラリー	説明
QZRCRVS	QSYS	サーバー・プログラム
QZRCRVS	QSYS	デーモン・プログラム

サインオン・サーバー

サインオン・サーバーは、クライアント向けのセキュリティーを提供します。このセキュリティー機能は、パスワードの有効期限が切れているユーザーがシステムにアクセスするのを防ぎ、ユーザー・プロファイルのパスワードを確認し、さらに、パスワード・キャッシング機能および iSeries^(TM) ナビゲーターのアプリケーション管理機能で使用するユーザー・プロファイル・セキュリティー情報を戻します。

このサーバーには、次の表に示すプログラムが含まれています。

サインオン・サーバーのプログラム

プログラム名	ライブラリー	説明
QZSOSIGN	QSYS	サーバー・プログラム
QZSOSIGN	QSYS	デーモン・プログラム

サーバー・ポートマッパー

ポートマッパーは、クライアントが、特定サービス (サーバー) 用のポートを見つけるための手段を提供します。ポートマッパーは、TCP/IP サービス・テーブル内でポートを見つけます。

このサーバーには、次の表に示すプログラムが含まれています。

サーバー・ポートマッパー

プログラム名	ライブラリー	説明
QZSOSMAPD	QSYS	サーバー・ポートマッパー・プログラム

拡張動的リモート SQL サーバー (QXDAEDRSQL)



クライアントは、QXDAEDRSQL サーバーを利用して、DB2^(R) UDB for iSeries^(TM) に搭載されている機能にアクセスすることができます。このサーバーは次の機能を提供します。

- リモート SQL アクセスに関するサポート
- XDA インターフェースを介したデータへのアクセス

- データベース機能 (ファイルの作成と削除およびファイル・メンバーの追加と削除など)

このサーバーには、次の表に示すプログラムが含まれています。

QXDAEDRSQL サーバー・プログラム

プログラム名	ライブラリー	説明
QXDARECVR	QSYS	サーバー・プログラム
QXDALISTEN	QSYS	デーモン・プログラム

注: QXDAEVT および QXDIAISP *SRVPGM オブジェクトは、QXDAEDRSQL サーバーにより使用されます。



DRDA/DDM サーバー



クライアントは、DRDA/DDM サーバーを利用して、DB2^(R) UDB for iSeries^(TM) に搭載されている機能にアクセスすることができます。このサーバーは次の機能を提供します。

- リモート SQL アクセスに関するサポート
- レコード・レベルのアクセスに関するサポート
- 遠隔ジャーナルに関するサポート

DRDA^(R) について詳しくは、「分散データベース・プログラミング」を参照してください。

DDM について詳しくは、「分散データ管理」を参照してください。

このサーバーには、次の表に示すプログラムが含まれています。

DRDA/DDM サーバー・プログラム

プログラム名	ライブラリー	説明
QRWTSRVR	QSYS	サーバー・プログラム
QRWTLSTN	QSYS	Listener プログラム



OS/400 ホスト・サーバーの使用

このトピックでは、OS/400^(R) サーバー・ジョブを管理する方法について説明します。説明の内容は、サーバーを実行するサブシステム、サーバーに影響を与えるオブジェクト、およびこれらのリソースの管理方法です。

OS/400 プログラムに付随して出荷される各種サーバーは、通常は、既存のシステム構成をまったく変更しなくても、正しく機能します。これらのサーバーは、OS/400 をインストールするときにセットアップされ構成されます。ユーザーとしては、要件を満たすため、問題を解決するため、システムのパフォーマンスを向上させるため、または単にシステム上のジョブを見るために、システムでサーバー・ジョブを管理する方法を変更したい場合があります。このような変更を行い、処理要件を満たすためには、どのオブジェクトがシステムのどの部分に影響を与えるか、そしてそれらのオブジェクトをどのように変更するかが分かっているなければなりません。システムの管理方法を完全に理解するには、このトピックの先に進む前に、「実行管理機能」を参照してください。

『クライアント/サーバー通信の確立』

このトピックでは、クライアント・サーバーとホスト・サーバー間の通信の開始と終了のためのプロセスについて説明します。さらに、個々の 41 ページの『ホスト・サーバーのポート番号』、およびサーバー・デーモンと通信における各デーモンの役割についても説明します。

45 ページの『iSeries サーバー上のサブシステム』

OS/400 サブシステムおよびジョブの自動開始および事前開始の方法について説明します。

57 ページの『iSeries サーバー上のシステム値』

クライアント/サーバー環境において重要なシステム値について説明します。

60 ページの『iSeries サーバー上のサーバー・ジョブの識別』

iSeries ナビゲーターまたは文字ベース・インターフェースを使用してサーバー・ジョブを表示する方法を示します。

63 ページの『ホスト・サーバーでの EZ セットアップおよび iSeries ナビゲーターの使用』

必要な通信パスが活動状態かどうかを判断する方法、および必要な場合にその通信パスを開始する方法について説明します。

クライアント/サーバー通信の確立

クライアント/サーバー通信を確立する手順は次のとおりです。

1. ソケット通信サポートを使用するサーバー・ジョブを開始するには、クライアント・システムは、特定のサーバーのポート番号に接続します。
2. クライアントの接続要求を listen し、受け入れるためのサーバー・デーモンを開始する必要があります (STRHOSTSVR コマンドを使用)。このサーバー・デーモンは、接続要求を受け入れると、クライアントの接続をサーバー・ジョブに結合するための内部要求を出します。
3. このサーバー・ジョブは、事前開始ジョブの場合もあり、また、事前開始ジョブが使用されていないときは、クライアント接続要求が処理されるときに実行依頼されるバッチ・ジョブの場合もあります。それ以後のクライアントとの通信はこのサーバー・ジョブが取り扱います。初期データ交換には、クライアント・ユーザーに関連したユーザー・プロファイルとパスワードを識別する要求が含まれています。
4. ユーザー・プロファイルとパスワードの妥当性検査が済むと、サーバー・ジョブはこのユーザー・プロファイルに切り替わり、このユーザー・プロファイル用に定義されている多くの属性 (会計コードや出力待ち行列など) を使用して、ジョブを変更します。

詳しくは、次のトピックを参照してください。

- 41 ページの『ホスト・サーバーのポート番号』
- 41 ページの『ホスト・サーバーの始動』
- 43 ページの『ホスト・サーバーの終了』

サーバーからホストへの通信

iSeries^(TM) Access for Windows^(R) は、TCP/IP を使用して iSeries システム・サーバーと通信します。最適化されたサーバーは、OS/400^(R) ソケット・サポートを使用してクライアントと通信します。OS/400 ソケットは、TCP/IP を介した Berkeley Software Distributions 4.3 ソケットとの互換性を備えています。ソケット・サポートは、iSeries サーバーにインストールされている 5722-TC1 製品から提供されます。

通信について詳しくは、「TCP/IP 構成および解説書」マニュアルを参照してください。

ホスト・サーバーのポート番号: サーバーの各タイプごとにそれぞれ専用のサーバー・デーモンがあり、このサーバーが、特定のポート上で着信クライアント接続要求を `listen` します。ただし、これにはいくつか例外があります。たとえば、ソケットを介した転送機能はデータベース・サーバー・デーモンを使用し、ネットワーク・ドライブ・サーバーはファイル・サーバー・デーモンを使用し、仮想プリント・サーバーはネットワーク・プリント・サーバー・デーモンを使用します。さらに、サーバー・マッパー・デーモンも指定されたポート上で `listen` し、指定されたサーバー用の現行のポート番号をクライアントが取得できるようにします。

各サーバー・デーモンは、指定されたサービス名用のサービス・テーブル内に提供されているポート番号で `listen` します。たとえば、提供された初期構成を持つネットワーク・プリント・サーバー・デーモンは、サービス名「`as-netprt`」に対応するポート番号 8474 で `listen` します。サーバー・マッパー・デーモンは、ウエルノウン・ポート上で `listen` します。ウエルノウン・サーバー・マッパー・ポート番号は、449 です。ウエルノウン・ポート番号は、OS/400^(R) ホスト・サーバー専用として予約されています。したがって、サービス名「`as-svrmap`」に該当する項目をサービス・テーブルから削除しないようにしてください。

各サーバー・デーモンのポート番号は固定的なものではありません。インストール先で違うポート番号を使用する必要がある場合は、それに合わせてサービス・テーブルを変更できます。iSeries^(TM) ナビゲーターの「システムのプロパティ、接続」タブで、ポート番号をどこから検索するかを変更することができます。ただし、サービス名は、変更されずに、下の表に示す名前のままになっていることが必要です。そうしないと、サーバー・デーモンは、クライアント接続を求める着信要求を受け入れるソケットを確立することができません。

あるサービスについて異なるポート番号を識別する新しいサービス・テーブルの項目を追加する場合は、そのサービス名に対応する既存のサービス・テーブル項目はすべて削除してください。これらの項目を削除することで、テーブル内でのサービス名の重複が排除され、サーバー・デーモンが開始されたときに予測外の結果が生じるおそれなくなります。

ホスト・サーバーおよびサーバー・マッパー用のポート番号

TCP 通信サポートを介してソケットを使用し、Secure Sockets Layer を使用する、最適化されたサーバーおよびサーバー・マッパーそれぞれのポート番号を表示します。

ホスト・サーバーの始動: OS/400^(R) ホスト・サーバーを始動するには、`STRHOSTSVR CL` コマンドを使用します。このコマンドは、ホスト・サーバー・デーモンおよびサーバー・マッパー・デーモンを開始します。また、そのサーバーに関連した事前開始ジョブを開始します。

注: iSeries^(TM) ナビゲーターを使用して、`STRTCP` コマンドで伝送制御プロトコル (TCP) を開始した時点で、サーバーが自動的に始動するようにシステムを構成できます。最近出荷されたシステムの場合、これはデフォルトとして設定されています。

各ホスト・サーバー・タイプには、1 つのサーバー・デーモンがあります。また、1 つのシステムごとに 1 つのサーバー・マッパー・デーモンがあります。クライアント PC アプリケーションは、ポート番号を使用してホスト・サーバー・デーモンに接続します。このサーバー・デーモンは、着信接続要求を受け入れ、それを処理のためにサーバー・ジョブに経路指定します。

STRHOSTSVR コマンド値 :



サーバー・タイプ

***ALL** すべてのホスト・サーバー・デーモンとサーバー・マッパー・デーモンを開始します。

*CENTRAL

QSYSWRK サブシステムでセントラル・サーバー・デーモンを開始します。デーモン・ジョブは QZSCSRVSD で、関連したサーバー事前開始ジョブは QZSCSRVS です。

*DATABASE

QSERVER サブシステムでデータベース・サーバー・デーモンを開始します。デーモン・ジョブは QZDASRVSD で、関連しているサーバー事前開始ジョブは QZDASOINIT、QZDASSINIT、および QTFPJTCP です。QTFPJTCP は QSERVER サブシステムで実行されます。

*DTAQ

QSYSWRK サブシステムでデータ待ち行列サーバー・デーモンを開始します。デーモン・ジョブは QZHQSRVD で、関連したサーバー事前開始ジョブは QZHQSSRV です。

*FILE QSERVER サブシステムでファイル・サーバー・デーモンを開始します。デーモン・ジョブは QPWFSERVSD で、関連したサーバー事前開始ジョブは QPWFSERVSO、QPWFSESVSS、および QPWFSERVS2 です。

*NETPRT

QSYSWRK サブシステムでネットワーク・プリント・サーバー・デーモンを開始します。デーモン・ジョブは QNPSEVRD で、関連したサーバー事前開始ジョブは QNPSEVRVS と QIWWPPJT です。QIWWPPJT は QSYSWRK サブシステムで実行されます。

*RMTCMD

QSYSWRK サブシステムでリモート・コマンド/分散プログラム呼び出しサーバー・デーモンを開始します。デーモン・ジョブは QZRCSRVD で、関連したサーバー事前開始ジョブは QZRCSRVS です。

*SIGNON

QSYSWRK サブシステムでサインオン・サーバー・デーモンを開始します。デーモン・ジョブは QZSOSGND で、関連したサーバー事前開始ジョブは QZSOSIGN です。

*SVRMAP

QSYSWRK サブシステムでサーバー・マッパー・デーモンを開始します。デーモン・ジョブは QZSOSMAPD です。

注: デーモン・ジョブが QSYSWRK ディレクトリー内で実行される場合、関連したサーバー事前開始ジョブは、デフォルトによって QUSRWRK ディレクトリーで実行されます。また、データベース・サーバー事前開始ジョブは、デフォルトでは QUSRWRK サブシステムで実行されます。

必要プロトコル

(このオプション・パラメーターは、ホスト・サーバー・デーモンを開始するために活動化する必要のある通信プロトコルを指定します。)

*ANY STRHOSTSVR コマンドを実行するときに、TCP/IP 通信プロトコルが活動状態になっていなければなりません。TCP/IP が活動状態になっていない場合は、診断メッセージ PWS3008 およびエスケープ・メッセージ PWS300D が発行され、ホスト・サーバー・デーモンは開始されません。

*NONE

ホスト・サーバー・デーモンを開始するために STRHOSTSVR コマンドを実行するときは、通信プロトコルを活動状態にしておく必要はありません。活動状態にないプロトコルに対してメッセージが出されることはありません。

***TCP** STRHOSTSVR コマンドを実行するときに、TCP/IP 通信プロトコルが活動状態になっていないければなりません。TCP/IP が活動状態になっていない場合は、診断メッセージ PWS3008 およびエスケープ・メッセージ PWS300D が発行され、ホスト・サーバー・デーモンは開始されません。



次に、STRHOSTSVR の『例: STRHOSTSVR』を示します。

例: STRHOSTSVR: 例 1: すべてのホスト・サーバー・デーモンを開始する

```
STRHOSTSVR(*ALL)
```

このコマンドは、少なくとも 1 つの通信プロトコルが活動状態であれば、すべてのサーバー・デーモンとサーバー・マップ・デーモンを開始します。

例 2: 特定のサーバー・デーモンを開始する

```
STRHOSTSVR SERVER(*CENTRAL *SVRMAP) RQDPCL(*NONE)
```

このコマンドは、活動状態の通信プロトコルが全くない場合でも、セントラル・サーバー・デーモンとサーバー・マップ・デーモンを開始します。

例 3: 必須プロトコルを 1 つ指定する

```
STRHOSTSVR SERVER(*ALL) RQDPCL(*TCP)
```

このコマンドは、TCP/IP が活動状態にある場合のみ、すべてのホスト・サーバー・デーモンおよびサーバー・マップ・デーモンを開始します。

ホスト・サーバーの終了: OS/400^(R) ホスト・サーバーを終了するには、ENDHOSTSVR CL コマンドを使用します。このコマンドは、ホスト・サーバー・デーモンとサーバー・マップ・デーモンを終了します。サーバー・デーモンが、そのタイプのサーバーがクライアント・アプリケーションに接続されている間に終了した場合、そのサーバー・ジョブは、オプションの ENDACTCNN パラメーターが指定されていない限り、クライアント・アプリケーションとの通信が終了するまで活動状態のままです。これ以降にクライアント・アプリケーションからそのサーバーへの接続要求が出されても、サーバー・デーモンが再度開始するまでは失敗します。

サーバー・マップ・デーモンが終了すると、サーバー・ジョブへの既存のクライアント接続はすべて無効になります。それ以降、クライアント・アプリケーションから、そのサーバー・マップとの接続要求が出されても、サーバー・マップが再度開始するまでは失敗します。

DATABASE** サーバーと ***FILE** サーバーとの活動状態の接続を終了させるために ENDACTCNN パラメーターを指定することができます。この指定によって、これらの接続をサービスしているサーバー・ジョブが終了することになります。活動状態の接続を終了できるのは、該当するデーモン・ジョブも終了する場合のみです。DATABASE** キーワードを指定すると、接続が活動状態である QZDASOINIT および QZDASSINIT ジョブが終了します。***FILE** キーワードを指定すると、接続が活動状態である QPWFSEVRSO および QPWFSEVRSVSS ジョブが終了します。

注: 活動状態でない特定のデーモンを ENDHOSTSVR コマンドを使用して終了すると、診断メッセージが表示されます。活動状態のデーモンを終了するには、ENDHOSTSVR SERVER(*ALL) を使用します。***ALL** 値を使用すると、診断メッセージは表示されません。

ENDHOSTSVR コマンド値:

サーバー・タイプ

***ALL** サーバー・デーモンとサーバー・マップパー・デーモンが活動状態であれば終了させます。これを使用した場合、システムでは、他の特殊値を使用できなくなります。

***CENTRAL**

QSYSWRK サブシステムでセントラル・サーバー・デーモンを終了させます。

***DATABASE**

QSERVER サブシステムでデータベース・サーバー・デーモンを終了させます。

***DTAQ**

QSYSWRK サブシステムでデータ待ち行列サーバー・デーモンを終了させます。

***FILE** QSERVER サブシステムでファイル・サーバー・デーモンを終了させます。

***NETPRT**

QSYSWRK サブシステムでネットワーク・プリント・サーバー・デーモンを終了させます。

***RMTCMD**

QSYSWRK サブシステムでリモート・コマンド/分散プログラム呼び出しサーバー・デーモンを終了させます。

***SIGNON**

QSYSWRK サブシステムでサインオン・サーバー・デーモンを終了させます。

***SVRMAP**

QSYSWRK サブシステムでサーバー・マップパー・デーモンを終了させます。

活動状態の接続を終了する

(このオプション・パラメーターは、特定のサーバーで活動状態の接続を終了させるかどうかを指定します。)

単一値:

***NONE**

活動状態の接続は終了しません。

その他の値:

***DATABASE**

QZDASOINIT および QZDASSINIT のサーバー・ジョブがサービスしている活動状態の接続が終了します。これらの接続をサービスしているサーバー・ジョブも終了することになります。

***FILE** QPWFSERVO および QPWFSERVSS のサーバー・ジョブがサービスしている活動状態の接続が終了します。これらの接続をサービスしているサーバー・ジョブも終了することになります。



次に、ENDHOSTSVR の『例: ENDMETHODSVR』をいくつか示します。

例: ENDMETHODSVR: 例 1: すべてのホスト・サーバー・デーモンを終了する

ENDHOSTSVR SERVER(*ALL)

このコマンドを使用すると、すべてのサーバー・デーモンとサーバー・マップパー・デーモンが終了します。

例 2: 特定のサーバー・デーモンを終了する

```
ENDHOSTSVR SERVER(*CENTRAL *SVRMAP)
```

セントラル・サーバー・デーモンとサーバー・マッパー・デーモンが終了します。

例 3: 特定のサーバー・デーモンと活動状態の接続を終了する

```
ENDHOSTSVR SERVER(*CENTRAL *DATABASE) ENDACTCNN(*DATABASE)
```

このコマンドは、QSYSWRK サブシステム内のセントラル・サーバー・デーモンと QSERVER サブシステム内のデータベース・サーバー・デーモンを終了させます。また、*DATABASE サーバーとの活動状態の接続、およびそれらの接続へのサービスを提供している QZDASOINIT と QZDASSINIT のサーバー・ジョブも終了します。

iSeries サーバー上のサブシステム

以下のトピックでは、各サーバー機能にどのシステム提供サブシステムを使用するかについて説明します。また、サブシステム記述がサーバー・ジョブにどのように関連するかについても詳しく説明します。

サブシステム記述は、どれだけの量の作業がどこでどのようにしてサブシステムに入り、サブシステムがその作業を実施するためにどのようなリソースを使用するかを定義します。

自動開始ジョブは、一回限りの初期化、または、特定のサブシステムに関連した反復作業を行います。特定のサブシステムに関連した自動開始ジョブは、そのサブシステムが始動されるたびに自動的に開始されます。

- サーバー・ジョブに使用されるサブシステム
- 自動開始ジョブの使用
- 事前開始ジョブの使用

サーバー・ジョブに使用されるサブシステム: サーバー・ジョブは、それぞれの機能に応じて異なるサブシステムで実行されるように構成されています。サーバー・ジョブには以下に示すサブシステムが使用されます。

QSYSWRK

このサブシステムでは、すべてのデーモン・ジョブ (ファイル・サーバー・デーモン・ジョブとデータベース・サーバー・デーモン・ジョブを除く) が実行されます。ファイル・サーバー・デーモン・ジョブとデータベース・サーバー・デーモン・ジョブは、QSERVER サブシステムで実行されます。

QUSRWRK

このサブシステムでは、次に示すサーバー用のサーバー・ジョブが実行されます。

- ネットワーク・プリント
- リモート・コマンドおよびプログラム呼び出し
- セントラル
- データ待ち行列
- サインオン
- データベース

QSERVER

このサブシステムでは、ファイル・サーバー・デーモン・ジョブ、それに関連した事前開始サーバー・ジョブ、およびデータベース・サーバー・デーモン・ジョブが実行されます。

このサブシステムが活動状態になっていない場合は、ファイル・サーバーまたはデータベース・サーバーとの接続を確立しようとする要求は失敗します。

サブシステムの自動開始

QSYSWRK サブシステムは、制御サブシステムに指定されている値に関係なく、IPL 時に自動的に開始されます。

システムに組み込まれているデフォルトの始動プログラムを使用した場合は、QSERVER および QUSRWRK サブシステムは IPL 時に自動的に開始されます。システム始動プログラムは QSTRUPPGM システム値で定義され、そのデフォルト値は QSTRUP QSYS です。

システムの始動を変更したいときは、QSTRUPPGM システム値を変更して、独自のプログラムが呼び出されるようにすることができます。ユーザーは、QSYS に含まれている QSTRUP プログラムをベースとして使用して、独自の始動プログラムを作成することができます。

注：データベース・サーバーまたはファイル・サーバーを使用しているときに、システム始動に変更を加えた場合は、始動プログラムが QSERVER サブシステムを始動することを確認する必要があります。

V5R1 では、システム始動プログラムに変更を加えなくても、TCP/IP は自動的に開始されます。ホスト・サーバーは、TCP/IP の開始時点で自動的に始動されます。TCP/IP は、開始後に、QUSRWRK および QSERVER が始動されていることを確認してから、ホスト・サーバーを始動します。

以前のリリースのシステムへの V5R1 (またはそれ以降) のスリップ・インストールの場合に、そのシステムが使用する始動プログラムが TCP/IP を開始するように変更されていた場合は、システムは自動的に TCP/IP を開始し、始動プログラムによる開始は失敗します。

IPL 属性 STRTCP を使用すると、システムが IPL 時に TCP/IP を自動的に開始できないようにすることができます。この値は出荷時の設定である *YES (TCP/IP の開始) のままにしておくことをお勧めしますが、必要であればこのオプションも使用できます。

自動開始ジョブの使用： QSERVER サブシステムでは、ファイル・サーバー・ジョブおよびデータベース・サーバー・ジョブ用の自動開始ジョブが定義されています。このジョブが実行されていない場合は、これらのサーバーは始動しません。ジョブがなくなってもサブシステムが終了することはありません。このジョブに問題が発生したときは、QSERVER サブシステムを終了し、再始動することができます。

QSYSWRK サブシステムでは、最適化されたすべてのサーバーに使用される、自動開始ジョブが定義されています。このジョブは、STRTCP コマンドが実行されたときに送られるイベントをモニターします。この方法により、サーバー・デーモン・ジョブは、いつ TCP/IP が活動状態になったかを動的に判断することができます。そして、デーモン・ジョブは、該当のポートでの listen を開始します。自動開始ジョブが活動状態でないときに、ホスト・サーバーが活動状態であり、TCP/IP が開始された場合は、TCP/IP の使用を開始するには、次の順序でコマンドを発行して、のプロトコルの使用を開始するには、次のコマンドを次の順序で実行する必要があります。

1. ENHOSTSVR *ALL
2. STRHOSTSVR *ALL

この自動開始ジョブの名前は QZBSEVTM です。このジョブが活動状態でない場合は、次のコマンドを発行してこのジョブを開始することができます。

```
QSYS/SBMJOB CMD(QSYS/CALL PGM(QSYS/QZBSEVTM)) JOB(QZBSEVTM) JOBID(QSYS/QZBSEJBD)
PRTDEV(*USRPRF) OUTQ(*USRPRF) USER(QUSER) PRTTXT(*SYSVAL) SYSLIBL(*SYSVAL)
CURLIB(*CRTDFT) INLLIBL(*JOBID) SRTSEQ (*SYSVAL) LANGID(*SYSVAL) CNTRYID(*SYSVAL)
CCSID(*SYSVAL)
```

注： QZBSEVTM プログラムのインスタンスは、一度に 1 つだけ実行することができます。

事前開始ジョブの使用： 事前開始ジョブは、リモート・システム上のプログラムがサーバーとの通信を開始する前に実行が開始されるバッチ・ジョブです。これは、サブシステム記述の事前開始ジョブ項目を使用して、ジョブの開始時にどのプログラム、クラス、および記憶域プールを使用するかを判断します。事前開始ジョブ項目には、事前開始ジョブのプールを作成および管理するのに使用するサブシステムの属性を指定しなければなりません。

事前開始ジョブを使用すると、サーバーとの接続を開始するときのパフォーマンスが向上します。事前開始ジョブ項目はサブシステム内で定義されます。事前開始ジョブは、サブシステムが始動されるときに活動状態になります。あるいは、事前開始ジョブの開始 (STRPJ) および事前開始ジョブの終了 (ENDPJ) コマンドを使用して、事前開始ジョブを制御することもできます。

事前開始ジョブに関連したシステム情報 (DSPACTPJ など) では、「プログラム始動要求」という語は、事前開始ジョブを始動するための要求を表す限定的な意味で使用されます。これは、そのシステム情報がソケット接続要求の結果として開始された事前開始ジョブに関連したものであっても同じです。

注：

- 事前開始ジョブは再使用できますが、いったん事前開始ジョブが使用され、その後プールに戻された後で、そのジョブの自動クリーンアップを行う機能はありません。事前開始ジョブが再使用される回数は、ADDPJE または CHGPJE CL コマンドの最大使用回数値 (MAXUSE) に指定されている値によって決まります。つまり、事前開始ジョブのユーザーの 1 人が使用しているリソースを、その事前開始ジョブの使用を終了する前にクリーンアップする必要があるということです。そうしないと、リソースは、次にこの事前開始ジョブを使用するユーザーに対しても同じ状況を維持することになります。たとえば、事前開始ジョブのユーザーの 1 人が開いたままで、閉じなかったファイルは、同じ事前開始ジョブの次のユーザーに対しても開いたままになり、そのユーザーが使用できる状態になっています。
- デフォルトでは、一部のサーバー・ジョブは QUSRWRK または QSERVER の中で実行されます。iSeriesTM ナビゲーターを使用すると、これらのサーバーの一部またはすべてを、選択したサブシステムで実行されるように構成することができます。
 - 「iSeries ナビゲーター → ネットワーク → サーバー → iSeries Access」の順にダブルクリックする。
 - サブシステム構成を行うサーバーを右マウス・ボタンでクリックし、「プロパティ」を選択する。
 - 「サブシステム」ページを使用して、サーバーを構成する。

ジョブをデフォルト・サブシステムから移動するには、次の操作が必要です。

- 専用のサブシステム記述を作成する。
- ADDPJE コマンドを使用して、独自の事前開始ジョブ項目を追加する。STRJOBS パラメーターを *YES に設定する。

この操作をしておかないと、ジョブは デフォルト・サブシステム の中で実行されることになります。

ソケット通信インターフェースによりサポートされる OS/400^(R) サーバーはすべて、事前開始ジョブをサポートします。

該当するサーバーには次のものがあります。

- ネットワーク・プリント・サーバー
- リモート・コマンドおよび分散プログラム呼び出しサーバー
- セントラル・サーバー
- データベース・サーバー
- セキュア・データベース・サーバー
- ファイル・サーバー
- セキュア・ファイル・サーバー
- データ待ち行列サーバー
- サインオン・サーバー (ソケット通信サポートを使用するサーバーに固有のもの)

次の表は、事前開始ジョブ項目属性のリストと、ソケット通信サポートを使用するホスト・サーバー用に構成される各属性の初期値を示しています。

サブシステム記述

事前開始ジョブ項目を含むサブシステム。

OS/400 サーバー	値
ネットワーク・プリント	QUSRWRK
リモート・コマンドおよびプログラム呼び出し	QUSRWRK
セントラル	QUSRWRK
データベース	QUSRWRK
セキュア・データベース	QUSRWRK
ファイル	QSERVER
セキュア・ファイル	QSERVER
データ待ち行列	QUSRWRK
サインオン	QUSRWRK

プログラム・ライブラリー/プログラム名

事前開始ジョブが開始されるときに呼び出されるプログラム。

OS/400 サーバー	値
ネットワーク・プリント	QSYS/QNPSESRVS
リモート・コマンドおよびプログラム呼び出し	QSYS/QZRCSRVS
セントラル	QSYS/QZSCSRVS
データベース	QSYS/QZDASOINIT
セキュア・データベース	QSYS/QZDASSINIT
ファイル	QSYS/QPWFSESRVSO
セキュア・ファイル	QSYS/QPWFSESRVSS
データ待ち行列	QSYS/QZHQSSRV
サインオン	QSYS/QZSOSIGN

ユーザー・プロファイル

ジョブを実行するユーザー・プロファイル。これは、ジョブでユーザー・プロファイルとして示されます。クライアントからサーバーの開始要求が受信されると、事前開始ジョブ機能は、その要求の中で指定されているユーザー・プロファイルに切り替えます。

OS/400 サーバー	値
ネットワーク・プリント	QUSER
リモート・コマンドおよびプログラム呼び出し	QUSER
セントラル	QUSER
データベース	QUSER
セキュア・データベース	QUSER
ファイル	QUSER
セキュア・ファイル	QUSER
データ待ち行列	QUSER
サインオン	QUSER

ジョブ名

ジョブが開始される時のそのジョブの名前。

OS/400 サーバー	値
ネットワーク・プリント	*PGM
リモート・コマンドおよびプログラム呼び出し	*PGM
セントラル	*PGM
データベース	*PGM
セキュア・データベース	*PGM
ファイル	*PGM
セキュア・ファイル	*PGM
データ待ち行列	*PGM
サインオン	*PGM

ジョブ記述

事前開始ジョブに使用するジョブ記述。*USRPRF が指定されている場合は、このジョブを実行するプロファイルのジョブ記述が使用されるという点に注意してください。これは、QUSER のジョブ記述が使用されるということです。要求元のジョブ記述からの属性もいくつか使用されます。印刷装置および出力待ち行列は、要求元ユーザーのジョブ記述からスワップされます。

OS/400 サーバー	値
ネットワーク・プリント	QSYS/QZBSJOB
リモート・コマンドおよびプログラム呼び出し	QSYS/QZBSJOB
セントラル	QSYS/QZBSJOB
データベース	QGPL/QDFTSVR
セキュア・データベース	QGPL/QDFTSVR
ファイル	QGPL/QDFTSVR
セキュア・ファイル	QGPL/QDFTSVR

OS/400 サーバー	値
データ待ち行列	QSYS/QZBSJOB
サインオン	QSYS/QZBSJOB

ジョブ開始

事前開始ジョブを、サブシステムの始動時に自動的に開始するかどうかを示します。この事前開始ジョブ項目は、出荷時には、サーバー・ジョブが必ず使用可能になるように *YES に設定されています。

STRHOSTSVR コマンドを使用すると、その処理の一環としてそれぞれの事前開始ジョブが開始されます。

OS/400 サーバー	値
ネットワーク・プリント	*YES
リモート・コマンドおよびプログラム呼び出し	*YES
セントラル	*YES
データベース	*YES
セキュア・データベース	*YES
ファイル	*YES
セキュア・ファイル	*YES
データ待ち行列	*YES
サインオン	*YES

初期ジョブ数

サブシステムの始動時に開始されるジョブの数。この値は、インストール先特有の環境と要件に合わせて調整できます。

OS/400 サーバー	値
ネットワーク・プリント	1
リモート・コマンドおよびプログラム呼び出し	1
セントラル	1
データベース	1
セキュア・データベース	1
ファイル	1
セキュア・ファイル	1
データ待ち行列	1
サインオン	1

しきい値

1 つの事前開始ジョブ項目について使用できる事前開始ジョブの最小数。このしきい値に達すると、追加の事前開始ジョブが自動的に開始されます。このしきい値により、プール内に一定数のジョブが保持されます。

OS/400 サーバー	値
ネットワーク・プリント	1
リモート・コマンドおよびプログラム呼び出し	1
セントラル	1

OS/400 サーバー	値
データベース	1
セキュア・データベース	1
ファイル	1
セキュア・ファイル	1
データ待ち行列	1
サインオン	1

追加ジョブ数

しきい値に達したときに開始される追加の事前開始ジョブの数。

OS/400 サーバー	値
ネットワーク・プリント	2
リモート・コマンドおよびプログラム呼び出し	2
セントラル	2
データベース	2
セキュア・データベース	2
ファイル	2
セキュア・ファイル	2
データ待ち行列	2
サインオン	2

最大ジョブ数

この項目について活動状態にすることのできる事前開始ジョブの最大数。

OS/400 サーバー	値
ネットワーク・プリント	*NOMAX
リモート・コマンドおよびプログラム呼び出し	*NOMAX
セントラル	*NOMAX
データベース	*NOMAX
セキュア・データベース	*NOMAX
ファイル	*NOMAX
セキュア・ファイル	*NOMAX
データ待ち行列	*NOMAX
サインオン	*NOMAX

最大使用数

ジョブの最大使用回数。この値が 200 であれば、事前開始ジョブは、サーバー始動要求が 200 回処理された時点で終了します。

OS/400 サーバー	値
ネットワーク・プリント	200
リモート・コマンドおよびプログラム呼び出し	1

OS/400 サーバー	値
セントラル	200
データベース	200
セキュア・データベース	200
ファイル	*NOMAX
セキュア・ファイル	*NOMAX
データ待ち行列	200
サインオン	200

ジョブ待機

最大ジョブ数に達したときに、サーバー・ジョブが使用可能になるまで、クライアント接続要求を待機させるかどうかを示します。

OS/400 サーバー	値
ネットワーク・プリント	*YES
リモート・コマンドおよびプログラム呼び出し	*YES
セントラル	*YES
データベース	*YES
セキュア・データベース	*YES
ファイル	*YES
セキュア・ファイル	*YES
データ待ち行列	*YES
サインオン	*YES

プール ID

この事前開始ジョブが実行されるサブシステム・プール。

OS/400 サーバー	値
ネットワーク・プリント	1
リモート・コマンドおよびプログラム呼び出し	1
セントラル	1
データベース	1
セキュア・データベース	1
ファイル	1
セキュア・ファイル	1
データ待ち行列	1
サインオン	1

クラス

事前開始ジョブを実行するクラスの名前とライブラリー。

OS/400 サーバー	値
ネットワーク・プリント	QGPL/QCASERVR

OS/400 サーバー	値
リモート・コマンドおよびプログラム呼び出し	QGPL/QCASERVR
セントラル	QGPL/QCASERVR
データベース	QSYS/QPWFSEVER
セキュア・データベース	QSYS/QPWFSEVER
ファイル	QSYS/QPWFSEVER
セキュア・ファイル	QSYS/QPWFSEVER
データ待ち行列	QGPL/QCASERVR
サインオン	QGPL/QCASERVR

事前開始ジョブ項目についてジョブ開始の値が *YES に設定されていて、その他の値がすべて初期設定のままになっているときは、各事前開始ジョブ項目について次のアクションがとられます。

- サブシステムの始動時に、各サーバーについて 1 つずつ事前開始ジョブが開始されます。
- 最初のクライアント接続が特定のサーバーに対して複数のプロセスを要求すると、初期ジョブが使用され、しきい値を超えます。
- そのサーバーについて、事前開始ジョブ項目内で定義されている数に従って追加のジョブが開始されます。
- 使用可能なジョブ数は、常に少なくとも 1 です。
- サブシステムは、要求を処理できる状態になっている事前開始ジョブの数を定期的に検査し、余分なジョブを終了します。サブシステムは、少なくとも初期ジョブ数パラメーターに指定されている数の事前開始ジョブだけは残しておきます。

事前開始ジョブのモニター

事前開始ジョブをモニターするには、活動事前開始ジョブの表示 (DSPACTPJ) コマンドを使用します。たとえば、サインオン・サーバー用の事前開始ジョブをモニターするには、事前開始ジョブが入っているサブシステム (QUSRWRK またはユーザー定義のサブシステム) と、プログラム (たとえば QZSOSIGN) が分かっている必要があります。

DSPACTPJ コマンドは次のような情報を提供します。

```

+-----+
                活動事前開始ジョブの表示
                AS400597
                01/03/07 04:10:02
サブシステム . . . . . : QUSRWRK   リセット日付 . . . . . : 01/11/95
プログラム   . . . . . : QZSOSIGN リセット時刻 . . . . . : 16:54:50
ライブラリー . . . . . :   QSYS     経過時間 . . . . . : 0023:12:21

事前開始ジョブ :
現在数 . . . . . : 10
平均数 . . . . . : 8.5
ピーク時の数 . . . . . : 25

使用中の事前開始ジョブ数 :
現在数 . . . . . : 5
平均数 . . . . . : 4.3
ピーク時の数 . . . . . : 25

                続く ...
  
```

		01/03/07 04:10:02
サブシステム :	QUSRWRK	リセット日付 : 01/11/95
プログラム :	QZSOSIGN	リセット時刻 : 16:54:50
ライブラリー :	QSYS	経過時間 : 0023:12:21
プログラム開始要求数 :		
待機中の現在数 :		0
待機中の平均数 :		.2
待機中のピーク時の数 :		0
平均待機時間 :		00:00:20.0
受け入れられた数 :		0
拒否された数 :		0
終わり		
続行するためには、実行キーを押してください。		
F3= 終了 F5= 最新表示 F12= 取り消し F13= 統計のリセット		

事前開始ジョブの管理

「活動事前開始ジョブの表示」画面が表示されているときに **F5** キーを押すと、特定の活動事前開始ジョブについて表示されている情報を最新表示することができます。プログラム開始要求に関する情報には、使用可能な事前開始ジョブ数を変更する必要があるかどうかを示される場合があります。事前開始ジョブが使用可能になるのを待っているプログラム開始要求があることが示されている場合は、事前開始ジョブ項目の変更 (CHGPJE) コマンドにより事前開始ジョブを変更することができます。

プログラム開始要求に対するアクションが速やかに行われない場合は、次の処置を任意に組み合わせて使用できます。

- しきい値を大きくする。
- 初期ジョブ数 (INLJOBS) のパラメーター値を大きくする。
- 追加のジョブ数 (ADLJOBS) のパラメーター値を大きくする。

重要なのは、すべての要求について必ず使用可能な事前開始ジョブがあるようにすることです。

事前開始ジョブの除去

サーバーが事前開始ジョブ機能を使用しないようにしたい場合は、次のことを行う必要があります。

1. 事前開始ジョブの終了 (ENDPJ) コマンドにより事前開始ジョブを終了する。

ENDPJ コマンドにより終了した事前開始ジョブは、事前開始ジョブ項目でジョブ開始に *YES が指定されている場合は次回のサブシステム始動時に開始され、また、指定されているサーバー・タイプを対象として STRHOSTSVR コマンドが実行された場合にも開始されます。事前開始ジョブを終了しただけで、次のステップを実行しなかった場合は、特定のサーバーを始動しようとする要求はすべて失敗します。

2. 事前開始ジョブ項目の除去 (RMVPJE) コマンドを使用して、サブシステム記述の中の事前開始ジョブ項目を除去する。

RMVPJE コマンドを使用して除去した事前開始ジョブ項目は、サブシステム記述から永久に除去されません。いったん、項目が除去されると、該当サーバーに対する新たな要求ができます。

経路指定項目の使用

デーモン・ジョブがサブシステムのいずれかに経路指定されるときに、ジョブはサブシステム記述の中の経路指定項目を使用します。ホスト・サーバー・デーモン・ジョブ用の経路指定項目は、STRHOSTSVR コマンドが実行されたときにサブシステム記述に追加されます。これらのジョブは QUSER ユーザー・プロファイルの下で実行されます。QSYSWRK サブシステムに投入されたデーモン・ジョブの場合は、QSYSNOMAX ジョブ待ち行列が使用されます。QSERVER サブシステムに投入されたサブシステムの場合は、QPWFSESERVER ジョブ待ち行列が使用されます。

サーバー・ジョブの特性は、それぞれの事前開始ジョブ項目からとられます。サーバーに事前開始ジョブを使用しない場合は、サーバー・ジョブは、それぞれに対応するデーモン・ジョブの特性を使用して開始されます。

次に示すのは、IBM^(R) 提供のサブシステムの中での、個々のサーバー・デーモン・ジョブの初期構成です。

ネットワーク・プリント・サーバー・デーモン

サブシステム	QSYS/QSYSWRK
ジョブ待ち行列	QSYSNOMAX
ユーザー	QUSER
データの経路指定	QNPSERVD
ジョブ名	QNPSERVD
クラス	QGPL/QCASERVER
シーケンス番号	2538

リモート・コマンドおよびプログラム呼び出しサーバー・デーモン

サブシステム	QSYS/QSYSWRK
ジョブ待ち行列	QSYSNOMAX
ユーザー	QUSER
データの経路指定	QZRCRVSD
ジョブ名	QZRCRVSD
クラス	QGPL/QCASERVER
シーケンス番号	2539

セントラル・サーバー・デーモン

サブシステム	QSYS/QSYSWRK
ジョブ待ち行列	QSYSNOMAX
ユーザー	QUSER
データの経路指定	QZSCRVSD
ジョブ名	QZSCRVSD
クラス	QGPL/QCASERVER

シーケンス番号	2536
---------	------

データベース・サーバー・デーモン

サブシステム	QSYS/QSERVER
ジョブ待ち行列	QPWFSEVER
ユーザー	QUSER
データの経路指定	QZDASRVSD
ジョブ名	QZDASRVSD
クラス	QSYS/QPWFSEVER
シーケンス番号	600

ファイル・サーバー・デーモン

サブシステム	QSYS/QSERVER
ジョブ待ち行列	QPWFSEVER
ユーザー	QUSER
データの経路指定	QPWFSEVSD
ジョブ名	QPWFSEVSD
クラス	QSYS/QPWFSEVER
シーケンス番号	200

データ待ち行列サーバー・デーモン

サブシステム	QSYS/QSYSWRK
ジョブ待ち行列	QSYSNOMAX
ユーザー	QUSER
データの経路指定	QZHQSRVD
ジョブ名	QZHQSRVD
クラス	QGPL/QCASERVER
シーケンス番号	2537

サインオン・サーバー・デーモン

サブシステム	QSYS/QSYSWRK
ジョブ待ち行列	QSYSNOMAX
ユーザー	QUSER
データの経路指定	QZSOSGND
ジョブ名	QZSOSGND
クラス	QGPL/QCASERVER
シーケンス番号	2540

サーバー・マップパー・デーモン

サブシステム	QSYS/QSYSWRK
ジョブ待ち行列	QSYSNOMAX
ユーザー	QUSER
データの経路指定	QZSOSMAPD
ジョブ名	QZSOSMAPD
クラス	QGPL/QCASERVR
シーケンス番号	2541

iSeries サーバー上のシステム値

システム値には、システムの特典部分を操作する制御情報が入っています。ユーザーは、作業環境を定義するために、システム値を変更することができます。システム値には、たとえばシステム日付やライブラリー・リストなどがあります。

iSeries^(TM) サーバーには多数のシステム値があります。以下に示す値は、クライアント/サーバー環境で特に重要な意味を持つものです。

QAUDCTL

監査制御。このシステム値には、オブジェクト・レベルおよびユーザー・レベルの監査をオンまたはオフにするスイッチが入っています。このシステム値に対する変更は即時に有効になります。

QAUDENDACN

監査ジャーナル・エラー・アクション。このシステム値は、オペレーティング・システムのセキュリティ監査ジャーナルが監査ジャーナル項目を送信しているときにエラーが発生した場合に、システムがとるアクションを指定します。このシステム値に対する変更は即時に有効になります。

QAUDFRCLVL

監査ジャーナル強制書き込み。このシステム値は、監査ジャーナル項目データが補助記憶域に強制書き込みされる前に、セキュリティ監査ジャーナルに書き込むことができる監査ジャーナル項目の数を指定します。このシステム値に対する変更は即時に有効になります。

QAUDLVL

セキュリティ監査レベル。このシステム値に対する変更は、システム上で実行されているすべてのジョブについて即時に有効になります。

QAUTOVRT

システムが自動的に仮想デバイスを作成するかどうかを決定します。これは、表示装置 (ディスプレイ) パススルー・セッションおよび Telnet セッションで使用されます。

QCCSID

次のものを識別するコード化文字セット ID

- 特定の一組のエンコード・スキーム ID
- 文字セット ID
- コード・ページ ID
- システムが必要とするコード化図形文字表現を一意的に識別する、その他のコーディング関連情報

この値は、システムにインストールされている言語によって異なります。この値によって、ユーザーに提示するデータを、事前に別の形式に変換する必要があるかどうかが決まります。デフォルト値は 65535 で、これはデータを変換しないことを意味します。

QCTLSBSD

制御サブシステム記述

QDSPSGNINF

5250 エミュレーション機能 (ワークステーション機能 PC5250) を使用したサインオンの後で、サインオン情報画面を表示するかどうかを決定します。

QLANGID

システムのデフォルトの言語 ID。これは、ジョブ CCSID が 65535 である場合に、ユーザーのジョブ用のデフォルトの CCSID を決定します。クライアントおよびサーバーは、このデフォルト・ジョブ CCSID 値を使用して、クライアントとサーバーの間で交換されるデータの正しい変換を判別します。

QLMTSECOFR

全オブジェクト (*ALLOBJ) またはサービス (*SERVICE) 特殊権限を持つユーザーが、どのデバイスも使用できるかどうかを制御します。この値が 1 に設定されている場合は、*ALLOBJ または *SERVICE 特殊権限を持つすべてのユーザーは、デバイスを使用するための特定の *CHANGE 権限を持っている必要があります。

これは、5250 エミュレーションの場合の仮想デバイスに影響を与えます。出荷時の値は 1 です。許可ユーザーが PC にサインオンできるようにしたいときは、その PC で使用するデバイスおよびコントローラーに対する特定権限をユーザーに付与するか、またはこの値を 0 に変更してください。

QMAXSIGN

ローカル・ユーザーおよびリモート・ユーザーが、サインオンの誤りを連続して何回繰り返すことができるかを制御します。QMAXSIGN の値に達した場合は、システムは QMAXSGNACN システム値に従ってアクションを決めます。

QMAXSGNACN の値が 1 (デバイスをオフに変更) である場合は、ユーザーが、接続を開始しようとして誤ったパスワードを PC に入力しても、QMAXSIGN の値に影響されません。

これは、PC ユーザーにとって機密漏れを起こすおそれがあります。したがって、QMAXSGNACN は 2 または 3 のいずれかに設定するようにしてください。

QMAXSGNACN

いずれかのデバイスでサインオン試行の最大回数に達したときに、システムがとる処置を決定します。1 (デバイスをオフに変更)、2 (ユーザー・プロファイルを使用不可にする)、または 3 (デバイスをオフに変更し、ユーザー・プロファイルを使用不可にする) のいずれかを指定できます。出荷時の値は 3 です。

QPWDEXPITV

パスワードの有効日数。このシステム値に対する変更は即時に有効になります。

QPWDLMTAJC

パスワード内で隣接する数字を使用することを制限します。このシステム値に対する変更は、次回にパスワードを変更する時点で有効になります。

QPWDLMTCHR

パスワード内での特定文字の使用を制限します。このシステム値に対する変更は、次回にパスワードを変更する時点で有効になります。

QPWDLMTREP

パスワード内での反復文字の使用を制限します。このシステム値に対する変更は、次回にパスワードを変更する時点で有効になります。

QPWDLVL

システムのパスワード・サポートのレベルを決定します。これには、iSeries サーバーがサポートするパスワードの長さ、パスワードに使用する暗号化のタイプ、および、Windows^(R) クライアント用の iSeries ネットサーバー・パスワードをシステムから削除するかどうか、などが含まれます。このシステム値に対する変更は次回の IPL 時に有効になります。

重要: この値を、長いパスワードをサポートするように設定する場合は、その前に、すべてのクライアント PC を、長いパスワードをサポートするもの (Express V5R1) にアップグレードする必要があります。こうしておかないと、V5R1 より前のバージョンのクライアントはすべて、iSeries サーバーにログオンできなくなります。

QPWDMAXLEN

パスワードに使用する文字の最大数。このシステム値に対する変更は、次回にパスワードを変更する時点で有効になります。

QPWDMINLEN

パスワードに使用する文字の最小数。このシステム値に対する変更は、次回にパスワードを変更する時点で有効になります。

QPWDPOSDIF

新しいパスワード内での文字の位置を制御します。このシステム値に対する変更は、次回にパスワードを変更する時点で有効になります。

QPWDRQDDGT

新しいパスワード内に数字が必要なことを指定します。このシステム値に対する変更は、次回にパスワードを変更する時点で有効になります。

QPWDRQDDIF

パスワードを前のパスワードと変えるべきかどうかを制御します。

QPWDLDPGM

コンピューター・システムが提供するパスワード検証プログラムの名前とライブラリー。オブジェクト名とライブラリー名の両方を指定できます。このシステム値に対する変更は、次回にパスワードを変更する時点で有効になります。

QRMTSIGN

システムがリモート・サインオン要求をどのように処理するかを指定します。TELNET セッションは、実際にはリモート・サインオン要求です。この値により、次に示すようないくつかのアクションを指定できます。

- ***FRCSIGNON**: すべてのリモート・サインオン・セッションは、通常のサインオン処理の過程を経る必要がある。
- ***SAMEPRF**: 5250 ディスプレイ装置のパススルーまたはワークステーション機能において、ソースとターゲットのユーザー・プロファイル名が同じである場合は、リモート・サインオンの試行に対してサインオンをバイパスできる。TELNET を使用している場合に、サインオンをバイパスすることができます。
- ***VERIFY**: システムは、ユーザーがシステムに対するアクセス権を持っていることを確認した上で、そのユーザーによるサインオンのバイパスを容認する。
- ***REJECT**: 5250 ディスプレイ装置のパススルーまたはワークステーション機能の場合のリモート・サインオンをすべて拒否する。QRMTSIGN が ***REJECT** に設定されていても、ユーザーは TELNET を使用すればシステムにサインオンすることができます。TELNET セッションは、通常の処理の過程を経ることになります。システムへの TELNET 要求もすべて拒否されるようにするには、TELNET サーバーを終了してください。

- 'program library' : ユーザーは、特定のプログラムおよびライブラリー (または *LIBL) を指定することにより、どのリモート・セッションが許容されるか、および、どのユーザー・プロファイルがどの場所から自動的にサインオンできるかを決定することができます。このオプションはパススルーの場合のみ有効です。

この値は、どのリモート・セッションが許容されるかを判別するために実行するプログラム名も指定します。

出荷時の値は *FRCSIGNON です。5250 エミュレーターのバイパス・サインオン機能をユーザーが使用できるようにするには、この値を *VERIFY に変更してください。

QSECURITY

システム・セキュリティー・レベル。このシステム値に対する変更は次回の IPL 時に有効になります。

- 20 は、サインオンのためにパスワードが必要なことを意味します。
 - 30 は、サインオン時にはパスワード・セキュリティーが必要であり、個々のアクセス時にはオブジェクト・セキュリティーが必要であることを意味します。ユーザーは、すべてのシステム・リソースへのアクセス権限を持っている必要があります。
 - 40 は、サインオン時にはパスワード・セキュリティーが必要であり、個々のアクセス時にはオブジェクト・セキュリティーが必要であることを意味します。サポートされないインターフェースを使用してオブジェクトにアクセスしようとするプログラムは、失敗します。
 - 50 は、サインオン時にパスワード・セキュリティーが必要であり、ユーザーがオブジェクトおよびシステム・リソースにアクセスする権限を持っていないことを意味します。
- QTEMP ライブラリーおよびユーザー・ドメイン・オブジェクトのセキュリティーと保水性は、強制適用されます。サポートされないインターフェースを使用してオブジェクトにアクセスしようとするプログラム、またはサポートされるインターフェースにサポートされないパラメーター値を渡そうとするプログラムは、失敗します。

QSTRUPPGM

制御サブシステムが始動される時、またはシステムが始動する時に実行されるプログラム。このプログラムは、サブシステムの始動といったセットアップ機能を実行します。

QSYSLIBL

ライブラリー・リストのシステム部分。ライブラリー・リストのこの部分は、他の部分より前に検索されます。一部のクライアント機能は、このリストを検索してオブジェクトを見つけます。

iSeries サーバー上のサーバー・ジョブの識別

エミュレーターまたは文字ベース・インターフェースを使用すると、ジョブを特定のパーソナル・コンピューターまたは個々のクライアント機能に関連付けすることが困難な場合があります。特定のジョブを識別できるようにすることは、問題を究明し、パフォーマンスへの影響を判断するための前提条件です。iSeries ナビゲーター・インターフェースを使用して、サーバー・ジョブを識別できるようになりました。

1. 「iSeriesTM ナビゲーター」アイコンをダブルクリックする。
2. 正符号 (+) をクリックして、「ネットワーク」をオープンする。
3. 正符号 (+) をクリックして、「サーバー」をオープンする。
4. どのタイプのサーバーについてジョブを表示するかを選択する (たとえば、TCP/IP または iSeriesTM Access for Windows^R など)。

5. 右側のペインにサーバーが表示されたら、ジョブを表示したいサーバーを右マウス・ボタンでクリックし、「サーバー・ジョブ」をクリックする。新しいウィンドウが開き、該当のサーバーのサーバー・ジョブ、ジョブ・タイプ、ジョブ状況、サーバーのシステムに入力された時刻、およびシステムに入力された日付が表示されます。

以下の各トピックでは、従来の文字ベース・インターフェースを使用してサーバー・ジョブを識別する方法を示します。

- 45 ページの『iSeries サーバー上のサブシステム』
- 『iSeries のジョブ名』
- 『サーバー・ジョブの表示』
- 62 ページの『ヒストリー・ログの表示』
- 63 ページの『特定ユーザーのサーバー・ジョブの表示』

iSeries のジョブ名: iSeries[™] で使用されるジョブ名は、次の 3 つの部分から成っています。

- 単純ジョブ名
- ユーザー ID
- ジョブ番号 (昇順)

サーバー・ジョブについては次のような規則があります。

- ジョブ名
 - 事前開始ジョブ以外の場合は、サーバー・ジョブ名はサーバー・プログラムの名前です。
 - 事前開始ジョブは、事前開始ジョブ項目内で定義されている名前を使用します。
 - サーバーが開始するジョブは、ジョブ記述名を使用するか、または、バッチ・ジョブの場合は指定された名前を使用します (ファイル・サーバーがこれを行います)。
- ユーザー ID
 - これは、事前開始ジョブが使用されるかどうかに関係なく、常に QUSER です。
 - ジョブ・ログには、どのユーザーがジョブを使用したかが示されます。
- ジョブ番号は実行管理機能により作成されます。

サーバー・ジョブの表示: サーバー・ジョブを識別するには、2 つの方法があります。第 1 の方法は、WRKACTJOB コマンドを使用する方法です。第 2 の方法は、ヒストリー・ログを表示して、どのジョブをどのクライアントが使用しているかを判別する方法です。

WRKACTJOB を使用して活動ジョブを表示する。 WRKACTJOB コマンドは、すべての活動ジョブ、および、サーバー・デーモンとサーバー・マッパー・デーモンを表示します。

次の図は、WRKACTJOB コマンドにより表示される状況の例です。図には、サーバーに関連したジョブだけが示されています。使用可能な事前開始ジョブを表示するには、**(F14)** を押す必要があります。

図に示されているのは次のタイプのジョブです。

- **(1)** - サーバー・マッパー・デーモン
- **(2)** - サーバー・デーモン
- **(3)** - 事前開始サーバー・ジョブ

活動ジョブの処理				AS400597
CPU %:	3.1	経過時間 :	21:38:40	01/02/26 18:36:52
		活動ジョブ数 :	77	

オプションを入力して、実行キーを押してください。

2= 変更 3= 保留 4= 終了 5= 処理 6= 解放 7= メッセージの表示
8= スプール・ファイルの処理 13= 切断 ...

OPT	サブシステム/ジョブ	ユーザー	タイプ	CPU %	機能	状況
___	QSYSWRK	QSYS	SBS	.0		DEQW
___	(1) QZSOSMAPD	QUSER	BCH	.0		SELW
___	(2) QZSOSGND	QUSER	BCH	.0		SELW
___	QZSCSRVSD	QUSER	BCH	.0		SELW
___	QZRCSRVD	QUSER	BCH	.0		SELW
___	QZHQSRVD	QUSER	BCH	.0		SELW
___	QNPSEVRD	QUSER	BCH	.0		SELW
___	QUSRWRK	QSYS	SBS	.0		DEQW
___	(3) QZSOSIGN	QUSER	PJ	.0		PSRW
___	QZSCSRVS	QUSER	PJ	.0		PSRW
___	QZRCSRVS	QUSER	PJ	.0		PSRW
___	QZHQSSRV	QUSER	PJ	.0		PSRW
___	QNPSEVS	QUSER	PJ	.0		PSRW
___	QZDASOINIT	QUSER	PJ	.0		PSRW

続く ...

活動ジョブの処理

AS400597

01/02/26 18:36:52

CPU %: 3.1 経過時間 : 21:38:40 活動ジョブ数 : 77

オプションを入力して、実行キーを押してください。

2= 変更 3= 保留 4= 終了 5= 処理 6= 解放 7= メッセージの表示
8= スプール・ファイルの処理 13= 切断 ...

OPT	サブシステム/ジョブ	ユーザー	タイプ	CPU %	機能	状況
___	QSERVER	QSYS	SBS	.0		DEQW
___	QSERVER	QPGMR	ASJ	.1		EVTW
___	(2) QPWFSERVSD	QUSER	BCH	.0		SELW
___	QZDASRVSD	QUSER	BCH	.0		SELW
___	(3) QPWFSERVSO	QUSER	PJ	.0		PSRW
___	QPWFSEVS	QUSER	PJ	.0		PSRW

続く ...

次のタイプのジョブが表示されています。

ASJ サブシステム用の自動開始ジョブ

PJ 事前開始されたサーバー・ジョブ

SBS サブシステム・モニター・ジョブ

BCH サーバー・デーモンおよびサーバー・マッパー・デーモン・ジョブ

ヒストリー・ログの表示: 各クライアントがサーバー・ジョブのいずれかに正常に接続するたびに、そのジョブは、該当のクライアント・ユーザーのプロファイルの下で実行するようにスワップされます。特定のクライアント・ユーザーにどのジョブが関連しているかを判別するには、**DSPLOG** コマンドを使用してヒストリー・ログを表示します。そして、次の文字で始まるメッセージを見つけてください。

- CPIAD0B (サインオン・サーバー・メッセージ)
- CPIAD09 (その他のすべてのサーバーに関するメッセージ)

特定ユーザーのサーバー・ジョブの表示: 特定のユーザーについてサーバー・ジョブを表示するには、次のようにします。

1. **iSeries^(TM) ナビゲーター** をオープンする (該当のアイコンをダブルクリック)。
2. 「**ユーザーおよびグループ**」をクリックし、次に「**すべてのユーザー**」をクリックする。
3. サーバー・ジョブを表示したいユーザーを右マウス・ボタンでクリックする。
4. 「**ユーザー・オブジェクト**」を選択し、「**ジョブ**」をクリックする。このユーザーのすべてのサーバー・ジョブを示すウィンドウが表示されます。

WRKOBJLCK コマンドを使用することもできます。その場合は、ユーザー・プロファイルおよび *USRPRF を指定します。

ホスト・サーバーでの EZ セットアップおよび iSeries ナビゲーターの使用

iSeries のサインオン・サーバー、セントラル・サーバー、およびリモート・コマンドおよび分散プログラム呼び出しサーバーで通信プロトコルが実行されていない場合でも、EZ セットアップおよび iSeries^(TM) ナビゲーターは、これらのサーバーに接続することができます。つまり、EZ セットアップでは、STRTCP が実行される前に接続することができます。EZ セットアップには、通信プロトコルを構成または開始する前に、ある程度の iSeries 初期セットアップができるパスが使用されています。このトピックでは、EZ セットアップおよびオペレーション・コンソールで使用される通信パスが活動状態にあるかどうかを判別する方法、および必要な場合にその通信パスを再始動する方法について説明します。

EZ セットアップにより使用される接続を構成する方法については、EZ セットアップのオンライン・ヘルプを参照してください。

EZ セットアップにより使用される通信パスが活動状態になるためには、QSYSWRK サブシステム内で 3 つのジョブ (QNEOSOEM) が実行されていることが必要です。QSYSWRK サブシステムには、この通信パス用の自動開始ジョブがあります。この自動開始ジョブ (QNEOSOEM) は、QSYSWRK サブシステム内にある QNEOSOEM という名前の他の 2 つのジョブを実行依頼します。これらのジョブのどちらかが活動状態にない場合は、次のコマンドを実行してそのジョブを開始してください。

```
QSYS/SBMJOB CMD(QSYS/CALL PGM(QSYS/QNEOSOEM)) JOB(QNEOSOEM)
JOBQ(QSYS/QNEOJOBQ) JOBQ(QSYS/QSYSNOMAX) PRTDEV(*JOBQ) OUTQ(*JOBQ)
USER(*JOBQ) PRTEXT(*JOBQ) SYSLIBL(*SYSVAL) INLLIBL(*JOBQ)
LOGCLPGM(*YES) MSGQ(*NONE) SRTSEQ(*SYSVAL) LANGID(*SYSVAL)
CNTRYID(*SYSVAL) CCSID(*SYSVAL)
```

このコマンドは、必要な場合に、これら 3 つの QNEOSOEM ジョブをすべて開始します。

サーバー出口プログラムの使用

システム管理者は、出口プログラムを使用して、個々の特定サーバーについてクライアント・ユーザーが行うことのできる活動を制御します。すべてのサーバーが、ユーザー作成の出口プログラムをサポートしています。このトピックでは、出口プログラムの使用方法と構成方法について説明します。また、サーバー機能へのアクセスを制御するために利用できるサンプル・プログラムも示します。

- 64 ページの『出口プログラムの登録』
- 67 ページの『出口プログラムの作成』
- 67 ページの『出口プログラムのパラメーター』
- 86 ページの『例: 出口プログラム』

注：法的に重要な情報については、「コードに関する特記事項」を参照してください。

出口プログラムの登録

どの出口プログラムを呼び出すかをサーバーが判断できるようにするには、出口プログラムを登録する必要があります。出口プログラムを登録するには、OS/400^(R) の登録機能を使用します。

登録機能の扱い

登録機能に出口プログラムを登録するには、登録情報の処理 (WRKREGINF) コマンドを使用します。

```
-----+-----
                          登録情報の処理 (WRKREGINF)
-----+-----

  選択項目を入力して、実行キーを押してください。

  出口点 . . . . . *REGISTERED
  出口点形式 . . . . . *ALL      名前, 総称* , *ALL
  出力 . . . . . *              *, *PRINT
-----+-----
```

登録された出口点を表示するには、実行キー (Enter) を押します。

```
-----+-----
                          登録情報の処理
-----+-----

  オプションを入力して、実行キーを押してください。
  5= 出口点の表示 8= 出口プログラムの処理

  Opt  出口点      出口点   登録   テキスト
        出口点      の形式   済み
  -   QIBM_QCA_CHG_COMMAND  CHGC0100 *YES  コマンド出口プログラムの変更
  -   QIBM_QCA_RTV_COMMAND  RTVC0100 *YES  コマンド出口プログラムの検索
  -   QIBM_QHQ_DTAQ          DTAQ0100 *YES  元のデータ待ち行列サーバー
  -   QIBM_QIMG_TRANSFORMS  XFRM0100 *YES
  -   QIBM_QJO_DLT_JRNRCV   DRCV0100 *YES  ジャーナル・レシーバーの削除
  -   QIBM_QLZP_LICENSE     LICM0100 *YES  元のライセンス管理サーバー
  -   QIBM_QMF_MESSAGE      MESS0100 *YES  元のメッセージ・サーバー
  -   QIBM_QMH_REPLY_INQ    RPYI0100 *YES  照会メッセージへの応答のハンドル
  8   QIBM_QNPS_ENTRY       ENTR0100 *YES  ネットワーク印刷サーバー - 入力
  -   QIBM_QNPS_SPLF        SPLF0100 *YES  ネットワーク印刷サーバー - スプール
  -   QIBM_QOE_OV_USR_ADM   UADM0100 *YES  OfficeVision/400 管理

  コマンド
  ===>
-----+-----
```

操作対象のサーバーについて定義されている出口点用の出口プログラムに対する作業を行うには、オプションを 8 を選択します。

```
-----+-----
                          出口プログラムの処理
-----+-----

  出口点:  QIBM_QNPS_ENTRY      形式:  ENTR0100

  オプションを入力して、実行キーを押してください。
  1= 追加  4= 除去  5= 表示  10= 置換え

  Opt      出口プログラム  出口   ライブラリー
        番号      プログラム
  1_      _____  _____  _____
-----+-----
```

(出口プログラムが見つからない。)

出口点に出口プログラムを追加するには、オプション 1 を使用します。

注:

- すでに定義されている出口プログラムの名前を変更するには、その前にそのプログラムを削除する必要があります。
- 登録機能は、1 つの出口点および様式名について複数のユーザー出口をサポートできますが、サーバーは常に出口プログラム 1 を検索します。
- 変更を有効にするには、事前開始ジョブをいったん終了してから、再開する必要があります。

-----+
出口プログラムの追加 (ADDEXITPGM)

選択項目を入力して、実行キーを押してください。

```
出口点 . . . . . > QIBM_QNPS_ENTRY
出口点形式 . . . . . > ENTR0100 名前
プログラム番号 . . . . . > 1      1-2147483647, *LOW, *HIGH
プログラム . . . . . MYPGM      名前
ライブラリー . . . . . MYLIB      Name, *CURLIB
スレッド・セーフ . . . . . *UNKNOWN *UNKNOWN, *NO, *YES
マルチスレッド・ジョブ・アクション . *SYSVAL *SYSVAL, *RUN, *MSG,
スレッド保護 . . . . . *UNKNOWN *UNKNOWN, *NO, *YES
マルチスレッド・ジョブの処置 . . . . *SYSVAL *SYSVAL, *RUN, *MSG, *NORUN
```

-----+
この出口点に該当するプログラムのプログラム名とライブラリーを入力します。

複数の出口点に同じプログラムを使用することもできます。プログラムは、入力として送られたデータを使用して、各タイプの要求をどのように取り扱うかを判断することができます。

次に、個々の OS/400 サーバー別に出口点と様式名を示します。

QIBM_QPWFS_FILE_SERV (ファイル・サーバー)

様式名	PWFS0100
アプリケーション名	*FILESRV

QIBM_QZDA_INIT (データベース・サーバー開始)

様式名	ZDAI0100
アプリケーション名	*SQL

QIBM_QZDA_NDB1 (データベース・サーバーのネイティブ・データベース要求)

様式名	ZDAQ0100 ZDAQ0200
アプリケーション名	*NDB

QIBM_QZDA_ROI1 (データベース・サーバーのオブジェクト情報検索要求)

様式名	ZDAR0100 ZDAR0200
アプリケーション名	*RTVOBJNF

QIBM_QZDA_SQL1 (データベース・サーバーの SQL 要求)

様式名	ZDAQ0100
アプリケーション名	*SQLSRV

QIBM_QZDA_SQL2 (データベース・サーバーの SQL 要求)

様式名	ZDAQ0200
アプリケーション名	*SQLSRV

QIBM_QZHQ_DATA_QUEUE (データ待ち行列サーバー)

様式名	ZHQ00100
アプリケーション名	*DATAQSRV

QIBM_QNPS_ENTRY (ネットワーク・プリント・サーバー)

様式名	ENTR0100
アプリケーション名	QNPSERVR

QIBM_QNPS_SPLF (ネットワーク・プリント・サーバー)

様式名	SPLF0100
アプリケーション名	QNPSERVR

QIBM_QZSC_LM (セントラル・サーバーのライセンス管理要求)

様式名	ZSCL0100
アプリケーション名	*CNTRLSRV

QIBM_QZSC_NLS (セントラル・サーバーの NLS 要求)

様式名	ZSCN0100
アプリケーション名	*CNTRLSRV

QIBM_QZSC_SM (ライセンス・サーバー)

様式名	ZSCS0100
アプリケーション名	*CNTRLSRV

QIBM_QZRC_RMT (リモート・コマンド/分散プログラム呼び出しサーバー)

様式名	CZRC0100
アプリケーション名	*RMTSRV

QIBM_QZSO_SIGNONSRV (サインオン・サーバー)

様式名	ZSOY0100
アプリケーション名	*SIGNON

出口プログラムの作成

ユーザーが出口プログラムを指定すると、サーバーは、ユーザーの要求を実行する前に、次の 2 つのパラメーターをその出口プログラムに渡します。

- 1 バイトの戻りコード値
- 要求に関する情報を含む構造 (この構造は出口点ごとに異なります)

出口プログラムは、この 2 つのパラメーターを使用して、要求の実行が可能かどうかを判断することができます。出口プログラムが戻りコードを X'F1' に設定している場合は、サーバーは要求を許可します。戻りコードが X'F0' に設定されている場合は、サーバーは要求を拒否します。X'F1' または X'F0' 以外の値が設定された場合の結果は、どのサーバーにアクセスしようとしているかによって異なります。

同じプログラムを、複数のサーバーおよび出口点に使用することができます。プログラムは、第 2 パラメーターとして渡された構造内のデータを調べて、どのサーバーが呼び出され、どの機能が使用されているかを判断することができます。

出口プログラムに送られる第 2 パラメーターの構造については、『出口プログラムのパラメーター』に説明があります。この情報を使用して、ユーザー独自の出口プログラムを作成することができます。

出口プログラムのパラメーター

以下のトピックでは、それぞれの OS/400^(R) サーバーについて、出口点形式の第 2 パラメーターのデータ構造を示します。

- 『ファイル・サーバー』
- 69 ページの『データベース・サーバー』
- 78 ページの『データ待ち行列サーバー』
- 79 ページの『ネットワーク・プリント・サーバー』
- 81 ページの『セントラル・サーバー』
- 83 ページの『リモート・コマンドおよび分散プログラム呼び出しサーバー』
- 85 ページの『サインオン・サーバー』

ファイル・サーバー: ファイル・サーバーには、出口点が 1 つ定義されています。

QIBM_QPWFS_FILE_SERV 様式 PWFS0100

QIBM_QPWFS_FILE_SERV 出口点は、次のタイプのファイル・サーバー要求に対して出口プログラムを実行するように定義されています。

- ファイル属性変更
- ストリーム・ファイル作成またはディレクトリー作成

- ファイル削除またはディレクトリー削除
- ファイル属性リスト
- 移動
- ストリーム・ファイルのオープン
- リネーム
- 会話割り振り

注 :

ファイル・サーバーの場合は、出口プログラム名は QSERVER サブシステムが活動化された時点で解決されず。プログラム名を変更した場合に、その変更を有効にするには、サブシステムを終了し、再始動する必要があります。

出口点 QIBM_QPWFS_FILE_SERV 様式 PWFS0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	ファイル・サーバーの場合の値は *FILESRV です。
20	14	BINARY(4)	要求機能	実行されている機能 : <ul style="list-style-type: none"> • X'0000' - ファイル属性変更要求 • X'0001' - ストリーム・ファイル作成またはディレクトリー作成要求 • X'0002' - ファイル削除またはディレクトリー削除要求 • X'0003' - ファイル属性リスト要求 • X'0004' - 移動要求 • X'0005' - ストリーム・ファイル・オープン要求 • X'0006' - リネーム要求 • X'0007' - 会話割り振り要求
24	18	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QPWFS_FILE_SERV の場合の様式名は PWFS0100 です。
32	20	CHAR(4)	ファイル・アクセス	要求された機能の値が '5' (オープン) である場合は、このフィールドには次の構造が入っています。 <ul style="list-style-type: none"> • 読み取りアクセス、CHAR(1) X'F1' - Yes X'F0' - No • 書き込みアクセス、CHAR(1) X'F1' - Yes X'F0' - No • 読み取り/書き込みアクセス、CHAR(1) X'F1' - Yes X'F0' - No • 削除可能、CHAR(1) X'F1' - Yes X'F0' - No
36	24	BINARY(4)	ファイル名の長さ	ファイル名 (次のフィールド) の長さ。この長さは最大 16MB です。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
40	28	CHAR(*)	ファイル名	ファイルの名前。このフィールドの長さは、「ファイル名の長さ (File Name Length) (前のフィールド) に指定されます。ファイル名は、ISO/IEC 10646 (UCS-2 レベル 1) 文字セット CCSID 61952 で戻されます。
<p>注：</p> <ul style="list-style-type: none"> この様式は、ライブラリー QSYSINC の中のファイル H、QRPGRSRC、QRPGLSRC、QLBLSRC、および QCBLLSRC 内のメンバー EPWFSEP により定義されます。 ISO/IEC 10646 (UCS-2 レベル 1) 文字セットについて詳しくは、「<i>Information Standard, ISO/IEC 10646-1: Information technology — Universal—Octet Character Set (UCS) — Part 1: Architecture and Basic Multilingual Plane</i>」(参照番号 ISO/IEC 10646-1: 1993(E)) を参照してください。 <p>UCS-2 レベル 1 との間の変換のために使用できる API は、iconv() および CDRCVRT です。</p>				

データベース・サーバー： データベース・サーバーには、5 つの異なる出口点が定義されています。

1. QIBM_QZDA_INIT
 - サーバーの開始時に呼び出されます。
2. QIBM_QZDA_NDB1
 - ネイティブ・データベース要求に対して呼び出されます。
3. QIBM_QZDA_SQL1
 - SQL 要求に対して呼び出されます。
4. QIBM_QZDA_SQL2
 - SQL 要求に対して呼び出されます。
5. QIBM_QZDA_ROI1
 - オブジェクト情報検索要求および SQL カタログ機能に対して呼び出されます。

ネイティブ・データベースおよびオブジェクト情報検索のための出口点には、要求される機能のタイプに応じた 2 つの様式が定義されています。

QIBM_QZDA_INIT 出口点は、サーバーの開始時に出口プログラムを実行するように定義されています。この出口点にプログラムが定義されている場合は、データベース・サーバーが開始されるたびにそのプログラムが呼び出されます。

出口点 QIBM_QZDA_INIT 様式 ZDAI0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	この出口点の場合の値は *SQL です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZDA_INIT の場合の様式名は ZDAI0100 です。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
28	1C	BINARY(4)	要求機能	実行されている機能。 この出口点の場合の有効な値は 0 だけです。
注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLLESRC 内のメンバー EZDAEP により定義されます。				

QIBM_QZDA_NDB1 出口点は、データベース・サーバーの場合のネイティブ・データベース要求に対して出口プログラムを実行するように定義されています。この出口点については、2 つの様式が定義されています。様式 ZDAD0100 は次の機能に使用されます。

- ソース物理ファイルの作成
- 既存のファイルを基にしたデータベース・ファイルの作成
- データベース・ファイル・メンバーの追加、クリア、削除
- データベース・ファイルのオーバーライド
- データベース・ファイル・オーバーライドの削除
- ファイルの削除

様式 ZDAD0200 は、ライブラリー・リストにライブラリーを追加する要求が受信されたときに使用されません。

出口点 QIBM_QZDA_NDB1 様式 ZDAD0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	この出口点の場合の値は *NDB です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 次の機能の場合は、様式名は ZDAD0100 です。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
28	1C	BINARY(4)	要求機能	実行されている機能。 このフィールドには次のいずれかが入ります。 <ul style="list-style-type: none"> • X'00001800' - ソース物理ファイルの作成 • X'00001801' - データベース・ファイルの作成 • X'00001802' - データベース・ファイル・メンバーの追加 • X'00001803' - データベース・ファイル・メンバーのクリア • X'00001804' - データベース・ファイル・メンバーの削除 • X'00001805' - データベース・ファイルのオーバーライド • X'00001806' - データベース・ファイル・オーバーライドの削除 • X'00001807' - 保管ファイルの作成 • X'00001808' - 保管ファイルのクリア • X'00001809' - ファイルの削除
32	20	CHAR(128)	ファイル名	要求された機能に使用されるファイルの名前。
160	A0	CHAR(10)	ライブラリー名	ファイルを含むライブラリーの名前。
170	AA	CHAR(10)	メンバー名	追加、クリア、または削除するメンバーの名前。
180	B4	CHAR(10)	権限	作成されるファイルに対する権限。
190	BE	CHAR(128)	基になるファイル名	既存のファイルをベースにしてファイルを作成する場合に使用するファイルの名前。
318	13E	CHAR(10)	基になるライブラリー名	ベースとなるファイルを含むライブラリーの名前。
328	148	CHAR(10)	オーバーライドされるファイル名	オーバーライドされるファイルの名前。
338	152	CHAR(10)	オーバーライドされるライブラリー名	オーバーライドされるファイルを含むライブラリーの名前。
348	15C	CHAR(10)	オーバーライドされるメンバー名	オーバーライドされるメンバーの名前。
注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLLESRC 内のメンバー EZDAEP により定義されます。				

出口点 QIBM_QZDA_NDB1 様式 ZDAD0200

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	この出口点の場合の値は *NDB です。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。ライブラリー・リストへの追加機能の場合の様式名は、ZDAD0200 です。
28	1C	BINARY(4)	要求機能	実行されている機能。 X'0000180C' - ライブラリー・リストの追加
32	20	BINARY(4)	ライブラリー数	ライブラリー (次のフィールド) の数
36	24	CHAR(10)	ライブラリー名	各ライブラリーのライブラリー名
注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLESRC 内のメンバー EZDAEP により定義されます。				

QIBM_QZDA_SQL1 出口点は、データベース・サーバーについて受信された特定の SQL 要求に対して出口プログラムを実行するように定義されています。この出口点用に定義されている様式は 1 つだけです。出口プログラムが呼び出されるのは、次の機能が要求された場合です。

- 準備
- オープン
- 実行
- 接続
- パッケージの作成
- パッケージのクリア
- パッケージの削除
- ストリーム・フェッチ
- 即時実行
- 作成と記述
- 準備と実行または準備とオープン
- オープンとフェッチ
- 実行またはオープン
- パッケージ情報の戻り

出口点 QIBM_QZDA_SQL1 様式 ZDAQ0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	この出口点の場合の値は *SQLSRV です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZDA_SQL1 の場合の様式名は ZDAQ0100 です。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
28	1C	BINARY(4)	要求機能	<p>実行されている機能。</p> <p>このフィールドには次のいずれかが入ります。</p> <ul style="list-style-type: none"> • X'00001800' - 準備 • X'00001803' - 準備と記述 • X'00001804' - オープン/記述 • X'00001805' - 実行 • X'00001806' - 即時実行 • X'00001809' - 接続 • X'0000180C' - ストリーム・フェッチ • X'0000180D' - 準備と実行 • X'0000180E' - オープンとフェッチ • X'0000180F' - パッケージの作成 • X'00001810' - パッケージのクリア • X'00001811' - パッケージの削除 • X'00001812' - 実行またはオープン • X'00001815' - パッケージ情報の戻り
32	20	CHAR(18)	ステートメント名	準備または実行機能に使用されるステートメントの名前。
50	32	CHAR(18)	カーソル名	オープン機能に使用されるカーソルの名前。
68	44	CHAR(2)	準備オプション	準備機能に使用されるオプション。
70	46	CHAR(2)	オープン属性	オープン機能に使用されるオプション。
72	48	CHAR(10)	拡張動的パッケージ名	拡張動的 SQL パッケージの名前。
82	52	CHAR(10)	パッケージ・ライブラリー名	拡張動的 SQL パッケージのライブラリーの名前。
92	5C	BINARY(2)	DRDA ^(R) インディケータ	<ul style="list-style-type: none"> • 0 - ローカル RDB に接続されている。 • 1 - リモート RDB に接続されている。
94	5E	CHAR(1)	コミットメント制御レベル	<ul style="list-style-type: none"> • 'A' - コミット *ALL • 'C' - コミット *CHANGE • 'N' - コミット *NONE • 'S' - コミット *CS (カーソル固定性) • 'L' - コミット *RR (反復可能読み取り)
95	5F	CHAR(512)	SQL ステートメント・テキストの最初の 512 バイト	SQL ステートメントの最初の 512 バイト。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLESRC 内のメンバー EZDAEP により定義されます。				

QIBM_QZDA_SQL2 出口点は、データベース・サーバーについて受信された特定の SQL 要求に対して出口プログラムを実行するように定義されています。QIBM_QZDA_SQL2 出口点は QIBM_QZDA_SQL1 出口点より優先されます。QIBM_QZDA_SQL2 出口点用に登録されているプログラムがある場合は、そのプログラムが呼び出され、QIBM_QZDA_SQL1 出口点用のプログラムは呼び出されません。出口プログラムが呼び出されるのは、次の機能が要求された場合です。

- 準備
- オープン
- 実行
- 接続
- パッケージの作成
- パッケージのクリア
- パッケージの削除
- ストリーム・フェッチ
- 即時実行
- 作成と記述
- 準備と実行または準備とオープン
- オープンとフェッチ
- 実行またはオープン
- パッケージ情報の戻り

表 A-6: 出口点 QIBM_QZDA_SQL2 様式 ZDAQ0200

0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	この出口点の場合の値は *SQLSRV です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZDA_SQL2 の場合の様式名は ZDAQ0200 です。

28	1C	BINARY(4)	要求機能	<p>実行されている機能。</p> <p>このフィールドには次のいずれかが入ります。</p> <ul style="list-style-type: none"> • X'00001800' - 準備 • X'00001803' - 準備と記述 • X'00001804' - オープン/記述 • X'00001805' - 実行 • X'00001806' - 即時実行 • X'00001809' - 接続 • X'0000180C' - ストリーム・フェッチ • X'0000180D' - 準備と実行 • X'0000180E' - オープンとフェッチ • X'0000180F' - パッケージの作成 • X'00001810' - パッケージのクリア • X'00001811' - パッケージの削除 • X'00001812' - 実行またはオープン • X'00001815' - パッケージ情報の戻り
32	20	CHAR(18)	ステートメント名	準備または実行機能に使用されるステートメントの名前。
50	32	CHAR(18)	カーソル名	オープン機能に使用されるカーソルの名前。
68	44	CHAR(2)	準備オプション	準備機能に使用されるオプション。
70	46	CHAR(2)	オープン属性	オープン機能に使用されるオプション。
72	48	CHAR(10)	拡張動的パッケージ名	拡張動的 SQL パッケージの名前。
82	52	CHAR(10)	パッケージ・ライブラリー名	拡張動的 SQL パッケージのライブラリーの名前。
92	5C	BINARY(2)	DRDA 標識	<ul style="list-style-type: none"> • 0 - ローカル RDB に接続されている。 • 1 - リモート RDB に接続されている。
94	5E	CHAR(1)	コミットメント制御レベル	<ul style="list-style-type: none"> • 'A' - コミット *ALL • 'C' - コミット *CHANGE • 'N' - コミット *NONE • 'S' - コミット *CS (カーソル固定性) • 'L' - コミット *RR (反復可能読み取り)
95	5F	CHAR(10)	デフォルトの SQL コレクション	iSeries TM データベース・サーバーが使用するデフォルトの SQL コレクションの名前。
105	69	CHAR(129)	予約済み	将来のパラメーター用に予約されています。
234	EA	BINARY(4)	SQL ステートメントのテキスト長さ	続くフィールドに入っている SQL ステートメント・テキストの長さ。この長さは最大 64K です。
238	EE	CHAR(*)	SQL ステートメントのテキスト	SQL ステートメント全体。

注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLLSRC 内のメンバー EZDAEP により定義されます。

QIBM_QZDA_ROI1 出口点は、データベース・サーバーに対して特定のオブジェクトに関する情報の検索が要求された場合に、出口プログラムを実行するように定義されています。また、この出口点は SQL カタログ機能にも使用されます。

この出口点には 2 つの様式が定義されています。次にこれらの様式について説明します。

様式 ZDAR0100 は、次のオブジェクトに関する情報の検索要求に対して使用されます。

- ライブラリー (またはコレクション)
- ファイル (またはテーブル)
- フィールド (または列)
- 索引
- リレーショナル・データベース (または RDB)
- SQL パッケージ
- SQL パッケージ・ステートメント
- ファイル・メンバー
- レコード様式
- 特殊列

様式 ZDAR0200 は、次のオブジェクトに関する情報の検索要求に対して使用されます。

- 外部キー
- 基本キー

出口点 QIBM_QZDA_ROI1 様式 ZDAR0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	データベース・サーバーの場合の値は *RTVOBJINF です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。次の機能の場合、様式名は ZDAR0100 です。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
28	1C	BINARY(4)	要求機能	<p>実行されている機能。</p> <p>このフィールドには次のいずれかが入ります。</p> <ul style="list-style-type: none"> • X'00001800' - ライブラリー情報の検索 • X'00001801' - リレーショナル・データベース情報の検索 • X'00001802' - SQL パッケージ情報の検索 • X'00001803' - SQL パッケージ・ステートメントの検索 • X'00001804' - ファイル情報の検索 • X'00001805' - ファイル・メンバー情報の検索 • X'00001806' - レコード様式情報の検索 • X'00001807' - フィールド情報の検索 • X'00001808' - 索引情報の検索 • X'0000180B' - 特殊列情報の検索
32	20	CHAR(20)	ライブラリー名	ライブラリー、パッケージ、パッケージ・ステートメント、ファイル、メンバー、レコード様式、フィールド、索引、および特殊列に関する情報を検索するときに使用するライブラリーまたは検索パターン。
52	34	CHAR(36)	リレーショナル・データベース名	RDB 情報を検索するために使用するリレーショナル・データベース名または検索パターン。
88	58	CHAR(20)	パッケージ名	パッケージまたはパッケージ・ステートメントに関する情報を検索するために使用するパッケージ名または検索パターン。
108	6C	CHAR(256)	ファイル名 (SQL 別名)	ファイル、メンバー、レコード様式、フィールド、索引、または特殊列に関する情報を検索するために使用するファイル名または検索パターン。
364	16C	CHAR(20)	メンバー名	ファイル・メンバーに関する情報を検索するために使用されるメンバー名または検索パターン。
384	180	CHAR(20)	様式名	レコード様式に関する情報を検索するために使用される様式名または検索パターン。
<p>注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLLESRC 内のメンバー EZDAEP により定義されます。</p>				

出口点 QIBM_QZDA_ROI1 様式 ZDAR0200

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロフィール名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	データベース・サーバーの場合の値は *RTVOBJINF です。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。次の機能の場合は、様式名は ZDAR0200 です。
28	1C	BINARY(4)	要求機能	実行されている機能。 このフィールドには次のいずれかが入ります。 <ul style="list-style-type: none"> • X'00001809' - 外部キー情報の検索 • X'0000180A' - 基本キー情報の検索
32	20	CHAR(10)	基本キー・テーブルのライブラリー名	基本キーおよび外部キーに関する情報を検索するときに使用する基本キー・テーブルが入っているライブラリーの名前。
42	2A	CHAR(128)	基本キー・テーブル名 (別名)	基本キーまたは外部キーに関する情報を検索するときに使用する基本キーが入っているテーブルの名前。
170	AA	CHAR(10)	外部キー・テーブルのライブラリー名	外部キー情報を検索するときに使用する外部キー・テーブルが入っているライブラリーの名前。
180	64	CHAR(128)	外部キー・テーブル名 (別名)	外部キー情報を検索するときに使用する外部キーが入っているテーブルの名前。
注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLLSRC 内のメンバー EZDAEP により定義されます。				

データ待ち行列サーバー: データ待ち行列サーバーには、出口点が 1 つ定義されています。

QIBM_QZHQ_DATA_QUEUE 様式 ZHQ00100

出口点 QIBM_QZHQ_DATA_QUEUE は、次のデータ待ち行列サーバー要求が受信されたときに、出口点プログラムを実行するように定義されています。

- 照会
- 受信
- 作成
- 削除
- 送信
- クリア
- 取り消し
- 検査

出口点 QIBM_QZHQ_DATA_QUEUE 様式 ZHQ00100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	データ待ち行列サーバーの場合の値は *DATAQSRV です。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZHQ_DATA_QUEUE の場合の様式名は ZHQ00100 です。
28	1C	BINARY(4)	要求機能	実行されている機能。 <ul style="list-style-type: none"> • X'0001' - データ待ち行列の属性の照会 • X'0002' - データ待ち行列からのメッセージの受信 • X'0003' - データ待ち行列の作成 • X'0004' - データ待ち行列の削除 • X'0005' - データ待ち行列へのメッセージの送信 • X'0006' - データ待ち行列からのメッセージのクリア • X'0007' - 保留中の受信要求の取り消し • X'0012' - データ待ち行列からメッセージを削除せずに受信
32	20	CHAR(10)	オブジェクト名	データ待ち行列名。
42	2A	CHAR(10)	ライブラリー名	データ待ち行列ライブラリー。
52	34	CHAR(2)	関係操作	要求に対するキー順操作の関係演算子。 X'0000' - 演算子なし 'EQ' - 等しい 'NE' - 等しくない 'GE' - より大か等しい 'GT' - より大 'LE' - より小か等しい 'LT' - より小
54	36	BINARY(4)	キー長	要求で指定されているキー長。
58	3A	CHAR(256)	キー値	要求で指定されているキー値。
注： この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLESRC 内のメンバー EZHQEP により定義されます。				

ネットワーク・プリント・サーバー: ネットワーク・プリント・サーバーには 2 つの出口点が定義されています。

1. QIBM_QNPS_ENTRY format ENTR0100
 - サーバーの開始時に呼び出されます。
2. QIBM_QNPS_SPLF 様式 SPLF0100
 - 既存のスパール出力ファイルを処理するために呼び出されます。

QIBM_QNPS_ENTRY 出口点は、ネットワーク・プリント・サーバーが始動されたときに、出口プログラムを実行するように定義されています。出口プログラムは、サーバーへのアクセスを検査するために使用できます。詳しくは、「印刷装置 プログラミング」(SD88-5073-02) を参照してください。

出口点 QIBM_QNPS_ENTRY 様式 ENTR0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	ネットワーク・プリント・サーバーの場合の値は QNPSEVR です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QNPS_ENTRY の場合の様式名は ENTR0100 です。
28	1C	BINARY(4)	機能 ID	実行されている機能。 QIBM_QNPS_ENTRY の場合の値は X'0802' です。
注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLESRC 内のメンバー ENPSEP により定義されます。				

QIBM_QNPS_SPLF 出口点は、ネットワーク・プリント・サーバーが、既存のスパール出力ファイルを処理する要求を受信した後で、出口プログラムを実行するように定義されています。プログラムは、スパール・ファイルに対する機能 (ファイルをファックスで送るなど) を実行するために使用できます。詳しくは、「印刷装置 プログラミング」(SD88-5073-02) を参照してください。

出口点 QIBM_QNPS_SPLF 様式 SPLF0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	ネットワーク・プリント・サーバーの場合の値は QNPSEVR です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QNPS_SPLF の場合の様式名は SPLF0100 です。
28	1C	BINARY(4)	機能 ID	実行されている機能。 QIBM_QNPS_SPLF の場合の値は X'010D' です。
32	20	CHAR(10)	ジョブ名	スパール・ファイルを作成したジョブの名前。
42	2A	CHAR(10)	ユーザー名	スパール・ファイルを作成したジョブのユーザー・プロファイル。
52	34	CHAR(6)	ジョブ番号	スパール・ファイルを作成したジョブの番号。
58	3A	CHAR(10)	スパール・ファイル名	要求されているスパール・ファイルの名前。
68	44	BINARY(4)	スパール・ファイル番号	要求されているスパール・ファイルの番号。
72	48	BINARY(4)	長さ	スパール・ファイル出口プログラム・データの長さ。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
76	4C	CHAR(*)	スプール・ファイル 出口プログラムデータ	スプール・ファイル出口プログラム・データには、出口点 QIBM_QNPS_SPLF 用として登録されている出口プログラムが使用する追加情報が含まれています。スプール・ファイル出口プログラム・データは、クライアント・アプリケーションから提供されます。
注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLLSRC 内のメンバー ENPSEP により定義されます。				

セントラル・サーバー: セントラル・サーバーには 3 つの出口点が定義されています。

1. QIBM_QZSC_LM 様式 ZSCL0100
 - ライセンス管理要求に対して呼び出されます。
2. QIBM_QZSC_SM 様式 ZSCS0100
 - システム管理要求に対して呼び出されます。
3. QIBM_QZSC_NLS 様式 ZSCN0100
 - 変換テーブル要求に対して呼び出されます。

QIBM_QZSC_LM 出口点は、セントラル・サーバーが受信したすべてのライセンス管理要求に対して、出口プログラムを実行するように定義されています。

出口プログラム QIBM_QZSC_LM 様式 ZSCL0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	セントラル・サーバーの場合の値は *CNTRLSRV です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZSC_LM の場合の様式名は ZSCL0100 です。
28	1C	BINARY(4)	要求機能	実行されている機能。 このフィールドには次のいずれかが入ります。 <ul style="list-style-type: none"> • X'1001' - ライセンスの要求 • X'1002' - ライセンスの解放 • X'1003' - ライセンス情報の検索
32	20	CHAR(255)	固有のクライアント名	固有クライアント名は、ネットワーク内で特定のワークステーションを識別するために使用されます。ライセンス製品の使用は、固有クライアント名により識別されるワークステーションに割り当てられます。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
287	11F	CHAR(8)	ライセンス・ユーザー・ハンドル	ライセンス・ユーザー・ハンドルは、ライセンス要求元とライセンス・リリース元が確実に同じになるようにするために使用されます。この値は、ライセンスが要求されたときと同じでなければなりません。
295	127	CHAR(7)	製品 ID	ライセンス使用が要求されている製品の ID。
302	12E	CHAR(4)	機能 ID	製品の機能。
306	132	CHAR(6)	リリース ID	製品または機能のバージョン、リリース、およびモディフィケーション・レベル。
312	138	BINARY(2)	情報タイプ	<p>検索する情報のタイプ。</p> <p>情報タイプ・フィールドは、ライセンス情報検索機能の場合のみ有効です。</p> <p>このフィールドには次のいずれかが入ります。</p> <ul style="list-style-type: none"> • X'0000' - 基本ライセンス情報 • X'0001' - 詳細ライセンス情報
<p>注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLESRC 内のメンバー EZSCEP により定義されます。</p>				

QIBM_QZSC_SM 出口点は、セントラル・サーバーが受信したすべてのクライアント管理要求に対して、出口プログラムを実行するように定義されています。

出口プログラム QIBM_QZSC_SM 様式 ZSCS0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	セントラル・サーバーの場合の値は *CNTRLSRV です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZSC_SM の場合の様式名は ZSCS0100 です。
28	1C	BINARY(4)	要求機能	<p>実行されている機能。</p> <p>このフィールドには次のいずれかが入ります。</p> <ul style="list-style-type: none"> • X'1101' - クライアントを活動状態に設定 • X'1102' - クライアントを非活動状態に設定
32	20	CHAR(255)	固有のクライアント名	ライセンス製品に割り当てられているクライアント・ワークステーション名。
287	11F	CHAR(255)	コミュニティー名	認証のために使用されるコミュニティー名 SNMP 構成フィールド。
542	21E	CHAR(1)	ノード・タイプ	<p>接続のタイプ。</p> <ul style="list-style-type: none"> • 3 - インターネット

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
543	21F	CHAR(255)	ノード名	ノードの名前。 ノード・タイプ 3 の場合のノード名は、IP アドレスです。
注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLLSRC 内のメンバー EZSCEP により定義されます。				

QIBM_QZSC-NLS 出口点は、変換マップを検索する要求をセントラル・サーバーが受信したときに、出口プログラムを実行するように定義されています。

出口プログラム QIBM_QZSC-NLS 様式 ZSCN0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	セントラル・サーバーの場合の値は *CNTRL SRV です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZSC-NLS の場合の様式名は ZSCN0100 です。
28	1C	BINARY(4)	要求機能	実行されている機能。 • X'1201' - 変換マップの検索
32	20	BINARY(4)	変換元 CCSID (Coded Character Set Identifier)	既存のデータの CCSID。
36	24	BINARY(4)	変換先 CCSID (Coded Character Set Identifier)	データが変換される CCSID。
40	28	BINARY(2)	変換タイプ	要求されたマッピング・タイプ。 • X'0001' - 往復 • X'0002' - 置換マッピング • X'0003' - 最適化マッピング
注: この様式は、ライブラリー QSYSINC 中のファイル H、QRPGSRC、QRPGLESRC、QLBLSRC、および QCBLLSRC 内のメンバー EZSCEP により定義されます。				

リモート・コマンドおよび分散プログラム呼び出しサーバー: リモート・コマンドおよび分散プログラム呼び出しサーバーには、出口点が 1 つ定義されています。

QIBM_QZRC_RMT 様式 CZRC0100

QIBM_QZRC_RMT 出口点は、リモート・コマンド呼び出し要求または分散プログラム呼び出し要求に対して、出口プログラムを呼び出すように定義されています。

パラメーター・フィールドの形式は、要求のタイプに応じて異なります。

リモート・コマンド要求の場合: 出口点 QIBM_QZRC_RMT 様式 CZRC0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	リモート・コマンド・サーバーの場合の値は *RMTSRV です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZRC_RMT の場合の様式名は CZRC0100 です。
28	1C	BINARY(4)	要求機能	実行されている機能。 X'1002' - リモート・コマンド
32	20	CHAR(10)	予約済み	リモート・コマンド要求には使用されません。
42	2A	CHAR(10)	予約済み	リモート・コマンド要求には使用されません。
52	34	BINARY(4)	次のフィールドの長さ	次のコマンド・ストリングの長さ。
56	38	CHAR (6000)	コマンド・ストリング	リモート・コマンド要求のコマンド・ストリング。

分散プログラム呼び出し要求の場合: 出口点 QIBM_QZRC_RMT 様式 CZRC0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	サーバーを呼び出しているユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	分散プログラム呼び出しサーバーの場合の値は *RMTSRV です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZRC_RMT の場合の様式名は CZRC0100 です。
28	1C	BINARY(4)	要求機能	実行されている機能。 X'1003' - 分散プログラム呼び出し
32	20	CHAR(10)	プログラム名	呼び出されているプログラムの名前。
42	2A	CHAR(10)	ライブラリー名	指定されたプログラムのライブラリー。
52	34	BINARY(4)	パラメーター数	プログラム呼び出しのパラメーターの合計数。これは、必ずしも後に続くパラメーターの数を示すものではありません。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
56	38	CHAR(*)	パラメーター情報	<p>指定されたプログラムに渡されているパラメーターに関する情報。パラメーター・ストリングは、パラメーターの使用目的タイプに関係なく、すべて次の形式をとります。構造の最後のフィールドは、入出力パラメーター使用目的タイプのために指定されます。</p> <ul style="list-style-type: none"> • BINARY(4) - このパラメーターのパラメーター情報の長さ • BINARY(4) - パラメーターの最大長 • BINARY(2) - パラメーター使用目的タイプ <ul style="list-style-type: none"> - 1 - 入力 - 2 - 出力 - 3 - 入出力 • CHAR(*) - パラメーター・ストリング

サインオン・サーバー: サインオン・サーバーには、出口点が 1 つ定義されています。

QIBM_QZSO_SIGNONSRV 様式 ZSOY0100

出口点 QIBM_QZSO_SIGNONSRV は、次のサインオン・サーバー要求が受信されたときに、出口点プログラムを実行するように定義されています。

- サーバー要求の開始
- サインオン情報の検索
- パスワードの変更
- 認証トークンの生成
- 他のユーザーの代理としての認証トークンの生成

出口点 QIBM_QZSO_SIGNONSRV 様式 ZSOY0100

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
0	0	CHAR(10)	ユーザー・プロファイル名	要求に関連したユーザー・プロファイルの名前。
10	A	CHAR(10)	サーバー ID	サインオン・サーバーの場合の値は *SIGNON です。
20	14	CHAR(8)	様式名	使用されているユーザー出口様式名。 QIBM_QZSO_SIGNONSRV の場合の様式名は ZSOY0100 です。

オフセット		タイプ	フィールド	説明
10 進数	16 進数			
28	1C	BINARY(4)	要求機能	実行されている機能。 <ul style="list-style-type: none"> • X'7002' - サーバー要求の開始 • X'7004' - サインオン情報の検索 • X'7005' - パスワードの変更 • X'7007' - 認証トークンの生成 • X'7008' - 他のユーザーの代理としての認証トークンの生成

例: 出口プログラム

このトピックのサンプル出口プログラムは、すべてのプログラミング上の考慮事項または手法を示すものではありませんが、設計とコーディングを始める前に、以下の例を検討してみてください。

コードに関する特記事項

IBM^(R) は、すべてのプログラミング・コードのサンプルを使用することができる非独占的な著作使用权を許諾します。お客様は、このサンプル・コードから、お客様独自の特別のニーズに合わせた類似のプログラムを作成することができます。

すべてのサンプル・コードは、例として示す目的でのみ、IBM により提供されます。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。

ここに含まれるすべてのプログラムは、現存するままの状態を提供され、いかなる保証も適用されません。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の保証の適用も一切ありません。

- 例: RPG による出口プログラムの作成
- 例: CL コマンドによる出口プログラムの作成

例: RPG による出口プログラムの作成: 次の例は、RPG* を使用してユーザー出口プログラムをセットアップする方法を示しています。

注: 法的に重要な情報については、「コードに関する特記事項」を参照してください。

```

**
** OS/400 SERVERS - SAMPLE USER EXIT PROGRAM
**
** THE FOLLOWING RPG PROGRAM UNCONDITIONALLY
** ACCEPTS ALL REQUESTS. IT CAN BE USED AS A SHELL
** FOR SPECIFIC APPLICATIONS. NOTE: REMOVE THE
** SUBROUTINES AND CASE STATEMENT ENTRIES FOR THE SERVERS
** THAT DO NOT REQUIRE
** SPECIFIC EXIT PROGRAM HANDLING FOR BETTER PERFORMANCE.
**
E*
E* NECESSARY ARRAY DEFINITIONS FOR TRANSFER FUNCTION
E* AND REMOTE SQL
E*
E          TFREQ    4096  1
E          RSREQ    4107  1

```

```

I*
I*
IPCSDTA      DS
I              1  10  USERID
I              11  20  APPLID
I*
I* SPECIFIC PARAMETERS FOR VIRTUAL PRINTER
I*
I              21  30  VPFUNC
I              31  40  VPOBJ
I              41  50  VPLIB
I              71  750VPIFN
I              76  85  VPOUTQ
I              86  95  VPQLIB
I*

I* SPECIFIC PARAMETERS FOR MESSAGING FUNCTION
I              21  30  MFFUNC
I*

I* SPECIFIC PARAMETERS FOR TRANSFER FUNCTION
I*
I              21  30  TFFUNC
I              31  40  TFOBJ
I              41  50  TFLIB
I              51  60  TFMBR
I              61  70  TFFMT
I              71  750TFLEN
I              764171  TFREQ
I*

I* SPECIFIC PARAMETERS FOR FILE SERVER
I*
I* NOTE: FSNAME MAY BE UP TO 16MB.
I* FSNLEN WILL CONTAIN THE ACTUAL SIZE OF FSNAME.
I*
I              B  21  240FSFID
I              25  32  FSFMT
I              33  33  FSREAD
I              34  34  FSWRIT
I              35  35  FSRDWR
I              36  36  FSDLT
I              B  37  400FSNLEN
I              41  296  FSNAME
I*

I* SPECIFIC PARAMETERS FOR DATA QUEUES
I*
I              21  30  DQFUNC
I              31  40  DQQ
I              41  50  DQLIB
I              70  750DQLEN
I              76  77  DQROP
I              78  820DQKLEN
I              83  338  DQKEY
I*

I* SPECIFIC PARAMETERS FOR REMOTE SQL
I*
I              21  30  RSFUNC
I              31  40  RSOBJ
I              41  50  RSLIB
I              51  51  RSCMT
I              52  52  RSMODE
I              53  53  RSCID
I              54  71  RSSTN
I              72  75  RSRSV
I              764182  RSREQ
I*

I* SPECIFIC PARAMETERS FOR NETWORK PRINT SERVER

```

```

I*
I          21 28 NPFT
I          B 29 320NPFLD
I* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT SPLF0100
I          33 42 NPJOBN
I          43 52 NPUSRN
I          53 58 NPJOB#
I          59 68 NPFILE
I          B 69 720NPFIL#
I          B 73 760NPLEN
I          77 332 NPDATA
I*
I* Data queue server:
I*
I* QIBM_QZHQ_DATA_QUEUE format ZHQ00100
I*
I          21 28 DQOFMT
I          B 29 320DQOFID
I          33 42 DQO0BJ
I          43 52 DQOLIB
I          53 54 DQOROP
I          B 55 580DQOLEN
I          59 314 DQOKEY
I*
I* Specific PARAMETERS FOR CENTRAL SERVER
I*
I          21 28 CSFMT
I          B 29 320CSFID
I* Central server:
I*
I* QIBM_QZSC_LM format ZSCL0100 for license management calls
I*
I*
I          33 287 CSLCNM
I          288 295 CSLUSR
I          296 302 CSLPID
I          303 306 CSLFID
I          307 312 CSLRID
I          B 313 3140CSLTYP
I*
I* Central server:
I*
I* QIBM_QZSC_LM format ZSCS0100 for system management calls
I*
I*
I          33 287 CSSCNM
I          288 542 CSSCMY
I          543 543 CSSNDE
I          544 798 CSSNNM
I*
I* Central server:
I*
I* QIBM_QZSC_LM format ZSCN0100 for retrieve conversion map calls
I*
I*
I          21 30 CSNXFM
I          29 320CSNFNC
I          B 33 360CSNFRM
I          B 37 400CSNTO
I          B 41 420CSNCNT
I*
I* SPECIFIC PARAMETERS FOR DATABASE SERVER
I*
I          21 28 DBFMT
I          B 29 320DBFID
I*

```

```

I* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAD0100
I          33 160 DBDFIL
I          161 170 DBDLIB
I          171 180 DBDMBR
I          181 190 DBDAUT
I          191 318 DBDBFL
I          319 328 DBDBLB
I          329 338 DBDOFL
I          339 348 DBDOLB
I          349 358 DBDOMB
I*
I* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAD0200
I          B 33 360DBNUM
I          37 46 DBLIB2
I*
I* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAQ0100
I          33 50 DBSTMT
I          51 68 DBCRSR
I          69 70 DBOPI
I          71 72 DBATTR
I          73 82 DBPKG
I          83 92 DBPLIB
I          B 93 940DBDRDA
I          95 95 DBCMT
I          96 351 DBTEXT
I* THE FOLLOWING PARAMETERS REPLACE DBTEXT FOR FORMAT ZDAQ0200
I          96 105 DBSQCL
I          B 133 1360DBSQLN
I          137 392 DBSQTX
I* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAR0100
I          33 52 DBLIBR
I          53 88 DBRDBN
I          89 108 DBPKGR
I          109 364 DBFILR
I          365 384 DBMBRR
I          385 404 DBFFT
I* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAR0200
I          33 42 DBRPLB
I          43 170 DBRPTB
I          171 180 DBRFLB
I          181 308 DBRFTB
I*
I* Remote command and distributed program call server:
I*
I* QIBM_QZRC_RMT format CZRC0100
I*   RCPGM AND RCLIB ARE NOT USED FOR REMOTE COMMAND CALLS
I*
I          21 28 RCFMT
I          B 29 320RCFID
I          33 42 RCPGM
I          43 52 RCLIB
I          B 53 560RCNUM
I          57 312 RCDATA
I*
I* signon server:
I*
I* QIBM_QZSO_SIGNONSRV format ZSOY0100 for TCP/IP signon server
I*
I          21 28 SOXFMT
I          B 29 320SOFID
I*
I*****
I*
I          '*VPRT      '      C          #VPRT
I          '*TRFCL    '      C          #TRFCL
I          '*FILESRV  '      C          #FILE
I          '*MSGFCL   '      C          #MSGF

```

```

I          '*DQSRV '          C          #DQSRV
I          '*RQSRV '          C          #RQSRV
I          '*SQL '           C          #SQL
I          '*NDB '           C          #NDBSV
I          '*SQLSRV '         C          #SQLSV
I          '*RTVOBJINF'       C          #RTVOB
I          '*DATAQSRV '       C          #DATAQ
I          'QNPSERVER '       C          #QNPSV
I          '*CNTRLSRV '       C          #CNTRL
I          '*RMTSRV '         C          #RMTSV
I          '*SIGNON '         C          #SIGN
I*
C*
C* EXIT PROGRAM CALL PARAMETERS
C*
C          *ENTRY    PLIST
C                      PARM          RTNCD  1
C                      PARM          PCSDTA
C*
C* INITIALIZE RETURN VALUE TO ACCEPT REQUEST
C*
C                      MOVE '1'      RTNCD
C*
C* COMMON PROCESSING
C*
C          COMMON LOGIC GOES HERE
C*
C* PROCESS BASED ON SERVER ID
C*
C          APPLID    CASEQ#VPRT    VPRT
C          APPLID    CASEQ#TRFCL    TFR
C          APPLID    CASEQ#FILE    FILE
C          APPLID    CASEQ#MSGF    MSG
C          APPLID    CASEQ#DQSRV    DATAQ
C          APPLID    CASEQ#RQSRV    RSQL
C          APPLID    CASEQ#SQL      SQLINT
C          APPLID    CASEQ#NDBSV    NDB
C          APPLID    CASEQ#SQLSV    SQLSRV
C          APPLID    CASEQ#RTVOB    RTVOBJ
C          APPLID    CASEQ#DATAQ    ODATAQ
C          APPLID    CASEQ#QNPSV    NETPRT
C          APPLID    CASEQ#CNTRL    CENTRL
C          APPLID    CASEQ#RMTSV    RMTCMD
C          APPLID    CASEQ#SIGN    SIGNON
C          END
C          SETON                      LR
C          RETRN
C*
C* SUBROUTINES
C*
C* VIRTUAL PRINT
C*
C          VPRT    BEGSR
C*              SPECIFIC LOGIC GOES HERE
C          ENDSR
C*
C* TRANSFER FUNCTION
C*
C* THE FOLLOWING IS AN EXAMPLE OF SPECIFIC PROCESSING
C* THAT THE EXIT PROGRAM COULD DO FOR TRANSFER FUNCTION.
C*
C* IN THIS CASE, USERS ARE NOT ALLOWED TO SELECT
C* DATA FROM ANY FILES THAT ARE IN LIBRARY QIWS.
C*
C          TFR      BEGSR
C          TFFUNC   IFEQ 'SELECT'

```

```

C          TFLIB      ANDEQ'QIWS'
C          MOVE '0'      RTNCD
C          END
C          ENDSR
C*
C* FILE SERVER
C*
C          FILE      BEGSR
C*          SPECIFIC LOGIC GOES HERE
C          ENDSR
C*
C* MESSAGING FUNCTION
C*
C          MSG      BEGSR
C*          SPECIFIC LOGIC GOFS HERE
C          ENDSR
C* DATA QUEUES
C*
C          DATAQ    BEGSR
C*          SPECIFIC LOGIC GOES HERE
C          ENDSR
C*
C* REMOTE SQL
C*
C          RSQL      BEGSR
C*          SPECIFIC LOGIC GOES HERE
C          ENDSR
C*
C* SERVERS
C*
C* DATABASE INIT
C*
C          SQLINT    BEGSR
C*          SPECIFIC LOGIC GOES HERE
C          ENDSR
C*
C* DATABASE NDB (NATIVE DATABASE)
C*
C          NDB      BEGSR
C*          SFECIFIC LOGIC GOES HERE
C          ENDSR
C*
C* DATABASE SQL
C*
C          SQLSRV    BEGSR
C*          SPECIFIC LOGIC GOES HERE
C          ENDSR
C*
C* DATABASE RETRIEVE OBJECT INFORMATION
C*
C          RTVOBJ    BEGSR
C*          SPECIFIC LOGIC GOES HERE
C          ENDSR
C*
C* DATA QUEUE SERVER
C*
C          ODATAQ    BEGSR
C*          SPECIFIC LOGIC GOES HERE
C          ENDSR
C*
C* NETWORK PRINT
C*
C          NETPRT    BEGSR
C*          SPECIFIC LOGIC GOES HERE
C          ENDSR

```

```

C*
C* CENTRAL SERVER
C*
C*
C* THE FOLLOWING IS AN EXAMPLE OF SPECIFIC PROCESSING
C* THAT THE EXIT PROGRAM COULD DO FOR LICENSE MANAGEMENT.
C*
C* IN THIS CASE, THE USER "USERALL" WILL NOT BE ALLOWED
C* TO EXECUTE ANY FUNCTIONS THAT ARE PROVIDED BY THE
C* CENTRAL SERVER FOR WHICH THIS PROGRAM IS A REGISTERED
C* EXIT PROGRAM - LICENSE INFORMATION, SYSTEM MANAGEMENT
C* OR RETRIVE A CONVERSION MAP.
C*
C          CENTRL  BEGSR
C          USERID  IFEQ 'USERALL'
C                   MOVE '0'          RTNCD
C                   ENDIF
C*                   SPECIFIC LOGIC GOES HERE
C                   ENDSR
C*
C* REMOTE COMMAND AND DISTRIBUTED PROGRAM CALL
C*
C* IN THIS CASE, THE USER "USERALL" WILL NOT BE ALLOWED
C* TO EXECUTE ANY REMOTE COMMANDS OR REMOTE PROGRAM CALLS
C*
C          RMTCMD   BEGSR
C          USERID   IFEQ 'USERALL'
C                   MOVE '0'          RTNCD
C                   ENDIF
C                   ENDSR
C*
C* SIGNON SERVER
C*
C          SIGNON   BEGSR
C*                   SPECIFIC LOGIC GOES HERE
C                   ENDSR
C

```

例: CL コマンドによる出口プログラムの作成: 次の例は、制御言語 (CL) コマンドを使用してユーザー出口プログラムをセットアップする方法を示しています。

注 : 法的に重要な情報については、「コードに関する特記事項」を参照してください。

```

/*****/
/*
/* iSeries SERVERS- SAMPLE USER EXIT PROGRAM
/*
/*
/* THE FOLLOWING CL PROGRAM UNCONDITIONALLY
/* ACCEPTS ALL REQUESTS. IT CAN BE USED AS A SHELL FOR DEVELOPING
/* EXIT PROGRAMS TAILORED FOR YOUR OPERATING ENVIRONMENT.
/*
/*
/*
/*****/

PGM PARM(&STATUS &REQUEST)

/* * * * * *
/*
/* PROGRAM CALL PARAMETER DECLARATIONS
/*
/*
/* * * * * *

DCL VAR(&STATUS) TYPE(*CHAR) LEN(1) /* Accept/Reject indicator */

DCL VAR(&REQUEST) TYPE(*CHAR) LEN(9999) /* Parm structure. LEN(9999) CL */

```

```

/*****/
/* */
/* PARAMETER DECLARES */
/* */
/*****/

/* COMMON DECLARES */
DCL VAR(&USER) TYPE(*CHAR) LEN(10)
/* User ID */
DCL VAR(&APPLIC) TYPE(*CHAR) LEN(10)
/* Server ID */
DCL VAR(&FUNCTN) TYPE(*CHAR) LEN(10) /* Function being performed */

/* VIRTUAL PRINT DECLARES */
DCL VAR(&VPOBJ) TYPE(*CHAR) LEN(10) /* Object name */
DCL VAR(&VPLIB) TYPE(*CHAR) LEN(10) /* Object library name */
DCL VAR(&VPLEN) TYPE(*DEC) LEN(5 0) /* Length of following fields*/
DCL VAR(&VPOUTQ) TYPE(*CHAR) LEN(10) /* Output queue name */
DCL VAR(&VPQLIB) TYPE(*CHAR) LEN(10) /* Output queue library name */

/* TRANSFER FUNCTION DECLARES */
DCL VAR(&TFOBJ) TYPE(*CHAR) LEN(10) /* Object name */
DCL VAR(&TFLIB) TYPE(*CHAR) LEN(10) /* Object library name */
DCL VAR(&TFMBR) TYPE(*CHAR) LEN(10) /* Member name */
DCL VAR(&TFMT) TYPE(*CHAR) LEN(10) /* Record format name */
DCL VAR(&TFLEN) TYPE(*DEC) LEN(5 0) /* Length of request */
DCL VAR(&TFREQ) TYPE(*CHAR) LEN(1925) /*Transfer request statement*/

/* FILE SERVER DECLARES */
DCL VAR(&FSFID) TYPE(*CHAR) LEN(4) /* Function identifier */
DCL VAR(&FSFMT) TYPE(*CHAR) LEN(8) /* Parameter format */
DCL VAR(&FSREAD) TYPE(*CHAR) LEN(1) /* Open for read */
DCL VAR(&FSWRITE) TYPE(*CHAR) LEN(1) /* Open for write */
DCL VAR(&FSRDWRT) TYPE(*CHAR) LEN(1) /* Open for read/write */
DCL VAR(&FSDLT) TYPE(*CHAR) LEN(1) /* Open for delete */
DCL VAR(&FSLEN) TYPE(*CHAR) LEN(4) /* fname length */
DCL VAR(&FSNAME) TYPE(*CHAR) LEN(2000) /* Qualified file name */

/* DATA QUEUE DECLARES */
DCL VAR(&DQQ) TYPE(*CHAR) LEN(10) /* Data queue name */
DCL VAR(&DQLIB) TYPE(*CHAR) LEN(10) /* Data queue library name */
DCL VAR(&DQLEN) TYPE(*DEC) LEN(5 0) /* Total request length */
DCL VAR(&DQROP) TYPE(*CHAR) LEN(2) /* Relational operator */
DCL VAR(&DQKLEN) TYPE(*DEC) LEN(5 0) /* Key length */
DCL VAR(&DQKEY) TYPE(*CHAR) LEN(256) /* Key value */

/* REMOTE SQL DECLARES */
DCL VAR(&RSOBJ) TYPE(*CHAR) LEN(10) /* Object name */
DCL VAR(&RSLIB) TYPE(*CHAR) LEN(10) /* Object library name */
DCL VAR(&RSCMT) TYPE(*CHAR) LEN(1) /* Commitment control level */
DCL VAR(&RSMODE) TYPE(*CHAR) LEN(1) /* Block/Update mode indicator*/
DCL VAR(&RSCID) TYPE(*CHAR) LEN(1) /* Cursor ID */
DCL VAR(&RSSTN) TYPE(*CHAR) LEN(18) /* Statement name */
DCL VAR(&RSRSU) TYPE(*CHAR) LEN(4) /* Reserved */
DCL VAR(&RSREQ) TYPE(*CHAR) LEN(1925) /* SQL statement */

/* NETWORK PRINT SERVER DECLARES */
DCL VAR(&NPFMT) TYPE(*CHAR) LEN(8) /* Format name */
DCL VAR(&NPFID) TYPE(*CHAR) LEN(4) /* Function identifier */
/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT SPLF0100 */

```

```

DCL VAR(&NPJOB#) TYPE(*CHAR) LEN(10)/* Job name */
DCL VAR(&NPUSRN) TYPE(*CHAR) LEN(10)/* User name */
DCL VAR(&NPJOB#) TYPE(*CHAR) LEN(6) /* Job number */
DCL VAR(&NPFILE) TYPE(*CHAR) LEN(10)/* File name */
DCL VAR(&NPFIL#) TYPE(*CHAR) LEN(4) /* File number */
DCL VAR(&NPLEN) TYPE(*CHAR) LEN(4) /* Data Length */
DCL VAR(&NPDATA) TYPE(*CHAR) LEN(2000) /* Data */

DCL VAR(&DBNUM) TYPE(*CHAR) LEN(4) /* Number of libraries */
DCL VAR(&DBLIB2) TYPE(*CHAR) LEN(10) /* Library name */

/* DATA QUEUE SERVER DECLARES */
DCL VAR(&DQFMT) TYPE(*CHAR) LEN(8) /* Format name */
DCL VAR(&DQFID) TYPE(*CHAR) LEN(4) /* Function IDENTIFIER */
DCL VAR(&DQOBJ) TYPE(*CHAR) LEN(10) /* Object name */
DCL VAR(&DQOLIB) TYPE(*CHAR) LEN(10) /* Library name */
DCL VAR(&DQOROP) TYPE(*CHAR) LEN(2) /* Relational operator */
1 DCL VAR(&DQOLEN) TYPE(*CHAR) LEN(4) /* Key length */
DCL VAR(&DQOKEY) TYPE(*CHAR) LEN(256) /* Key */

/* CENTRAL SERVER DECLARES */
DCL VAR(&CSFMT) TYPE(*CHAR) LEN(8) /* Format name */
DCL VAR(&CSFID) TYPE(*CHAR) LEN(4) /* Function identifier */
/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZSCL0100 */
DCL VAR(&CSCNAM) TYPE(*CHAR) LEN(255) /* Unique client name */
DCL VAR(&CSLUSR) TYPE(*CHAR) LEN(8) /* License users handle */
DCL VAR(&CSPID) TYPE(*CHAR) LEN(7) /* Product identification */
DCL VAR(&CSFID) TYPE(*CHAR) LEN(4) /* Feature identification */
DCL VAR(&CSRID) TYPE(*CHAR) LEN(6) /* Release identification */
DCL VAR(&CSTYPE) TYPE(*CHAR) LEN(2) /* Type of information req */
/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZSCS0100 */
DCL VAR(&CSCNAM) TYPE(*CHAR) LEN(255) /* Unique client name */
DCL VAR(&CSCMTY) TYPE(*CHAR) LEN(255) /* Community name */
DCL VAR(&CSNODE) TYPE(*CHAR) LEN(1) /* Node type */
DCL VAR(&CSNNAM) TYPE(*CHAR) LEN(255) /* Node name */
/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZSCN0100 */
DCL VAR(&CSFROM) TYPE(*CHAR) LEN(4) /* From CCSID */
DCL VAR(&CSTO) TYPE(*CHAR) LEN(4) /* To CCSID */
DCL VAR(&CSCTYP) TYPE(*CHAR) LEN(2) /* Type of conversion */
/* DATABASE SERVER DECLARES */
DCL VAR(&DBFMT) TYPE(*CHAR) LEN(8) /* Format name */
DCL VAR(&DBFID) TYPE(*CHAR) LEN(4) /* Function identifier

/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAD0100 */
DCL VAR(&DBFILE) TYPE(*CHAR) LEN(128) /* File name */
DCL VAR(&DBLIB) TYPE(*CHAR) LEN(10) /* Library name */
DCL VAR(&DBMBR) TYPE(*CHAR) LEN(10) /* Member name */
DCL VAR(&DBAUT) TYPE(*CHAR) LEN(10) /* Authority to file */
DCL VAR(&DBBFIL) TYPE(*CHAR) LEN(128) /* Based on file name */
DCL VAR(&DBBLIB) TYPE(*CHAR) LEN(10) /* Based on library name */
DCL VAR(&DBOFIL) TYPE(*CHAR) LEN(10) /* Override file name */
DCL VAR(&DBOLIB) TYPE(*CHAR) LEN(10) /* Override library name */
DCL VAR(&DBOMBR) TYPE(*CHAR) LEN(10) /* Override member name

/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAD0200 */
DCL VAR(&DBNUM) TYPE(*CHAR) LEN(4) /* Number of libraries */
DCL VAR(&DBLIB2) TYPE(*CHAR) LEN(10) /* Library name

/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAQ0100 */

```

```

DCL VAR(&DBSTMT) TYPE(*CHAR) LEN(18) /* Statement name */
DCL VAR(&DBCUSR) TYPE(*CHAR) LEN(18) /* Cursor name */
DCL VAR(&DBOPT) TYPE(*CHAR) LEN(2) /* Prepare option */
DCL VAR(&DBATTR) TYPE(*CHAR) LEN(2) /* Open attributes */
DCL VAR(&DBPKG) TYPE(*CHAR) LEN(10) /* Package name */
DCL VAR(&DBPLIB) TYPE(*CHAR) LEN(10) /* Package library name */
DCL VAR(&DBDRDA) TYPE(*CHAR) LEN(2) /* DRDA(R) indicator */
DCL VAR(&DBCMT) TYPE(*CHAR) LEN(1) /* Commit control level */
DCL VAR(&DBTEXT) TYPE(*CHAR) LEN(512) /* First 512 bytes of stmt */

```

```

/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAR0100 */
DCL VAR(&DBLIBR) TYPE(*CHAR) LEN(20) /* Library name */
DCL VAR(&DBRDBN) TYPE(*CHAR) LEN(36) /* Relational Database name */
DCL VAR(&DBPKGR) TYPE(*CHAR) LEN(20) /* Package name */
DCL VAR(&DBFILR) TYPE(*CHAR) LEN(256) /* File name (SQL alias) */
DCL VAR(&DBMBRR) TYPE(*CHAR) LEN(20) /* Member name */
DCL VAR(&DBFFMT) TYPE(*CHAR) LEN(20) /* Format name */

```

```

/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAR0200 */
DCL VAR(&DBPLIB) TYPE(*CHAR) LEN(10) /* Primary key table lib */
DCL VAR(&DBPTBL) TYPE(*CHAR) LEN(128) /* Primary key table */
DCL VAR(&DBFLIB) TYPE(*CHAR) LEN(10) /* Foreign key table lib */
DCL VAR(&DBFTBL) TYPE(*CHAR) LEN(128) /* Foreign key table */

```

```

/* REMOTE COMMAND SERVER DECLARES */
DCL VAR(&RCFMT) TYPE(*CHAR) LEN(8) /* Format name */
DCL VAR(&RCFID) TYPE(*CHAR) LEN(4) /* Function identifier */
DCL VAR(&RCPGM) TYPE(*CHAR) LEN(10) /* Program name */
DCL VAR(&RCLIB) TYPE(*CHAR) LEN(10) /* Program library name */
DCL VAR(&RCNUM) TYPE(*CHAR) LEN(4) /* Number of parms or cmdlen */

```

```

DCL VAR(&RCDATA) TYPE(*CHAR) LEN(9999) /* Command string nor parms */

```

```

/* SIGNON SERVER DECLARES */

```

```

DCL VAR(&SOFMT) TYPE(*CHAR) LEN(8) /* Format name */
DCL VAR(&SOFID) TYPE(*CHAR) LEN(4) /* Function identifier */

```

```

/*****/
/*
/* OTHER DECLARES
/*
/*****/
DCL VAR(&WRKLEN) TYPE(*CHAR) LEN(5)
DCL VAR(&DECLN) TYPE(*DEC) LEN(8 0)

```

```

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/*
/* EXTRACT THE VARIOUS PARAMETERS FROM THE STRUCTURE */
/*
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

```

```

/* HEADER */
CHGVAR VAR(&USER) VALUE(%SST(&REQUEST 1 10))
CHGVAR VAR(&APPLIC) VALUE(%SST(&REQUEST 11 10))
CHGVAR VAR(&FUNCTN) VALUE(%SST(&REQUEST 21 10))

```

```

/* VIRTUAL PRINTER */

```

```

CHGVAR VAR(&VPOBJ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&VPLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 71 5))
CHGVAR VAR(&VPLEN) VALUE(%BINARY(&WRKLEN 1 4))
CHGVAR VAR(&VPOUTQ) VALUE(%SST(&REQUEST 76 10))
CHGVAR VAR(&VPQLIB) VALUE(%SST(&REQUEST 86 10))

```

```
/* TRANSFER FUNCTION */
```

```

CHGVAR VAR(&TFOBJ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&TFLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&TFMBR) VALUE(%SST(&REQUEST 51 10))
CHGVAR VAR(&TFMT) VALUE(%SST(&REQUEST 61 10))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 71 5))
CHGVAR VAR(&TFLEN) VALUE(%BINARY(&WRKLEN 1 4))
CHGVAR VAR(&TFREQ) VALUE(%SST(&REQUEST 76 1925))

```

```
/* FILE SERVER */
```

```

CHGVAR VAR(&FSFID) VALUE(%SST(&REQUEST 21 4))
CHGVAR VAR(&FSFMT) VALUE(%SST(&REQUEST 25 8))
CHGVAR VAR(&FSREAD) VALUE(%SST(&REQUEST 33 1))
CHGVAR VAR(&FSWRITE) VALUE(%SST(&REQUEST 34 1))
CHGVAR VAR(&FSRDWRT) VALUE(%SST(&REQUEST 35 1))
CHGVAR VAR(&FSDLT) VALUE(%SST(&REQUEST 36 1))
CHGVAR VAR(&FSLEN) VALUE(%SST(&REQUEST 37 4))
CHGVAR VAR(&DECLEN) VALUE(%BINARY(&FSLEN 1 4))
CHGVAR VAR(&FSNAME) VALUE(%SST(&REQUEST 41 &DECLEN))

```

```
/* DATA QUEUES */
```

```

CHGVAR VAR(&DQQ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&DQLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 71 5))
CHGVAR VAR(&DQLEN) VALUE(%BINARY(&WRKLEN 1 4))
CHGVAR VAR(&DQROP) VALUE(%SST(&REQUEST 76 2))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 78 5))
CHGVAR VAR(&DQKLEN) VALUE(&WRKLEN)
CHGVAR VAR(&DQKEY) VALUE(%SST(&REQUEST 83 &DQKLEN))

```

```
/* REMOTE SQL */
```

```

CHGVAR VAR(&RSOBJ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&RSLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&RSCMT) VALUE(%SST(&REQUEST 51 1))
CHGVAR VAR(&RSMODE) VALUE(%SST(&REQUEST 52 1))
CHGVAR VAR(&RSCID) VALUE(%SST(&REQUEST 53 1))
CHGVAR VAR(&RSSTN) VALUE(%SST(&REQUEST 54 18))
CHGVAR VAR(&RSRSU) VALUE(%SST(&REQUEST 72 4))
CHGVAR VAR(&RSREQ) VALUE(%SST(&REQUEST 76 1925))

```

```
/* NETWORK PRINT SERVER */
```

```

CHGVAR VAR(&NPFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&NPFID) VALUE(%SST(&REQUEST 29 4))

```

```
/* IF FORMAT IS SPLF0100 */
```

```

IF COND(&NPFMT *EQ 'SPLF0100') THEN(DO)
CHGVAR VAR(&NPJOBN) VALUE(%SST(&REQUEST 33 10))
CHGVAR VAR(&NPUSRN) VALUE(%SST(&REQUEST 43 10))
CHGVAR VAR(&NPJOB#) VALUE(%SST(&REQUEST 53 6))

```

```

CHGVAR VAR(&NPFILE) VALUE(%SST(&REQUEST 59 10))
CHGVAR VAR(&NPFIL#) VALUE(%SST(&REQUEST 69 4))
CHGVAR VAR(&NPLEN) VALUE(%SST(&REQUEST 73 4))
CHGVAR VAR(&DECLEN) VALUE(%BINARY(&NPLEN 1 4))
CHGVAR VAR(&NPDATA) VALUE(%SST(&REQUEST 77 &DECLEN))
ENDDO

/* DATA QUEUE SERVER */
CHGVAR VAR(&DQFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&DQFID) VALUE(%SST(&REQUEST 29 4))
CHGVAR VAR(&DQOOBJ) VALUE(%SST(&REQUEST 33 10))
CHGVAR VAR(&DQOLIB) VALUE(%SST(&REQUEST 43 10))
CHGVAR VAR(&DQOROP) VALUE(%SST(&REQUEST 53 2))
CHGVAR VAR(&DQOLEN) VALUE(%SST(&REQUEST 55 4))
CHGVAR VAR(&DQOKEY) VALUE(%SST(&REQUEST 59 256))

/* CENTRAL SERVER */
CHGVAR VAR(&CSFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&CSFID) VALUE(%SST(&REQUEST 29 4))

/* IF FORMAT IS ZSCL0100 */
IF COND(&CSFMT *EQ 'ZSCL0100') THEN(DO)
CHGVAR VAR(&CSCNAM) VALUE(%SST(&REQUEST 33 255))
CHGVAR VAR(&CSLUSR) VALUE(%SST(&REQUEST 288 8))
CHGVAR VAR(&CSPID) VALUE(%SST(&REQUEST 296 7))
CHGVAR VAR(&CSFID) VALUE(%SST(&REQUEST 303 4))
CHGVAR VAR(&CSRID) VALUE(%SST(&REQUEST 307 6))
CHGVAR VAR(&CSTYPE) VALUE(%SST(&REQUEST 313 2))
ENDDO

/* IF FORMAT IS ZSCS0100 */
IF COND(&CSFMT *EQ 'ZSCS0100') THEN(DO)
CHGVAR VAR(&CSCNAM) VALUE(%SST(&REQUEST 33 255))
CHGVAR VAR(&CSCMTY) VALUE(%SST(&REQUEST 288 255))
CHGVAR VAR(&CSNODE) VALUE(%SST(&REQUEST 543 1))
CHGVAR VAR(&CSNNAM) VALUE(%SST(&REQUEST 544 255))
ENDDO

/* IF FORMAT IS ZSCN0100 */
IF COND(&CSFMT *EQ 'ZSCN0100') THEN(DO)
CHGVAR VAR(&CSFROM) VALUE(%SST(&REQUEST 33 4))
CHGVAR VAR(&CSTO) VALUE(%SST(&REQUEST 37 4))
CHGVAR VAR(&CSCTYP) VALUE(%SST(&REQUEST 41 2))
ENDDO

/* DATABASE SERVER */
CHGVAR VAR(&DBFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&DBFID) VALUE(%SST(&REQUEST 29 4))
/* IF FORMAT IS ZDAD0100 */
IF COND(&CSFMT *EQ 'ZDAD0100') THEN(DO)
CHGVAR VAR(&DBFILE) VALUE(%SST(&REQUEST 33 128))
CHGVAR VAR(&DBLIB) VALUE(%SST(&REQUEST 161 10))
CHGVAR VAR(&DBMBR) VALUE(%SST(&REQUEST 171 10))
CHGVAR VAR(&DBAUT) VALUE(%SST(&REQUEST 181 10))
CHGVAR VAR(&DBBFIL) VALUE(%SST(&REQUEST 191 128))
CHGVAR VAR(&DBBLIB) VALUE(%SST(&REQUEST 319 10))
CHGVAR VAR(&DBOFIL) VALUE(%SST(&REQUEST 329 10))
CHGVAR VAR(&DBOLIB) VALUE(%SST(&REQUEST 339 10))

```

```
CHGVAR VAR(&DBOMBR) VALUE(%SST(&REQUEST 349 10))
ENDDO
```

```
/* IF FORMAT IS ZDAD0200 */
IF COND(&CSFMT *EQ 'ZDAD0200') THEN(DO)
CHGVAR VAR(&DBNUM) VALUE(%SST(&REQUEST 33 4))
CHGVAR VAR(&DBLIB2) VALUE(%SST(&REQUEST 37 10))
ENDDO
```

```
/* IF FORMAT IS ZDAQ0100 */
IF COND(&CSFMT *EQ 'ZDAQ0100') THEN(DO)
CHGVAR VAR(&DBSTMT) VALUE(%SST(&REQUEST 33 18))
CHGVAR VAR(&DBCRRSR) VALUE(%SST(&REQUEST 51 18))
CHGVAR VAR(&DBOPT) VALUE(%SST(&REQUEST 69 2))
CHGVAR VAR(&DBATTR) VALUE(%SST(&REQUEST 71 2))
CHGVAR VAR(&DBPKG) VALUE(%SST(&REQUEST 73 10))
CHGVAR VAR(&DBPLIB) VALUE(%SST(&REQUEST 83 10))
CHGVAR VAR(&DBDRDA) VALUE(%SST(&REQUEST 93 2))
CHGVAR VAR(&DBCMT) VALUE(%SST(&REQUEST 95 1))
CHGVAR VAR(&DBTEXT) VALUE(%SST(&REQUEST 96 512))
ENDDO
```

```
/* IF FORMAT IS ZDAR0100 */
IF COND(&CSFMT *EQ 'ZDAR0100') THEN(DO)
CHGVAR VAR(&DBLIBR) VALUE(%SST(&REQUEST 33 20))
CHGVAR VAR(&DBRDBN) VALUE(%SST(&REQUEST 53 36))
CHGVAR VAR(&DBPKGR) VALUE(%SST(&REQUEST 69 20))
CHGVAR VAR(&DBATTR) VALUE(%SST(&REQUEST 89 20))
CHGVAR VAR(&DBFILR) VALUE(%SST(&REQUEST 109 256))
CHGVAR VAR(&DBMBRR) VALUE(%SST(&REQUEST 365 20))
CHGVAR VAR(&DBFFMT) VALUE(%SST(&REQUEST 385 20))
ENDDO
```

```
/* THE FOLLOWING PARAMETERS ADDITIONAL FOR FORMAT ZDAR0200 */
/* IF FORMAT IS ZDAR0200 */
IF COND(&CSFMT *EQ 'ZDAR0200') THEN(DO)
CHGVAR VAR(&DBPLIB) VALUE(%SST(&REQUEST 33 10))
CHGVAR VAR(&DBPTBL) VALUE(%SST(&REQUEST 43 128))
CHGVAR VAR(&DBFLIB) VALUE(%SST(&REQUEST 171 10))
CHGVAR VAR(&DBFTBL) VALUE(%SST(&REQUEST 181 128))
ENDDO
```

```
/* REMOTE COMMAND SERVER */
CHGVAR VAR(&RCFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&RCFID) VALUE(%SST(&REQUEST 29 4))
CHGVAR VAR(&RCPGM) VALUE(%SST(&REQUEST 33 10))
CHGVAR VAR(&RCLIB) VALUE(%SST(&REQUEST 43 10))
CHGVAR VAR(&RCNUM) VALUE(%SST(&REQUEST 53 4))
CHGVAR VAR(&RCDATA) VALUE(%SST(&REQUEST 57 6000))
```

```
/* SIGNON SERVER DECLARES */
CHGVAR VAR(&SOFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&SOFID) VALUE(%SST(&REQUEST 29 4))
```

```

/*****/
/*                                     */
/* BEGIN MAIN PROGRAM                 */
/*                                     */
/*****/

CHGVAR VAR(&STATUS) VALUE('1') /* INITIALIZE RETURN +
                               VALUE TO ACCEPT THE REQUEST */

/* ADD LOGIC COMMON TO ALL SERVERS */

/* PROCESS BASED ON SERVER ID */
IF COND(&APPLIC *EQ '*VPRT') THEN(GOTO CMDLBL(VPRT)) /* IF VIRTUAL PRINTER */
IF COND(&APPLIC *EQ '*TFRFCL') THEN(GOTO CMDLBL(TFR)) /* IF TRANSFER FUNCTIO*/
IF COND(&APPLIC *EQ '*FILESRV') THEN(GOTO CMDLBL(FLR)) /* IF FILE SERVERS */
IF COND(&APPLIC *EQ '*MSGFCL') THEN(GOTO CMDLBL(MSG)) /* IF MESSAGING FUNCT */
IF COND(&APPLIC *EQ '*DQSRV') THEN(GOTO CMDLBL(DATAQ)) /* IF DATA QUEUES */
IF COND(&APPLIC *EQ '*RQSRV') THEN(GOTO CMDLBL(RSQL)) /* IF REMOTE SQL */
IF COND(&APPLIC *EQ '*SQL') THEN(GOTO CMDLBL(SQLINIT)) /* IF SQL */
IF COND(&APPLIC *EQ '*NDB') THEN(GOTO CMDLBL(NDB)) /* IF NATIVE DATABASE */
IF COND(&APPLIC *EQ '*SQLSRV') THEN(GOTO CMDLBL(SQLSRV)) /* IF SQL */
IF COND(&APPLIC *EQ '*RTVOBJINF') THEN(GOTO CMDLBL(RTVOBJ)) /* IF RETRIEVE OB*/
IF COND(&APPLIC *EQ '*DATAQSRV') THEN(GOTO CMDLBL(ODATAQ)) /* IF D*/
IF COND(&APPLIC *EQ '*QNPSERV') THEN(GOTO CMDLBL(NETPRT)) /* IF NETWORK PRI*/
IF COND(&APPLIC *EQ '*CNTRLSRV') THEN(GOTO CMDLBL(CENTRAL)) /* IF CENTRAL SER*/
IF COND(&APPLIC *EQ '*RMTSRV') THEN(GOTO CMDLBL(RMTCMD)) /* IF RMTCMD/DPC */
IF COND(&APPLIC *EQ '*SIGNON') THEN(GOTO CMDLBL(SIGNON)) /* IF SIGNON */

GOTO EXIT

/* * * * * * * * * * * * * * * * * */
/* SUBROUTINES                         */
/*                                     */
/* * * * * * * * * * * * * * * * * */

/* VIRTUAL PRINTER */
VPRT:

/* SPECIFIC LOGIC GOES HERE */

GOTO EXIT
/* TRANSFER FUNCTION */
TFR:

/* SPECIFIC LOGIC GOES HERE */

GOTO EXIT

/* FILE SERVERS */
FLR:

/* SPECIFIC LOGIC GOES HERE */

GOTO EXIT
/* MESSAGING FUNCTION */
MSG:

/* SPECIFIC LOGIC GOES HERE */

GOTO EXIT
/* DATA QUEUES */

```

```

DATAQ:
/* SPECIFIC LOGIC GOES HERE */
GOTO EXIT

/* REMOTE SQL */
RSQL:
/* SPECIFIC LOGIC GOES HERE */
GOTO EXIT
/* DATABASE INIT */
SQLINIT:
/* SPECIFIC LOGIC GOES HERE */
GOTO EXIT
/* NATIVE DATABASE */
NDB:
/* SPECIFIC LOGIC GOES HERE */
GOTO EXIT
/* DATABASE SQL */
SQLSRV:
/* SPECIFIC LOGIC GOES HERE */
GOTO EXIT

/* RETRIEVE OBJECT INFORMATION */
RTVOBJ:
/* SPECIFIC LOGIC GOES HERE */
GOTO EXIT

/* DATA QUEUE SERVER */
ODATAQ:
/* SPECIFIC LOGIC GOES HERE */
GOTO EXIT
/* NETWORK PRINT SERVER */
NETPRT:
/* SPECIFIC LOGIC GOES HERE */
GOTO EXIT
/* CENTRAL SERVER */
CENTRAL:
/* SPECIFIC LOGIC GOES HERE */
GOTO EXIT
/* REMOTE COMMAND AND DISTRIBUTED PROGRAM CALL */
RMTCMD:

/* IN THIS CASE IF A USER ATTEMPTS TO DO A REMOTE COMMAND AND DISTRIBUTED */
/* PROGRAM CALL AND HAS A USERID OF userid THEY WILL NOT BE ALLOWED TO */
/* CONTINUE. */
IF COND(&USER *EQ 'userid') THEN(CHGVAR VAR(&STATUS) VALUE('0'))

```

```
GOTO EXIT
/* SIGNON SERVER */
SIGNON:

/* SPECIFIC LOGIC GOES HERE */

GOTO EXIT

EXIT:
ENDPGM
```

iSeries ネットサーバーの管理



iSeries^(TM) Access for Windows^(R) では、IBM^(R) Operating System/400^(R) (OS/400^(R)) の機能である IBM iSeries Support for Windows Network Neighborhood (iSeries ネットサーバー) を利用できます。この機能によって、ファイル・サービスとプリント・サービスを利用できます。

iSeries ネットサーバーのセットアップ、管理、および使用方法に関する詳しい説明については、「iSeries ネットサーバー」を参照してください。

ポリシーとアプリケーション管理機能を使用した制限の設定

iSeries^(TM) Access for Windows^(R) では、ネットワークを管理するための主要な手段として、アプリケーション管理とポリシーの 2 つがあります。アプリケーション管理は、iSeries ユーザー・プロファイルに基づいて制限を課するもので、iSeries ナビゲーターにより管理されます。ポリシーは、構成設定と制限を課するもので、特定の PC および個々の Windows ユーザー・プロファイルのどちらにも適用できます。したがって、ポリシーの方がアプリケーション管理よりきめ細かな管理能力を提供しますが、セットアップおよび管理の方法は各段に難しくなっています。ポリシーを使用するには、105 ページの『Microsoft システム・ポリシー・エディター』をダウンロードし、設定したポリシーの保管、検索、および適用ができるように、PC と iSeries サーバーを構成する必要があります。制限する必要がある機能がすべてアプリケーション管理可能なものであり、かつ使用している OS/400^(R) のバージョンがアプリケーション管理をサポートしている場合は、一般に、アプリケーション管理を利用するのが得策です。

V5R2 では、アプリケーション管理には、セントラル設定のサポートが追加されています。アプリケーション管理のセントラル設定サポートにより、iSeries Access for Windows が以下のポリシー・テンプレートによって制御するほとんどの機能を管理できるようになります。

- 実行時の制限 (caerestr.adm)
- 指定された接続プロパティ (config.adm)
- 構成ポリシー (caecfg.adm)

アプリケーション管理について詳しくは、「アプリケーション管理」を参照してください。

ポリシーについては、次のトピックを参照してください。

- 102 ページの『iSeries Access for Windows ポリシーの概要』
- 104 ページの『ポリシーを使用するためのシステム・セットアップ』
- 106 ページの『iSeries Access for Windows ポリシー・リスト』

iSeries Access for Windows ポリシーの概要

iSeries^(TM) Access for Windows^(R) のシステム・ポリシーを使用すると、ユーザーが特定のアクションを実行できないように制限したり、特定の構成機能を提案したり、特定の構成機能を必須指定としたりすることができます。システム・ポリシーは、個別の Windows のユーザー・プロファイルにも、特定の PC にも適用することができます。ただし、これらのポリシーは iSeries サーバー・リソースに対する制御を提供するものではなく、したがって iSeries セキュリティーの代替にはなりません。これらのポリシーを使用していることについては、103 ページの『ポリシーのタイプとスコープ』を参照してください。

iSeries Access for Windows の使用および構成を制御するためのグループ・ポリシーの使用のテストは限定されており、予期せぬ結果が生じることがあります。グループ・ポリシーについては詳しくは、Microsoft^(R) の資料を参照してください。このトピックの残りでは、iSeries Access for Windows ポリシーのテスト済みで、サポートされる使用法について説明します。

ネットワーク内でのポリシー・サポート

ポリシーは、ファイル・サーバーに常駐します。ユーザーが Windows ワークステーションにサインオンするたびに、ワークステーションでは、その Windows ユーザー・プロファイルに適用されるすべてのポリシーがダウンロードされます。ユーザーの PC は、ユーザーがそのワークステーションで何らかの作業を行う前に、それらのポリシーをレジストリーに適用します。それぞれの Windows オペレーティング・システムには、ポリシーのダウンロードに必要なコードが組み込まれています。

ポリシーの全機能を使用するには、以下のものがが必要です。

- 1 次ログオン・サーバー
- ポリシー・サーバー

ポリシー・サーバーには、Windows Network Neighborhood 用の IBM^(R) iSeries iSeries サポート (iSeries ネットサーバー) を使用できます。Windows NT/2000 および Novell Netware は、両方のタイプのサーバーとして使用できます。

詳しくは、104 ページの『ポリシーを使用するためのシステム・セットアップ』を参照してください。

ポリシー・ファイル

ポリシー定義はポリシー・テンプレートに含まれています。テンプレートは、ポリシーをカテゴリー別に分けて編成しています。iSeries Access for Windows が提供するポリシー・テンプレートは 5 つあり、それぞれに次のいずれかの機能が対応しています。

- 特定システムに対する iSeries Access for Windows の機能の制限 (sysname.adm)
- 実行時の iSeries Access for Windows の機能の制限 (caerestr.adm)
- ユーザーがインストールまたはアンインストールできるコンポーネントの制限 (caeinrst.adm)
- 特定の環境、環境内のシステム、およびシステムの構成可能値について、特定の構成設定を提案または必須指定 (config.adm)
- グローバル構成可能値を提案または必須指定 (caecfg.adm)

特定のポリシーを作成または変更するには、その前に、CWBADGEN ユーティリティーを使用して、ポリシー・テンプレートを生成しておく必要があります。次に、105 ページの『Microsoft システム・ポリシー・エディター』を使用して、テンプレートを活動化し、それぞれの構成要素となっているポリシーを設定します。ポリシー設定が終わったら、ポリシー・ファイル (たとえば (nt)config.pol) に変更を保管します。

注: 異なる Windows オペレーティング・システム用のポリシーは、別々に作成し保守する必要があります。詳しくは、Microsoft の資料を参照してください。

詳しくは、105 ページの『ポリシー・ファイルの作成』を参照してください。

ポリシーのタイプとスコープ

iSeries^(TM) Access for Windows^(R) が提供するポリシーは、制限ポリシーまたは構成ポリシーのいずれかであり、それぞれ 1 つ以上のスコープを対象として使用できます。

制限ポリシー

制限ポリシーは、通常どりのスコープにも設定でき、次の目的に使用できます。

- iSeries Access for Windows の機能またはアクションの使用を制限または許可する。
- コンポーネント、サービス・パック、アップグレード、または製品全体のインストールまたはアンインストールを制限する。
- 以下の制限を設定する。たとえば、特定タイプのデータ転送アップロードを制限すること、または、iSeries サーバーへのすべてのデータの転送の防止ポリシーを使用して、一度にすべてのタイプのデータ転送アップロードを制限することなどができます。
- 通常は選択可能になっているコントロールやオプションを隠す (つまり「ぼかし表示」にする)。
- 制限ポリシーが原因で、ユーザーが実行しようとした機能が完了できない場合に、(一般的には) コンソールやウィンドウにメッセージを表示して、ユーザーに通知する。

構成ポリシー

構成ポリシーはユーザー・スコープのみに設定できるもので、次の目的に使用できます。

- 通常、エンド・ユーザーが自分で構成する設定を事前に構成する。
- 値、通常はユーザーが使用可能または使用禁止にできる機能、環境のリスト、および接続を構成する。
- 必須指定値を「ぼかし表示」にする。構成ポリシーにより特定の値の使用が必須指定される場合、その値の入力フィールドは変更不能にされます。

構成ポリシーは、提案または必須指定のいずれかの形をとることができます。

- 提案値。この値は、ユーザーが明示的に構成していたり、アプリケーション・プログラムで設定していない場合に使用されます。この値は、iSeries Access for Windows で使用する通常のデフォルト値を効果的にオーバーライドします。ただし、この値は使用しなくてもかまいません。新しい値を指定して、この提案値をオーバーライドできます。
- 必須指定値。この値が使用されます。ユーザーやアプリケーション・プログラムは、この値を変更できません。

ポリシーのスコープ

各ポリシーは、マシン・スコープ、ユーザー・スコープ、および iSeries 接続スコープの 3 つのスコープのいずれかに設定できます。複数のスコープで設定できるポリシーもあれば、1 つのスコープだけでしか設定できないポリシーもあります。

スコープ	説明
マシン・スコープ	このスコープで設定されたポリシーは、PC のすべてのユーザーに適用されます。マシン・スコープの設定値をオーバーライドするために、特定ユーザーに対して同じポリシーが設定されている場合だけは例外です。
ユーザー・スコープ	このスコープで設定されたポリシーは、ユーザー単位で適用できます。一部のユーザーに対して設定し、他のユーザーに対しては設定しないという使い方ができます。「デフォルト・ユーザー」（個人のポリシー構成は持たないユーザー）に対しても設定できます。ユーザー・スコープ・ポリシーの中には、マシン・スコープの設定値に関係なく、機能の使用を可能にする設定値を提供するものがあります。この設定値が使用されると、マシン・スコープの設定値は無視されます。
iSeries 接続（「システム単位」）スコープ	<p>ユーザー・スコープやマシン・スコープで設定できるポリシーは、そのユーザー・スコープやマシン・スコープの中で、iSeries 接続スコープによりさらに限定された設定を行うことができます。iSeries 接続スコープで設定された場合、ポリシー設定値は、指定された iSeries システムを使用した作業時にだけ適用されます。たとえば、制限ポリシーがユーザー・スコープ内の iSeries 接続スコープで設定された場合に、iSeries システム名が SYS1 でユーザーが USER1 であるとすれば、該当の機能は、USER1 が SYS1 を使用して作業するときだけ制限されます。</p> <p>注: ポリシーが iSeries 接続スコープで設定されると、この設定値は、ユーザー・スコープ設定値やマシン・スコープ設定値よりも優先されます。たとえば、デフォルト・モードが、ユーザー USER1 に対して、「Use default user id (デフォルト ID を使用)」になるように指定されているのに、システム SYS1 に対しては、「Use Windows user id and password (Windows ユーザー ID とパスワードを使用)」になるように設定されると、USER1 と SYS1 との接続時には、そのユーザーの Windows ユーザー ID とパスワードが使用されます。USER1 と他のシステムとの接続時には、指定されているデフォルトのユーザー ID が使用されます。</p> <p>注: このスコープでのポリシーの設定を可能にするには、次のポリシー・テンプレートのいずれか 1 つまたは両方を生成する必要があります。</p> <ul style="list-style-type: none"> • config.adm — 構成済みの環境および接続テンプレート • sysname.adm — システム単位 (iSeries システム名別) テンプレート

ポリシーを使用するためのシステム・セットアップ

iSeries^(TM) Access for Windows^(R) ポリシーを使用するための手順は、次のとおりです。

1. 『ポリシーのための iSeries^(TM) サーバーの構成』
2. 『ポリシーのためのクライアント PC の構成』
3. 105 ページの『ポリシー・ファイルの作成』

ポリシーのための iSeries^(TM) サーバーの構成

ポリシーを提供するように iSeries サーバーを構成するための手順は、次に示すとおりです。この手順では、ネットワーク内に Windows^(R) PC があることが前提となっています。

- iSeries サーバーがまだ iSeries ネットサーバーとして構成されていない場合は、そのように構成する。
- ポリシー・ファイルを保持する統合ファイル・システム・フォルダーを作成する。

ポリシーのためのクライアント PC の構成

ネットワーク内のクライアント PC が iSeries^(TM) システムからポリシーのダウンロードを受け入れるようにするには、そのための構成が必要です。

➤ ネットワーク内の各 Windows^(R) ワークステーションに、作成したポリシー・ファイルをダウンロードすることができます。この操作を行うツールをダウンロードできます。cwbpoluz を

www.as400.ibm.com/clientaccess/cadownld.htm  からダウンロードします。◀

また、iSeries ログオン・サーバーで共用される NETLOGON 上にポリシー・ファイルを配置しておけば、ユーザーの PC は、ユーザーが iSeries ドメインにログオンしたときに、自動的にポリシー・ファイルをダウンロードします。

ポリシーのための Windows の構成: ネットワーク内の各 Windows^(R) ワークステーションに、作成したポリシー・ファイルをダウンロードすることができます。この操作を行うツールをダウンロードできます。

cwbpoluz を www.as400.ibm.com/clientaccess/cadownld.htm  からダウンロードします。

ポリシー・ファイルの作成

特定のポリシーを作成または変更するには、Microsoft^(R) からポリシー・エディターをダウンロードし、ポリシー・テンプレートを生成し、ポリシー・ファイルを作成または変更する必要があります。

1. 『Microsoft システム・ポリシー・エディター』。
2. 『iSeries Access for Windows ポリシー・テンプレートの作成』
3. 106 ページの『ポリシー・ファイルの作成および更新』。

注: 異なる Windows オペレーティング・システム用のポリシーは、別々に作成し保守する必要があります。詳しくは、Microsoft の資料を参照してください。

Microsoft システム・ポリシー・エディター: ユーザー独自のポリシー・ファイルを作成できるようにするには、Microsoft^(R) が提供するポリシー・エディターが必要です。ポリシー・エディターの現行バージョンは、Windows NT^(R) Server、Windows NT Workstation Resource Kit、および Office 97 Resource Kit と一緒に出荷されます。これは、Microsoft Web サイトからも入手できます。Windows^(R) 2000 の場合は、独自のバージョンのポリシー・エディターが必要ですが、これは、Windows 2000 サーバー・バージョンと一緒に出荷されます。

www.microsoft.com/japan/ 

ポリシー・エディターを検索してください。ポリシー・エディターの旧バージョンは、Windows 95 のインストール CD に入れられて出荷されています。このバージョンは使用しないでください。このバージョンでは、一度に 1 つのポリシー・テンプレートしかロードできません。

エディターに付随している以下の指示に従って、ファイルを抽出し、ポリシー・エディターとテンプレートをインストールしてください。

iSeries Access for Windows ポリシー・テンプレートの作成: iSeries^(TM) Access for Windows^(R) には、ポリシーを制御するために必要なポリシー・テンプレートを作成するプログラムが含まれています。

1. コマンド・プロンプト・ウィンドウをオープンする。
2. iSeries Access for Windows ディレクトリーに進む。これは通常、次の場所にあります。

[C:]¥Program Files¥IBM¥Client Access¥

3. 設定したいポリシーのテンプレートを指定するコマンドとパラメーターを入力する。

ポリシー・テンプレートのコマンド

コマンド cwbadgen とパラメーター	説明
<code>cwbadgen /ps S1034345</code> (ここで <code>s1034345</code> はシステム名)	システム固有のポリシー <code>S1034345.adm</code> を設定するためのテンプレートを生成する。
<code>cwbadgen /std</code>	<code>caecfg.adm</code> (グローバル構成を行う)、 <code>caeinrst.adm</code> (インストール制限を行う)、および <code>caerestr.adm</code> (実行時制限を行う) を生成する。
<code>cwbadgen /cfg config.adm</code>	<code>config.adm</code> (このコマンドを実行する PC に存在するシステム構成に基づく構成ポリシー) を生成する。/cfg 引き数の後にファイルの名前を指定してください。この例では、テンプレート・ファイルは <code>config.adm</code> です。

ポリシー・ファイルの作成および更新: ポリシー・ファイルを作成して、デフォルト・コンピューターまたはデフォルト・ユーザーの処置を制御します。

➤ 注: 次の指示は、グループ・ポリシーの使用を対象としていません。グループ・ポリシーを使用して **iSeries^(TM) Access for Windows^(R)** 機能を管理するには、Microsoft^(R) のグループ・ポリシーの使用法に関する資料を参照してください。◀

1. **poledit.exe** をダブルクリックしてポリシー・エディターを開始する。
2. 「オプション → ポリシー・テンプレート → 追加」の順に進む。
3. ポリシー・テンプレートの作成時に作成した `.adm` ファイルの保管場所に進む。
4. 追加したい `.adm` ファイルを選択して、「追加」を押す。使用したい `.adm` ファイルをすべて追加するまで、これを実行します。その後、「OK」をクリックします。
5. 「ファイル → 新規作成」を選択する。
6. ポリシーを設定して、ポリシー・ファイルを保管する。

```
¥¥QYOURSYS¥POLICIES¥ntconfig.pol
```

ここで、

- `QYOURSYS` は、iSeries ネットサーバーの名前です。
- `POLICIES` は、iSeries ネットサーバー上の共用ファイル・フォルダーの名前です。
- `config.pol` は、ポリシー・ファイルの名前です。

ポリシー・ファイルを更新するには、ポリシー・エディターを使用してポリシー・ファイルをオープンし、そのファイルを変更してから上記の位置に保管します。

注: 異なる Windows オペレーティング・システム用のポリシーは、別々に作成し保守する必要があります。詳しくは、Microsoft の資料を参照してください。

iSeries Access for Windows ポリシー・リスト

iSeries^(TM) Access for Windows^(R) は、Microsoft^(R) システム・ポリシーをサポートします。管理者は、ポリシーを使用して、それぞれのユーザーに対して使用できるようにする機能と設定値を制御できます。このトピックでは、iSeries Access for Windows が提供するすべてのポリシーをリストし、それぞれのポリシーの効果とスコープについて説明します。

➤ ポリシーの設定は、テンプレート・ファイルで定義します。iSeries Access for Windows のポリシー・テンプレート (`.adm` files) は、iSeries Access for Windows をインストールした PC で **cwbadgen** コマンド

を使用して生成できます。詳しくは、105 ページの『iSeries Access for Windows ポリシー・テンプレートの作成』を参照してください。以下のいずれかのリンクを選択して、既存のポリシー・リストを参照してください。 <<

- 『機能別のポリシー』
影響を受ける機能別にポリシーを分けたリストを示します。
- 110 ページの『テンプレート別のポリシー』
テンプレートとそれに関連したポリシーのリストを示します。

iSeries Access for Windows でのポリシーの全般的な説明については、102 ページの『iSeries Access for Windows ポリシーの概要』を参照してください。

機能別のポリシー

次の表では、iSeries^(TM) Access for Windows^(R) のポリシーを、影響を受ける機能別にリストしています。

機能	関連ポリシー
.NET Data Provider	.NET Data Provider の使用の防止
ActiveX オートメーション・オブジェクト	<ul style="list-style-type: none"> • データ転送アップロード・オートメーション・オブジェクトの防止 • データ転送ダウンロード・オートメーション・オブジェクトの防止 • リモート・コマンド・オートメーション・オブジェクトの防止 • リモート・プログラム・オートメーション・オブジェクトの防止 • データ待ち行列オートメーション・オブジェクトの防止
通信	<ul style="list-style-type: none"> • デフォルト・ユーザー・モード • TCP/IP ルックアップ • ポート・ルックアップ・モード • 必須セキュア・ソケット • 活動環境への変更の防止 • 環境リストへの変更の防止 • 未定義のシステムとの接続の防止 • 必須指定されていない環境の使用の防止 • 接続タイムアウト
データ転送: アップロード	<ul style="list-style-type: none"> • iSeries サーバーへのすべてのデータ転送の防止 • ホスト・ファイルの追加または置換の防止 • データ転送 GUI アップロードの防止 • RFROMPCB の使用の防止 • 自動開始アップロードの防止 • Excel アドイン・アップロードの防止

機能	関連ポリシー
データ転送: ダウンロード	<ul style="list-style-type: none"> • iSeries サーバーからのすべてのデータ転送の防止 • データ転送 GUI ダウンロードの防止 • RTOPCB の使用の防止 • 自動開始ダウンロードの防止 • Excel アドイン・ダウンロードの防止
データ転送: iSeries サーバー・ファイルの作成	<ul style="list-style-type: none"> • ホスト・ファイル作成の防止 • ウィザード iSeries サーバー・ファイル作成の防止 • 非ウィザード iSeries サーバー・ファイル作成の防止
ディレクトリー更新	ディレクトリー更新の使用の防止
受信リモート・コマンド	<ul style="list-style-type: none"> • システム・コマンドとして実行 • コマンド・モード • キャッシュ・セキュリティー • 総称セキュリティーの許可 • 総称セキュリティーでコマンドをログオン・ユーザーとして実行
インストール	<p>➤</p> <ul style="list-style-type: none"> • 選択セットアップのソース・ディレクトリー • セットアップの防止 • 選択セットアップの防止 • アンインストールの防止 • サービス・バック・レベル検査の防止 • サービス・バックのインストールの防止 • アップグレードの防止 • 個々のコンポーネントのインストールの防止 • アドインのインストールの防止 <p>◀</p>
ライセンス管理	ライセンスが解放されるまでの遅延時間
各国語サポート	<ul style="list-style-type: none"> • ANSI コード・ページ • OEM コード・ページ • EBCDIC コード・ページ • データの両方向 (BiDi) 変換

機能	関連ポリシー
ODBC	<ul style="list-style-type: none"> • 名前付きデータ・ソース • プログラム生成データ・ソースの防止
OLE DB	OLE DB Provider の使用の防止
iSeries ナビゲーター	iSeries ナビゲーターの使用の防止
パスワード	<p>➤</p> <ul style="list-style-type: none"> • iSeries パスワード失効前にユーザーに警告 • iSeries Access for Windows パスワード変更の防止 <p>◀</p>
PC5250 エミュレーション	<ul style="list-style-type: none"> • ディスプレイ・セッションの構成の防止 • プリンター・セッションの構成の防止 • PC5250 エミュレーターの使用の防止 • PC5250 セッションの最大数 • .WS プロファイルの変更の防止 • メニュー構成の防止 • ツールバー構成の防止 • マルチセッション構成の防止 • キーボード構成の防止 • マウス構成の防止 • Java^(TM) アプレット実行の防止 • マクロへのアクセスの防止 • エミュレーター・セッション・マネージャーへのプロファイルのインポートの防止 • エミュレーター・セッション・マネージャーでのプロファイルの削除の防止 • エミュレーター・セッション・マネージャーでのディレクトリー変更の防止

機能	関連ポリシー
PC コマンド	<ul style="list-style-type: none"> • Cwblogon • Cwbcfg • Cwback • Cwbrest • Cwbenv • cwbundbs • Wrksplf • wrkmsg • wrkppt • wrkusrj
サービス	<ul style="list-style-type: none"> • 検査時期 • 遅延時間 • 頻度 • PC へのイメージのコピー • サイレント実行 • サービス・パス • 自動開始バックグラウンド・サービス・ジョブ
ユーザー・インターフェース	デスクトップ・アイコンの作成の防止

テンプレート別のポリシー

次に示すテンプレート・ファイルは、ポリシーを制御するために使用します。詳しくは、105 ページの『iSeries Access for Windows ポリシー・テンプレートの作成』を参照してください。

テンプレート・ファイル	説明
caecfg.adm	特定の構成可能値を提案または必須指定するポリシー。これを生成するには、/std オプションを指定して cwbadgen を実行します。
caerestr.adm	iSeries TM Access for Windows の特定の機能を制限するポリシー。これを生成するには、/std オプションを指定して cwbadgen を実行します。
config.adm	特定の環境、それらの環境内のシステム、およびそれらのシステムの構成可能値について、特定の構成設定を提案または必須指定するポリシー。これを生成するには、/cfg オプションを付けて cwbadgen を実行します。
caeinrst.adm	インストールまたはアンインストールを実行できるユーザーを制限するポリシー。また、インストールに関連した他の機能も制限します。これを生成するには、/std オプションを指定して cwbadgen を実行します。
SYSNAME.adm	iSeries Access for Windows の特定機能を特定のシステムに制限するポリシー。これを生成するには、/ps オプションを付けて cwbadgen を実行します。

Secure Sockets Layer の管理

Secure Sockets Layer (SSL) は、PC クライアントがサーバーを認証できるようにし、すべてのデータと要求を暗号化する最も一般的なセキュリティー機構です。これは、クライアントとサーバーの間で機密データを転送するときに使用します。クレジット・カードや銀行の計算書情報の転送は、SSL を利用するクライアント/サーバー・トランザクションの典型的な例です。SSL には、暗号化と暗号化解除の処理が追加されるために、パフォーマンスに対する負荷が増大します。 ➤

iSeries^(TM) Access for Windows^(R) には、オプションとしてインストール可能な、Secure Sockets Layer (SSL) のサポートおよび **IBM^(R) キー管理** を使用してキー・データベースを管理する方法が組み込まれています。受信リモート・コマンドを除き、iSeries Access for Windows のすべての機能は、SSL を介して通信できます。iSeries Access for Windows は、128 ビット・レベルの暗号化を用いて、iSeries サーバーと SSL 通信を行うことができます。 ⏪

クライアントの認証は PC5250 で使用可能です。

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。

〒106-0032
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、お客様に対してなん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 49XA
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、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にお願います。

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。お客様は、IBM 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。

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。

(C) IBM Corp. 2004. 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。 (C) Copyright IBM Corp. 1999-2004. 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。

商標

以下は、IBM Corporation の商標です。

Advanced Function Presentation

AFP

Application System/400

AS/400

DB2

DB2 Universal Database

Distributed Relational Database Architecture

DRDA

e (ロゴ)

IBM

iSeries

Operating System/400

OS/2

OS/400

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。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. 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などはそれぞれ各社の商標または登録商標です。

資料に関するご使用条件

お客様がダウンロードされる資料につきましては、以下の条件にお客様が同意されることを条件にその使用が認められます。

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^(R) 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、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。

これらの資料の著作権はすべて、IBM Corporation に帰属しています。

お客様が、このサイトから資料をダウンロードまたは印刷することにより、これらの条件に同意されたものとさせていただきます。

コードに関する特記事項

本書には、プログラミングの例が含まれています。

IBM^(R) は、すべてのプログラミング・コードのサンプルを使用することができる非独占的な著作使用権を許諾します。お客様は、このサンプル・コードから、お客様独自の特別のニーズに合わせた類似のプログラムを作成することができます。

すべてのサンプル・コードは、例として示す目的でのみ、IBM により提供されます。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。

ここに含まれるすべてのプログラムは、現存するままの状態を提供され、いかなる保証も適用されません。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の保証の適用も一切ありません。



Printed in Japan