

DB2<sup>®</sup> コネクト



# 使用者の手引き

バージョン 7



DB2<sup>®</sup> コネクト



# 使用者の手引き

バージョン 7

**ご注意!**

本書、および本書がサポートする製品をご使用になる前に、247ページの『付録H. 特記事項』にある一般的な情報を必ずお読みください。

本書において、日本では発表されていない IBM 製品 (機械およびプログラム)、プログラミング、またはサービスについて言及または説明する場合があります。しかし、このことは、弊社がこのような IBM 製品、プログラミング、またはサービスを、日本で発表する意図があることを必ずしも示すものではありません。

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)

原典：	SC09-2954-00 IBM® DB2® Connect User's Guide Version 7
発行：	日本アイ・ビー・エム株式会社
担当：	ナショナル・ランゲージ・サポート

第1刷 2000.6

この文書では、平成明朝体™W3、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™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 1993, 2000. All rights reserved.

Translation: © Copyright IBM Japan 2000

# 目次

本書について . . . . .	vii	アプリケーション・サーバーと DB2 コネクト . . . . .	38
本書の対象読者 . . . . .	viii	DB2 コネクトとアプリケーション・サーバーの構成 . . . . .	39
<b>第1部 DB2 コネクトの紹介 . . . . .</b>	<b>1</b>	DB2 コネクトをトランザクション処理モニターと組み合わせて使う . . . . .	40
<b>第1章 DB2 コネクトの概要 . . . . .</b>	<b>3</b>	TP モニターの例 . . . . .	42
データベースの概念 . . . . .	4	Tuxedo と DB2 コネクト . . . . .	43
DB2 コネクトのセットアップ . . . . .	6	X/Open 分散トランザクション処理 (DTP) モデル . . . . .	43
DB2 コネクトと SQL . . . . .	6	DB2 コネクトで XA 準拠トランザクション・マネージャーを使用する方法 . . . . .	43
管理ユーティリティー . . . . .	7	<b>第4章 DB2 コネクト環境でのプログラミング 47</b>	
<b>第2章 分散リレーショナル・データベース体系の概念 . . . . .</b>	<b>9</b>	分散環境でのプログラミング . . . . .	47
DRDA と DB2 コネクト . . . . .	9	データ定義言語 (DDL) の使用 . . . . .	49
リモート作業単位 . . . . .	11	データ操作言語 (DML) の使用 . . . . .	49
分散要求 . . . . .	12	データ制御言語 (DCL) の使用 . . . . .	50
複数サイト更新 (2 フェーズ・コミット) の使用可能化 . . . . .	14	接続および切断 . . . . .	51
SPM を必要とするホストおよび AS/400 複数サイト更新のシナリオ . . . . .	15	プリコンパイル . . . . .	52
DRDA とデータ・アクセス . . . . .	20	分類順序の定義 . . . . .	54
コントロール・センターによる複数サイト更新の使用可能化 . . . . .	20	参照保全の管理 . . . . .	54
DRDA 関連のオンライン資料 . . . . .	22	ロック . . . . .	55
<b>第3章 DB2 コネクト使用のシナリオ . . . . .</b>	<b>23</b>	SQLCODE および SQLSTATE に関する相違点 . . . . .	55
直接的なデータベース・アクセス . . . . .	24	システム・カタログの使用 . . . . .	56
接続サーバーとしての DB2 コネクト エンタープライズ・エディション . . . . .	27	検索割り当てにおける数値変換オーバーフロー . . . . .	56
DB2 コネクトと Web アプリケーション . . . . .	29	分離レベル . . . . .	56
従来の CGI プログラミングの利点と限界 . . . . .	29	ストアド・プロシージャ . . . . .	57
Web サーバー上の DB2 コネクト . . . . .	30	NOT ATOMIC 複合 SQL . . . . .	61
Java アプリケーション・サーバーとしての DB2 コネクト . . . . .	32	DB2 コネクトによる複数サイト更新 . . . . .	61
Net.Data . . . . .	33	DB2 コネクトがサポートするホストまたは AS/400 サーバー SQL ステートメント . . . . .	63
IBM WebSphere . . . . .	34	DB2 コネクトが拒否するホストまたは AS/400 サーバー SQL ステートメント . . . . .	63
DB2 コネクトをアプリケーション・サーバーと組み合わせて使う . . . . .	36	DB2 ユニバーサル・データベース (OS/390 版) でのチャージバック会計の実装 . . . . .	64
アプリケーション・サーバーによるソリューション . . . . .	37	会計情報を DB2 (OS/390 版) サーバーに送信する . . . . .	66
		会計ストリングの設定 . . . . .	67

参考資料 . . . . .	67	コマンド行プロセッサ . . . . .	121
<b>第5章 独自のアプリケーションの実行 . . . . .</b>	<b>69</b>	インポートおよびエクスポート・ユーティリ ティ어의使用 . . . . .	122
データベース・ユーティリティ어의バインド	70	ワークステーションから S/390 または AS/400 データベース・サーバーへのデー タの移動 . . . . .	123
CLI/ODBC プログラムの実行 . . . . .	70	DRDA サーバーからワークステーション へのデータの移動 . . . . .	123
CLI/ODBC のアクセスに関するプラットフォーム 固有の詳細 . . . . .	72	1 バイトおよび 2 バイト・データの混合	124
詳細な構成情報 . . . . .	77	SQLQMF ユーティリティ어의置き換え	124
Java プログラムの実行 . . . . .	77		
環境の構成 . . . . .	79		
Java アプリケーション . . . . .	81		
Java アプレット . . . . .	82		
<hr/>			
<b>第2部 リファレンスおよびトラブル シューティング . . . . .</b>	<b>83</b>		
<b>第6章 データベース・ディレクトリーの更新</b>	<b>85</b>	<b>第10章 機密保護 . . . . .</b>	<b>125</b>
情報の収集 . . . . .	86	認証 . . . . .	125
ノード・ディレクトリー . . . . .	86	機密保護タイプ . . . . .	128
DCS ディレクトリー . . . . .	87	APPC 接続用の機密保護タイプ . . . . .	128
システム・データベース・ディレクトリー	96	TCP/IP 接続用の機密保護タイプ . . . . .	130
同一のデータベースへの複数項目の定義 . . . . .	97	機密保護タイプについて . . . . .	130
ディレクトリーの更新 . . . . .	97	MVS パスワードの変更 . . . . .	131
		DB2 コネクト・ワークステーションでの パスワード有効期限管理の構成 . . . . .	132
<b>第7章 アプリケーションとユーティリティ 어의バインド . . . . .</b>	<b>101</b>	ホストでのパスワード有効期限管理の構成	133
BIND コマンド . . . . .	107	機密保護に関する追加のヒント . . . . .	133
再バインド . . . . .	107	拡張機密保護コード . . . . .	134
		検査済みの TCP/IP 機密保護 . . . . .	134
<b>第8章 データベース・システム・モニター</b>	<b>109</b>	デスクトップ ODBC および Java アプリ ケーションの機密保護 . . . . .	134
リモート・クライアント接続のモニター . . . . .	109	パスワード変更サポート . . . . .	135
DB2 コネクト用のモニター・スイッチを オンにする . . . . .	110		
モニター・スイッチ状況のリスト . . . . .	110	<b>第11章 SQLCODE マッピング . . . . .</b>	<b>137</b>
GET SNAPSHOT コマンドの使用 . . . . .	110	SQLCODE マッピングをオフにする . . . . .	137
DCS アプリケーション状況のリスト . . . . .	113	SQLCODE マッピングの調整 . . . . .	137
LIST DCS APPLICATIONS . . . . .	113		
LIST DCS APPLICATIONS SHOW		<b>第12章 パフォーマンス . . . . .</b>	<b>143</b>
DETAIL . . . . .	114	パフォーマンスの概念とツール . . . . .	143
LIST DCS APPLICATIONS EXTENDED	117	データ・フロー . . . . .	143
DB2 コントロール・センターを使用して 拡張 DCS アプリケーション情報をリスト する . . . . .	118	ボトルネック . . . . .	145
Windows パフォーマンス・モニタの使用 . . . . .	119	ベンチマーク . . . . .	145
		パフォーマンス・ツール . . . . .	146
<b>第9章 管理ユーティリティ . . . . .</b>	<b>121</b>	ODBC アクセスの最適化 . . . . .	147
		アプリケーションの設計 . . . . .	148
		複合 SQL およびストアド・プロシ ジャー . . . . .	149
		要求のグループ化 . . . . .	150
		述部論理 . . . . .	150
		データのブロック化 . . . . .	150
		静的 SQL と動的 SQL . . . . .	152
		その他の SQL 考慮事項 . . . . .	152

DB2 コネクトのチューニング	153	SQL5043N	199
RQRIOBLK	153	SQL30020	200
DIR_CACHE	154	SQL30060	200
他の DB2 コネクト・パラメーター	154	SQL30061	200
接続のプール	155	CONNECT 時の SQL30073 (戻りコード 119C)	201
接続プールの働き	156	SQL30081N (戻りコード 1)	202
DB2 コネクトの接続コンセントレーター	157	SQL30081N (戻りコード 2)	203
データベースのチューニング	163	SQL30081N (戻りコード 9)	204
ネットワークのチューニング	166	SQL30081N (戻りコード 10)	204
システム・リソースの競合	170	SQL30081N (戻りコード 20)	205
パフォーマンスのトラブルシューティング	171	SQL30081N (戻りコード 27)	205
SNA パフォーマンス調整の追加のヒント	172	SQL30081N (戻りコード 79)	205
DB2 コネクトに関する一般パフォーマンス 情報	172	SQL30081N (プロトコル固有のエラー・コ ード 10032)	206
ネットワーク接続機構の選択と調整	172		
DB2 コネクトのパフォーマンスに関する その他の情報源	174		
ESCON を介した SNA のマルチ・パス・ チャンネル・サポート	174		
NCP を介する DB2 コネクト接続の調整 方法	174		
OSA-2 機能強化に関する情報	178		
その他の情報源	179		
その他の関連資料	179		
WWW の使用	180		
SNA ユーザーに関する追加のヒント	180		
<b>第13章 問題判別</b>	<b>181</b>		
その他の情報源	181		
問題判別の手引きの使用	181		
WWW の使用	181		
APPC、CPI-C、および SNA センス・コー ドの資料	182		
関係する情報の収集	182		
初期接続が正常に行われなかった場合	182		
初期接続後に発生する問題	184		
診断ツール	185		
トレース・ユーティリティ (ddcstrc)	186		
トレース構文	187		
トレース・パラメーター	188		
トレース出力	189		
トレース出力ファイルの分析	190		
よくある DB2 接続問題	196		
SQL0965 または SQL0969	197		
CONNECT 時の SQL1338	197		
CONNECT 時の SQL1403N	198		
<b>第3部 付録および後付け</b>	<b>209</b>		
<b>付録A. 以前のリリースで追加された機能</b>	<b>211</b>		
DB2 コネクト バージョン 6 リリース 1	211		
DB2 コネクト バージョン 5 リリース 2	212		
DB2 コネクト バージョン 5.0	212		
DDCS バージョン 2 リリース 4	214		
DDCS バージョン 2 リリース 3	215		
<b>付録B. ディレクトリー・カスタマイズ・ワ ークシート</b>	<b>217</b>		
<b>付録C. 各国語サポートの考慮事項</b>	<b>219</b>		
文字データの変換	219		
<b>付録D. DCE ディレクトリー・サービスの使 用</b>	<b>223</b>		
データベース・オブジェクトの作成	224		
データベース・ロケーター・オブジェクトの 作成	227		
経路指定情報オブジェクトの作成	229		
構成パラメーターの設定	230		
データベースのカタログ作成	231		
DCE ディレクトリー・サービスと機密保護	231		
<b>付録E. バック・レベル・クライアント用ユ ーティリティのバインド</b>	<b>235</b>		
<b>付録F. CLISCHEMA キーワードによる CLI/ODBC アプリケーションのパフォーマ ンス調整</b>	<b>237</b>		

ターゲット環境 . . . . .	237
CLI/ODBC . . . . .	237
DB2 の CLISHEMA 初期化キーワード . . . . .	238
使用上の注意 . . . . .	239
db2cli および bldscem ユーティリティ . . . . .	240
アプローチに関する提案事項 . . . . .	241
追加のヒント . . . . .	242
db2ocat カタログ最適化ツール . . . . .	242
その他の情報源 . . . . .	243

<b>付録G. 追加情報や関連情報の情報源 . . . . .</b>	<b>245</b>
-------------------------------------	------------

他の関連資料 . . . . .	245
------------------	-----

<b>付録H. 特記事項 . . . . .</b>	<b>247</b>
商標 . . . . .	250

<b>索引 . . . . .</b>	<b>253</b>
---------------------	------------

<b>IBM と連絡をとる . . . . .</b>	<b>263</b>
製品情報 . . . . .	263



---

## 本書について

本書には、以下に示す IBM DB2 コネクト製品の使用方法に関する一般的な情報が記載されています。

- DB2 コネクト パーソナル・エディション (OS/2 および Windows 32 ビット オペレーティング・システム版)
- DB2 コネクト エンタープライズ・エディション (EE) (AIX、HP-UX、Linux、DYNIX/ptx、Solaris、OS/2、および Windows 32 ビット オペレーティング・システム版)
- DB2 コネクト アンリミテッド・エディション (OS/390 版)

DB2 コネクト 使用者の手引きは、次の 3 つの部分に分かれています。

- 第1部 DB2 コネクトの紹介では、DB2 コネクトおよび分散関係データベース体系 (DRDA) の概念について概説し、実際にありそうな使用シナリオについて説明しています。
- 第2部 リファレンスおよびトラブルシューティングには、データベース・ディレクトリーの更新、アプリケーションのバインド、管理ユーティリティー、DB2 システム・モニター、機密保護、問題判別、およびパフォーマンスについての情報が記載されています。
- 第3部 付録には、その他の情報やヒントについて記載されています。

本書は、すべての DB2 コネクト製品に適用される概念についても説明しています。特定のプラットフォームに関する情報は次の資料を参照してください。

- OS/2 および Windows 32 ビット オペレーティング・システム上でシングルユーザー向け DB2 コネクトをセットアップする方法については、*DB2 コネクト パーソナル・エディション 概説およびインストール*
- Linux 上でシングルユーザー向け DB2 コネクトをセットアップする方法については、*DB2 コネクト パーソナル・エディション (Linux 版) 概説およびインストール*
- OS/2 または Windows 32 ビット オペレーティング・システム上でマルチユーザー向け DB2 コネクト・ゲートウェイをセットアップする方法については、*DB2 コネクト エンタープライズ・エディション (OS/2 および Windows 版) 概説およびインストール*

- AIX、HP-UX、Linux、DYNIX/ptx、または Solaris 上でマルチユーザー向け DB2 コネクトをゲートウェイをセットアップする方法については、*DB2 コネクト エンタープライズ・エディション (UNIX 版) 概説およびインストール*

---

## 本書の対象読者

本書は、DB2 コネクト接続のセットアップや保守を担当するプログラマーおよび管理者を対象にしています。ここで対象となる接続は、DB2 クライアントと、以下のいずれかの分散関係データベース体系 (DRDA) アプリケーション・サーバー・データベース管理システムとの間に存在するものです。

- DB2 ユニバーサル・データベース (OS/390 版) バージョン 5 以上
- DB2 (MVS 版) バージョン 3 以上
- DB2 (VSE および VM 版)
- DB2 ユニバーサル・データベース (AS/400 版)
- DRDA アプリケーション・サーバー機能を実行する他のすべてのリレーショナル・データベース管理システム

### 注:

1. DB2 ユニバーサル・データベース (DB2 UDB) では、ホスト・アプリケーションや AS/400 アプリケーションが DB2 UDB データへアクセスするのに DB2 コネクトを必要としません。
2. TCP/IP データベース接続および複数行の応答セットのあるストアード・プロシージャを含む、DRDA レベル 3 の機能を使用するには、DB2 ユニバーサル・データベース (OS/390 版) のバージョン 5.1 以降が必要です。
3. DB2 コネクトでサポートされている DRDA レベル 4 機能を使用するには、DB2 ユニバーサル・データベース (OS/390 版) バージョン 6.1 以上が必要です。これらの機能には、大整数、ラージ・オブジェクト、行 ID、およびユーザー定義の特殊データ・タイプのサポートが含まれます。

---

## 第1部 DB2 コネクトの紹介



---

## 第1章 DB2 コネクトの概要

DB2 コネクトは、e-business 用の IBM メインフレームや、各種の UNIX および非 UNIX 系オペレーティング・システムで動作するその他のアプリケーションに対する、非常に高速かつ堅固な接続を提供します。

DB2 コネクトには、接続に関する複数のソリューションがあります。DB2 コネクト パーソナル・エディションはホスト・データベースや AS/400 データベースへの直接接続を提供しますが、DB2 コネクト エンタープライズ・エディションは間接接続を提供し、クライアントが DB2 コネクト・サーバー経由でホスト・データベースや AS/400 データベースにアクセスできます。DB2 コネクト アンリミテッド・エディションは、製品の選択とライセンス入手が簡単に行える、ユニークなパッケージ・ソリューションを提供します。

### DB2 コネクト エンタープライズ・エディション

DB2 コネクト エンタープライズ・エディションは、複数のデスクトップ・クライアントや Web アプリケーションからホスト・システムや AS/400 サーバーで実行される DB2 データベース・サーバーへの接続を集中管理する接続サーバーです。IBM の DB2 (AS/400 版)、DB2 (OS/390 版)、DB2 (VSE および VM 版) などのデータベースは、世界中の大手企業の最重要データを管理するシステムとして最適であり、これからもそのようなシステムとして位置付けられます。しかし、これらのホスト・データベースや AS/400 データベースでデータを管理する一方で、そのデータを Windows、UNIX、OS/2 などのワークステーション上で動作するアプリケーションに統合したいという強い要望が存在しています。

DB2 コネクト エンタープライズ・エディションを利用すれば、ローカルおよびリモートのクライアント・アプリケーションで、SQL (構造化照会言語)、DB2 API (アプリケーション・プログラミング・インターフェース)、ODBC (Open Database Connectivity)、JDBC (Java Database Connectivity)、SQLJ (Embedded SQLJ for Java)、DB2 CLI (コール・レベル・インターフェース) を使い、DB2 データベースやホスト・システムの作成、更新、制御、管理を行うことが可能になります。さらに、DB2 コネクトは、Microsoft Windows のデータ・インターフェース (ActiveX Data Object (ADO)、Remote Data Object (RDO)、OLE DB など) もサポートしています。

DB2 コネクト エンタープライズ・エディションは現時点で、AIX、HP-UX、Linux、OS/2、DYNI/ptx、Solaris、および Windows 32 ビット オペレーティング・システム用が使用可能です。これらのサーバーは、OS/2、UNIX (AIX、HP-UX、Linux、DYNI/ptx、Solaris、Silicon Graphics IRIX)、および Windows 32 ビット・ワークステーション上で動作するアプリケーションをサポートします。

### **DB2 コネクト パーソナル・エディション**

DB2 コネクト パーソナル・エディションは単一のワークステーションからの、サーバー (MVS/ESA、OS/390、OS/400、VM および VSE など) 上に置かれている DB2 データベースへの、または OS/2、UNIX、および Windows 32 ビット オペレーティング・システム上にある DB2 ユニバーサル・データベース・サーバーへのアクセスを提供します。DB2 コネクト パーソナル・エディションには DB2 コネクト エンタープライズ・エディションと同じ豊富な API セットが用意されており、すべての Windows プラットフォーム上での統合化された SNA サポートを特色としています。

この製品は現時点で、OS/2、Linux、Windows 32 ビット オペレーティング・システム用が使用可能です。

### **DB2 コネクト アンリミテッド・エディション**

DB2 コネクト アンリミテッド・エディション は、DB2 コネクトを柔軟に配置することができ、製品の選択やライセンス入手も簡単に行える、ユニークなパッケージ製品です。この製品には DB2 コネクト パーソナル・エディションと DB2 コネクト エンタープライズ・エディションの両方が含まれており、ライセンス条件では、任意の DB2 コネクト製品を無制限に配置できるようになっています。ライセンス料金は、DB2 コネクトのユーザーが使用する System/390 の規模に応じて決まります。

この新しいパッケージ製品は、OS/390 システム用だけが使用可能です。また、ライセンス供与は DB2 (OS/390 版) のデータ・ソースに対してのみ有効です。

---

## **データベースの概念**

データベース という用語は、本書を通して関係データベース管理システム (RDBMS) の記述について使用されます。DB2 コネクトが通信するその他のシステムでは、若干異なる概念を記述するためにデータベースという用語が使用されます。DB2 コネクトにおいて、データベースという用語は次のものも指します。

## **MVS (バージョン 4 以前)**

DB2 (MVS/ESA 版) サブシステム。その LOCATION NAME によって識別される。

LOCATION NAME は、TSO にログインした後で、利用可能ないずれかの照会ツールを使って以下に示す SQL 照会を発行することで判別できます。

```
select current server from sysibm.sysdummy1
```

さらに、LOCATION NAME はブートストラップ・データ・セット (BSDS) でも定義されており、 DSNL004I メッセージ (LOCATION=location) に表示されます。これは、分散データ機能 (DDF) の起動時に書き込まれます。

## **OS/390 (バージョン 5 以降)**

DB2 ユニバーサル・データベース (OS/390 版) サブシステム。その LOCATION NAME によって識別される。

LOCATION NAME は、TSO にログインした後で、利用可能ないずれかの照会ツールを使って以下に示す SQL 照会を発行することで判別できます。

```
select current server from sysibm.sysdummy1
```

さらに、LOCATION NAME はブートストラップ・データ・セット (BSDS) でも定義されており、 DSNL004I メッセージ (LOCATION=location) に表示されます。これは、分散データ機能 (DDF) の起動時に書き込まれます。

**VSE** 区画内で稼働している DB2 (VSE 版)。その DBNAME によって識別される。

**VM** CMS 仮想計算機内で稼働している DB2 (VM 版)。その DBNAME によって識別される。

## **OS/400**

DB2 ユニバーサル・データベース (AS/400 版)。OS/400 オペレーティング・システムの統合された部分。AS/400 マシン上には 1 つのデータベースだけが存在することができます。そのデータベースが AS/400 システム以外のアプリケーションによって使用される場合は、そのデータベースには、関係データベース・ディレクトリー内で 1 つの名前を付与される必要があります。この名前のことを、関係データベース名 (RDB 名) といいます。

ご使用の AS/400 システムの RDB 名を表示するには、AS/400 上でコマンド **WRKRDBDIRE** を実行します。ローカル・システムの RDB 名では、そのリモート・ロケーション欄に \*LOCAL が指定されています。RDB 名の変更には、コマンド **CHGRDBDIRE** を使用します。

---

## DB2 コネクトのセットアップ

DB2 コネクトを使用する前に、以下のステップを実行する必要があります。

- ステップ 1. DB2 コネクトをインストールし、ホストまたは AS/400 サーバーとワークステーションの両方の通信を構成します。該当する *DB2 コネクト 概説* および *インストール* または *インストール* および *構成 補足* を参照してください。
- ステップ 2. データベース・ディレクトリーの更新。これは 85 ページの『第 6 章 データベース・ディレクトリーの更新』に記述されているとおりに行います。

**注:** OS/2 および Windows 32 ビット オペレーティング・システムでは、クライアント構成アシスタント (CCA) の使用をお勧めします。

その他のプラットフォームでは、データベース・ディレクトリーの更新は DB2 コマンド行プロセッサ (CLP) を使って行います。それぞれのアプローチについては、*インストール* および *構成 補足* で説明されています。

- ステップ 3. DB2 コネクト・ユーティリティーを、各ホストまたは AS/400 データベース管理システムにバインドします。101 ページの『第 7 章 アプリケーションとユーティリティーのバインド』を参照してください。

この作業は、CCA または「データ・ソースのセットアップ (Data Sources Setup)」ダイアログ (存在する場合) を使って行うことができます。

---

## DB2 コネクトと SQL

DB2 コネクトは、アプリケーション・プログラムにより実行依頼された SQL ステートメントを、ホストまたは AS/400 データベース・サーバーに転送します。DB2 コネクトは、有効な SQL ステートメントであれば、そのほとんどを送ることができます。その例外については、63 ページの『DB2 コネクトが拒否するホストまたは AS/400 サーバー SQL ステートメント』に記述されています。



組み込み SQL の処理方式には、静的 SQL と動的 SQL という 2 つのタイプがあります。静的 SQL は、事前に処理することによって、SQL ステートメントの実行に要する時間を最小化します。動的 SQL は、ホストまたは AS/400 データベース・サーバーに実行依頼された時点で処理されます。動的 SQL はより柔軟な実行が可能ですが、処理速度が遅くなってしまうことがあります。静的 SQL を使用するか、動的 SQL を使用するかは、アプリケーション・プログラマーが判断します。DB2 コネクトはそのどちらもサポートしています。

ホストまたは AS/400 データベース・サーバーが異なると、SQL の実装方法も異なります。すべての IBM システムでサポートされている共通の SQL ステートメントについては、*SQL 解説書* を参照してください。

DB2 コネクトは、共通の IBM SQL に加えて、DB2 ユニバーサル・データベース (OS/390 版)、DB2 (MVS/ESA 版)、DB2 (VSE および VM 版) (以前の SQL/DS)、および DB2 ユニバーサル・データベース (AS/400 版) で実装される SQL も完全にサポートしています。データベースの独立性を維持するには、IBM SQL の利用を強くお勧めします。詳細については、47ページの『第4章 DB2 コネクト環境でのプログラミング』を参照してください。

---

## 管理ユーティリティー

以下のユーティリティーは、DB2 コネクトの管理者に役立ちます。

- コマンド行プロセッサ。ホストまたは AS/400 データベース・サーバー・データベースに対して SQL ステートメントを発行します。コマンド行プロセッサは、指定したデータベースに SQL ステートメントを流します。
- DB2 コマンド・センターは、コマンド行プロセッサへのグラフィカル・インターフェースを提供します。
- インポートおよびエクスポート・ユーティリティーは、ワークステーション上のファイルと、ホストまたは AS/400 データベース・サーバー・データベースの間で、データのロード、インポート、およびエクスポートを実行します。これらのファイルは、次にデータをデータベース、スプレッドシート、およびワークステーションで稼働中のその他のアプリケーションにインポートするために使用することができます。インポートおよびエクスポート・ユーティリティーについて詳しくは、*データ移動ユーティリティー 手引きおよび解説書* を参照してください。
- Windows NT および Windows 2000 上で DB2 コネクト エンタープライズ・エディションを実行しているユーザーは、イベント・ビューアとパフォーマンス・モニタを使用できます。イベント・ビューアを使えば、DB2 コ

ネクトによってログに記録された例外イベントを表示することができます。パフォーマンス・モニタを使えば、DB2 コネクト・サーバーのパフォーマンスのモニターと管理をローカルまたはリモートに行うことができます。

- DB2 コントロール・センターは、DB2 コネクト・サーバーのすべての面を管理およびモニターすることができます。さらに、管理者はこれを使ってDB2 (OS/390 版) のデータベース・オブジェクト (表、視点、バッファー・プール、スレッドなど) を操作することもできます。DB2 (OS/390 版) システムを DB2 コントロール・センターから管理する方法については、アプリケーション開発の手引き を参照してください。

これらのユーティリティーの詳細については、121ページの『第9章 管理ユーティリティー』を参照してください。

さらに、データベース・システム・モニター・ユーティリティーは、システム管理者がシステム接続をモニターできるようにします。また、システム管理者がエラーの原因を判別するのに役立ちます。システム管理者は、クライアント・アプリケーションを、ホストまたは AS/400 データベース・サーバーで実行中の対応ジョブと相関させることができます。詳細については、109ページの『第8章 データベース・システム・モニター』を参照してください。

---

## 第2章 分散リレーショナル・データベース体系の概念

分散関係データベース体系 (DRDA) は、複数のデータベース・システム (IBM と非 IBM の両方)、およびアプリケーション・プログラムの協働を許可する一連のプロトコルです。DRDA を使用する関係データベース管理製品であれば、どのような組み合わせであれ接続して、分散リレーショナル・データベース管理システムを形成することができます。DRDA は、何をどのように交換すべきかを定義することによって、システム間の通信を調整します。

DB2 コネクトの説明では、作業単位という用語がよく使われます。作業単位 (UOW) とは、単一の論理トランザクションのことです。作業単位は、全操作の実行が成功した場合の SQL ステートメントの順序、もしくは全体が失敗したとみなされた場合の SQL ステートメントの順序のどちらかによって構成されます。

別の主要な概念は、分散作業単位 (複数サイト更新ともいう) です。分散作業単位 (DUOW) では、1 つの作業単位内に複数のデータベース・サーバーが関係しています。複数サイト更新 は、より具体的に、以下の特性をもつトランザクションとして定義されます。

- 作業単位ごとに 2 つ以上のデータベース管理サーバーが更新される。
- アプリケーションが作業の分散を誘導し、コミットを開始する。
- 作業単位ごとに複数の要求が存在できる。
- 1 つの要求につき 1 つのデータベース管理サーバーが存在する。
- コミットメントが複数のデータベース・サーバーにわたって調整される。

複数サイト更新について詳しくは、14ページの『複数サイト更新 (2 フェーズ・コミット) の使用可能化』を参照してください。

---

### DRDA と DB2 コネクト

DB2 コネクトは DRDA 体系を採用しており、DB2 ユニバーサル・データベース (AS/400 版)、DB2 ユニバーサル・データベース (OS/390 版)、DB2 (MVS/ESA 版)、DB2 (VSE および VM 版)、およびその他の DRDA 準拠のデータベース・サーバー内に保管されているデータにアクセスする際のコストと複雑さを減らすことができます。DRDA 体系を十分に活用することによって、DB2 コネクトは、顧客の要求するシステム管理特性を備えながら、高パフォーマンスかつ低コストの解決方法を提供することになります。

DRDA の用語で、アプリケーション・リクエスター (AR) とは、分散接続のアプリケーション側を処理するコードであり、データを要求するアプリケーションでもあります。アプリケーション・サーバー (AS) とは、接続のデータベース側を処理するコードです。DB2 コネクト環境では、DB2 コネクト・ワークステーションは、アプリケーション・プログラム用のアプリケーション・リクエスターとしてのみ機能します。

図1 は、DB2 コネクト・ワークステーションと DRDA サーバーとの間のデータの流れを示します。(ここでは、ローカル・クライアントだけがある場合の例です。) さらに、私用プロトコルが DB2 コネクト・ワークステーションとリモート・クライアントの間に存在します。

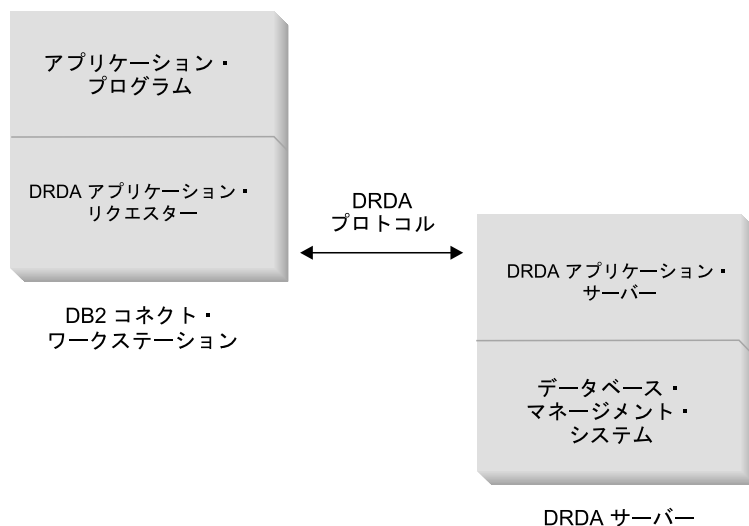


図1. DB2 コネクト・ワークステーションと DRDA サーバーの間におけるデータの流れ

DRDA サーバーのデータベース管理システムとデータベース・クライアントとの間の接続を実施するために、DRDA は以下の体系を使用します。

- 文字データ表現体系 (CDRA)
- 分散データ管理体系 (DDM)
- 定様式データ・オブジェクト内容体系 (FD:OCA)
- システム・ネットワーク体系 (SNA)
- SNA 管理サービス体系 (MSA)
- 伝送制御プロトコル / インターネット・プロトコル (TCP/IP)

これらの体系は、建築ブロックとして使われます。ネットワーク上を流れるデータ・ストリームは、DRDA 体系によって指定されます。DRDA 体系は、分散リレーショナル・データベース・アクセスをサポートする、データ・ストリーム・プロトコルを文書化したものです。

いろいろなタイプの通信情報およびアクセス中の DRDA サーバー・データベース名を記録しているディレクトリーを使用することにより、要求を正しい宛先へ差し向けることができます。

---

## リモート作業単位

リモート作業単位 は、作業単位ごとに 1 つのロケーションにおいてデータの読み取りまたは更新を行うように、ユーザーまたはアプリケーション・プログラムに指示します。それは、1 つの作業単位内での 1 つのデータベースへのアクセスをサポートします。アプリケーション・プログラムは、複数のリモート・データベースを更新することができますが、1 つの作業単位内では 1 つのデータベースだけにアクセスすることができます。

リモート作業単位には、以下のような特性があります。

- 作業単位ごとに複数の要求 (SQL ステートメント) をサポートします。
- 作業単位ごとに複数のカーソルをサポートします。
- それぞれの作業単位が更新できるデータベースは 1 つだけです。
- アプリケーション・プログラムは作業単位のコミットまたはロールバックのいずれかを行います。特定のエラー状況では、データベース・サーバーまたは DB2 コネクトが作業単位をロールバックする可能性があります。

例として、12ページの図2 には送金業務用アプリケーションが稼働しているデータベース・クライアントが示されています。このアプリケーションがアクセスするデータベースには、当座預金口座と普通預金口座の表とともに銀行業務の料率表も入っています。このアプリケーションでは次のような作業を行わなければなりません。

- 送金金額をユーザー・インターフェースから受け取る。
- その金額を普通預金口座から差し引いて、新しい残高を算出する。
- 料率表を読み込んで、算出した残高による普通預金口座に対する取り引き手数料を算出する。
- その取り引き手数料を普通預金口座から差し引く。
- 送金金額を当座預金口座に追加する。
- このトランザクション (作業単位) をコミットする。

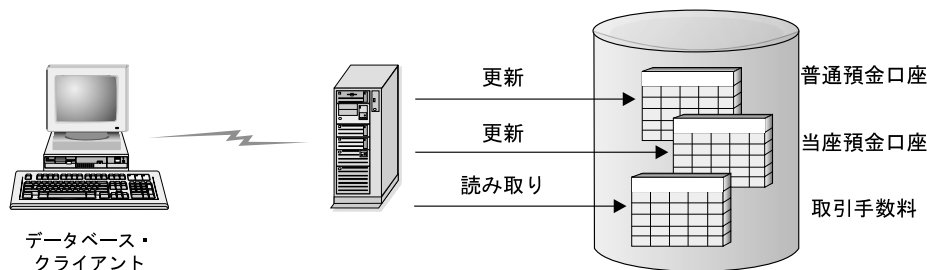


図2. 1つのトランザクションで単一のデータベースを使用する

このようなアプリケーションをセットアップするには、以下の作業が必要になります。

1. 普通預金口座、当座預金口座、銀行業務の料率表のそれぞれに対応する表を、同一のデータベース内に作成する (管理の手引き を参照)。
2. 物理的にリモートである場合は、適切な通信プロトコルを使うようデータベース・サーバーをセットアップする (概説およびインストール を参照)。
3. 物理的にリモートである場合は、ノードとデータベースをカタログ化し、そのデータベースがデータベース・サーバー上で識別されるようにする (概説およびインストール を参照)。
4. タイプ 1 接続を指定してアプリケーション・プログラムをプリコンパイルする。つまり、PREP コマンドで CONNECT(1) を指定する (アプリケーション開発の手引き を参照)。

## 分散要求

分散要求 は分散データベース機能の 1 つで、アプリケーションやユーザーが、単一のステートメントで 2 つ以上の DBMS またはデータベースを参照する SQL ステートメントを実行依頼することを可能にします。たとえば、2 つの異なる DB2 (OS/390 版) サブシステムの表の間で結合を行うことができます。

DB2 コネクト バージョン 7 は、複数のデータベースおよび DBMS 間での分散要求をサポートしています。たとえば、DB2 表と Oracle 視点との間で UNION 操作を実行することができます。サポートされている DBMS には、DB2 ファミリーの製品 (DB2 UDB (Windows、UNIX、および OS/2 版)、DB2 (OS/390 版)、DB2 (AS/400 版) など) と Oracle が含まれます。

分散要求は、データベース・オブジェクトの位置の透過性を実現するものです。(表や視点の中の) 情報が移動した場合に、その情報を要求するアプリケ

ーションには変更を加えることなく、その情報への参照 (ニックネーム という) を更新できます。分散要求はまた、すべての DB2 SQL ダイアレクトをサポートしているわけではない、あるいは特定の最適化機能をサポートしていない DBMS に対する補正 も行います。そのような DBMS では実行できない操作 (再帰的 SQL など) は、DB2 コネクトの下で実行されます。

分散要求は半自律型の 方式で機能します。たとえば、Oracle オブジェクトへの参照を含む DB2 照会を、Oracle アプリケーションが同じサーバーにアクセスしている最中に実行依頼することができます。また、分散要求は Oracle オブジェクトや他の DBMS オブジェクトへのアクセスを、独占したり (整合性やロックに関する制約以上に) 制限したりすることはありません。

分散要求機能のインプリメンテーションは、DB2 コネクト バージョン 7 の単一インスタンス (連合データベース (federated database: 複数のデータベースから構成されるが、単一のデータベース・イメージを提供するデータベースを意味します) として機能するデータベース) と、1 つまたは複数のリモート・データ・ソースで構成されています。連合データベース には、データ・ソースとその特性を識別するカタログ項目が含まれています。データ・ソース は DBMS とデータで構成されます。アプリケーションが連合データベースに接続する方法は、他の DB2 データベースに接続する方法と同じです。DB2 コネクトの連合データベースには、ユーザー・データを管理するためのライセンスは含まれていません。このデータベースの使用目的は、データ・ソースに関する情報の格納だけに限定されています。

連合システムをセットアップした後は、複数のデータ・ソース内の情報に対して、それが 1 つの大規模なデータベース内にあるかのようにアクセスできます。ある連合データベースにユーザーやアプリケーションが照会を送ると、その連合データベースが必要に応じて DB2 ファミリーおよび Oracle のシステムからデータを取り出します。ユーザーやアプリケーションは照会でニックネームを指定します。これらのニックネームはデータ・ソースの中にある表や視点への参照となるものです。エンド・ユーザーの観点からすると、ニックネームは別名のようなものです。

分散要求のパフォーマンスには多くの要因が影響します。最も重要なのは、データ・ソースとそのオブジェクトに関する正確な最新情報が、連合データベースのグローバル・カタログに格納されるようにすることです。この情報は DB2 最適化プログラムによって利用され、データ・ソースでの評価に使われる操作をプッシュダウンするかどうかを決めるのに影響します。連合システムのパフォーマンスについて詳しくは、**管理の手引き: パフォーマンス** を参照してください。

---

## 複数サイト更新 (2 フェーズ・コミット) の使用可能化

複数サイト更新 (分散作業単位 (DUOW) および 2 フェーズ・コミットともいう) は、アプリケーションが、整合性を保証しながら、複数のリモート・データベース・サーバー上のデータを更新する機能です。たとえば、ある口座から、異なるデータベース・サーバー上にある別の口座に送金する銀行業務トランザクションはその一例です。

そのようなトランザクションでは、別の口座への貸方処理に必要な更新がコミットされるまで、1 つの口座からの借方操作を行う更新をコミットしないことが非常に重要です。複数サイト更新に関する考慮事項が当てはまるのは、これらの口座を表すデータがそれぞれ別のデータベース・サーバーにより管理される場合です。

DB2 製品は、複数サイト更新を包括的にサポートしています。このサポートは、通常の SQL を使って開発されたアプリケーションだけでなく、X/Open XA インターフェイス仕様を実装したトランザクション・モニター (TP モニター) 製品を使ったアプリケーションでも利用できます。そのような TP モニター製品の例としては、IBM TxSeries (CICS および Encina)、IBM Message and Queuing Series、IBM Component Broker Series、IBM San Francisco Project、さらに Microsoft Transaction Server (MTS)、BEA Tuxedo、その他が挙げられます。ネイティブ SQL の複数サイト更新を使うか、TP モニターの複数サイト更新を使うかで、セットアップ要件は異なります。

ネイティブ SQL の複数サイト更新プログラムも TP モニターの複数サイト更新プログラムも、CONNECT 2 SYNCPOINT TWOPHASE オプションを指定してプリコンパイルする必要があります。どちらのプログラムでも、SQL Connect ステートメントを使って、後続の SQL ステートメントでどのデータベースを使用するかを指定することができます。トランザクションを調整することを DB2 に伝える (つまり、データベース接続を確立するための `xa_open` を DB2 に送信する) TP モニターがない場合は、DB2 ソフトウェアがトランザクションを調整するために使用されます。

TP モニターの複数サイト更新を使用する場合、アプリケーションは TP モニターの API (たとえば、CICS SYNCPOINT、Encina `Abort()`、MTS `SetAbort()`) を使って、コミットまたはロールバックを要求しなければなりません。

ネイティブ SQL の複数サイト更新を使用する場合は、通常の SQL COMMIT および ROLLBACK を使用します。



TP モニターの複数サイト更新は、DB2 にアクセスするトランザクションと、非 DB2 リソース・マネージャー (Oracle、Informix、SQLServer など) にアクセスするトランザクションの両方を調整できます。ネイティブ SQL の複数サイト更新は DB2 サーバーだけに使用します。

複数サイト更新トランザクションを実行するには、分散トランザクションに関与する各データベースが、分散作業単位をサポートしていなければなりません。現在、分散トランザクションに参加するための DUOW サポートを提供している DB2 サーバーは以下のとおりです。

- DB2 UDB (UNIX 版、OS/2 版、および Windows 版) V5 以降
- DB2 (MVS/ESA 版) V3.1 および 4.1
- DB2 (OS/390 版) V5.1
- DB2 ユニバーサル・データベース (OS/390 版) V6.1 以降
- DB2/400 V3.1 以降 (SNA のみ)
- DB2 Server (VM および VSE 版) V5.1 以降 (SNA のみ)
- Database Server 4

サポートされているデータベース・サーバーであれば、どんな組み合わせでも 1 つの分散トランザクションで更新することができます。たとえば、DB2 ユニバーサル・データベース (Windows NT または Windows 2000 版)、DB2 (OS/390 版) データベース、および DB2 (AS/400 版) データベースにある複数の表を、1 つのトランザクションで更新することができます。

## SPM を必要とするホストおよび AS/400 複数サイト更新のシナリオ

ホストおよび AS/400 データベース・サーバーでは、PC、UNIX、および Web アプリケーションから出された分散トランザクションに DB2 コネク트가関与する必要があります。さらに、ホストおよび AS/400 データベース・サーバーが関係した多くの複数サイト更新シナリオでは、同期点管理プログラム (SPM) 構成要素を構成する必要があります。DB2 インスタンスが作成されるときに、DB2 SPM は自動的にデフォルト設定で構成されます。

SPM が必要かどうかは、使用するプロトコル (SNA または TCP/IP) および使用する TP モニターによって決まります。以下の表では、SPM を使用する必要のあるシナリオがまとめられています。さらに、この表は Intel または UNIX マシンからホストまたは AS/400 にアクセスする場合に、DB2 コネク트가必要なことを示しています。複数サイト更新ではまた、SNA を経由する場合や、TP モニターを使用する場合に、DB2 コネク트의 SPM 構成要素が必要です。

表 1. SPM を必要とするホストおよび AS/400 複数サイト更新のシナリオ

TP モニターを使用	プロトコル	SPM が必要	必要な製品 (1 つ選択)	サポートされているホストおよび AS/400 データベース
はい	TCP/IP	はい	<ul style="list-style-type: none"> <li>• DB2 コネクト エンタープライズ・エディション</li> <li>• DB2 ユニバーサル・データベース エンタープライズ・エディション</li> <li>• DB2 ユニバーサル・データベース エンタープライズ拡張エディション</li> </ul>	<ul style="list-style-type: none"> <li>• DB2 (OS/390 版) V5.1</li> <li>• DB2 ユニバーサル・データベース (OS/390 版) V6.1 以降</li> </ul>

表1. SPM を必要とするホストおよび AS/400 複数サイト更新のシナリオ (続き)

TP モニターを使用	プロトコル	SPM が必要	必要な製品 (1 つ選択)	サポートされているホストおよび AS/400 データベース
はい	SNA	はい	<ul style="list-style-type: none"> <li>• DB2 コネクト エンタープライズ・エディション*</li> <li>• DB2 ユニバーサル・データベース エンタープライズ・エディション*</li> <li>• DB2 ユニバーサル・データベース エンタープライズ拡張エディション</li> </ul> <p>注: *AIX、OS/2、Windows NT、および Windows 2000 プラットフォームのみ。</p>	<ul style="list-style-type: none"> <li>• DB2 (MVS/ESA 版) V3.1 および 4.1</li> <li>• DB2 (OS/390 版) V5.1</li> <li>• DB2 ユニバーサル・データベース (OS/390 版) V6.1 以降</li> <li>• DB2 (AS/400 版) V3.1 以降 (ただし、以前の名称は DB2 AS/400 版)</li> <li>• DB2 Server (VSE および VM 版) V5.1 以降</li> </ul>

表 1. SPM を必要とするホストおよび AS/400 複数サイト更新のシナリオ (続き)

TP モニターを使用	プロトコル	SPM が必要	必要な製品 (1 つ選択)	サポートされているホストおよび AS/400 データベース
いいえ	TCP/IP	いいえ	<ul style="list-style-type: none"> <li>• DB2 コネクト パーソナル・エディション</li> <li>• DB2 コネクト エンタープライズ・エディション</li> <li>• DB2 ユニバーサル・データベース エンタープライズ・エディション</li> <li>• DB2 ユニバーサル・データベース エンタープライズ拡張エディション</li> </ul>	<ul style="list-style-type: none"> <li>• DB2 (OS/390 版) V5.1</li> <li>• DB2 ユニバーサル・データベース (OS/390 版) V6.1 以降</li> </ul>

表 1. SPM を必要とするホストおよび AS/400 複数サイト更新のシナリオ (続き)

TP モニターを使用	プロトコル	SPM が必要	必要な製品 (1 つ選択)	サポートされているホストおよび AS/400 データベース
いいえ	SNA	はい	<ul style="list-style-type: none"> <li>• DB2 コネクト エンタープライズ・エディション*</li> <li>• DB2 ユニバーサル・データベース エンタープライズ・エディション*</li> <li>• DB2 ユニバーサル・データベース エンタープライズ拡張エディション</li> </ul> <p>注: *AIX、OS/2、Windows NT、および Windows 2000 プラットフォームのみ。</p>	<ul style="list-style-type: none"> <li>• DB2 (MVS/ESA 版) V3.1 および 4.1</li> <li>• DB2 (OS/390 版) V5.1</li> <li>• DB2 ユニバーサル・データベース (OS/390 版) V6.1 以降</li> <li>• DB2 (AS/400 版) V3.1 以降 (ただし、以前の名称は DB2 AS/400 版)</li> <li>• DB2 Server (VSE および VM 版) V5.1 以降</li> </ul>

**注:** サポートされているデータベース・サーバーであれば、どんな組み合わせでも 1 つの分散トランザクションで更新することができます。たとえば、DB2 UDB (Windows NT 版)、DB2 (OS/390 版) データベース、および DB2 (AS/400 版) データベースにある複数の表を、1 つのトランザクションで更新することができます。

2 フェーズ・コミットに関する詳細、および一般的ないくつかの TP モニターのセットアップ手順については、**管理の手引き** を参照してください。

また、Web 上の DB2 Product and Service Technical Library にアクセスすることもできます。

1. <http://www.ibm.com/software/data/db2/library/> にアクセスします。
2. 「**DB2 Universal Database**」リンクを選択します。
3. 検索キーワード「DDCS」、「SPM」、「MTS」、「CICS」、「ENCINA」を使用して、「Technotes」を検索します。

## DRDA とデータ・アクセス

DRDA は、データベース通信プロトコルを定義しますが、アプリケーション・プログラマーが使用するプログラミング・インターフェース、つまり API は定義しません。一般に、DRDA はアプリケーション・プログラムによって使用され、宛先 DRDA サーバーが実行できる要求を渡します。現在使用可能なすべての DRDA サーバーは、アプリケーション・プログラムによって DB2 コネクト経由で送られる SQL 要求を実行することができます。

IBM は、Windows、OS/2、およびいくつかの UNIX プラットフォーム用の SQL 要求を生成するためのツールを、アプリケーション・プログラマーに提供します。これらのツールは、DB2 アプリケーション開発クライアントです。DB2 アプリケーション開発クライアントは複数の API タイプ (組み込み SQL、JDBC、SQLJ、DB2 コール・レベル・インターフェース (DB2 CLI) など) をサポートしています。これらの API は、各種のプログラミング言語でアプリケーションを構築するプログラマーが使用します。これらの API の詳細については、アプリケーション構築の手引き を参照してください。

アプリケーション開発者は、他社が提供する API を使用することもできます。たとえば、Microsoft ODBC および ADO はデータベース・アプリケーションを開発するために Windows アプリケーション・プログラマーによって使用されています。DB2 コネクトには、ODBC および ADO API を使って開発されたアプリケーションをサポートする、ODBC ドライバーと OLE DB Provider が用意されています。ODBC アプリケーション開発用のツールは、IBM ではなく、Microsoft Corporation が提供しています。

## コントロール・センターによる複数サイト更新の使用可能化

コントロール・センターを使って複数サイト更新を行えます。以下に示すように、その手順は簡単なものです。複数サイト更新の構成手順 (システムを手操作で構成する方法も含む) について詳しくは、オンラインのコネクティビティ 補足 を参照してください。

### 複数サイト更新ウィザードの開始

コントロール・センターで [+] 符号をクリックし、ツリー視点を展開します。構成するインスタンスを右マウス・ボタンで選択します。ポップアップ・メニュー

ューが開きます。「複数サイト更新 (Multisite Update)」→「構成 (Configure)」の順にメニュー項目を選択します。

### ウィザードのステップ

このウィザードはノートブック形式のインターフェースになっています。ウィザードの各ページでは、構成対象に関する特定の情報を入力するよう要求されます。以下に、これらのページを表示される順に示します。

ステップ 1. 「トランザクション処理プログラム (TP) モニターの指定 (Specify a Transaction Processor (TP) monitor)」。

このフィールドには、使用可能にした TP モニターの省略時値が表示されます。TP モニターを使用しないときは、「**TP モニターを使用しない (Do Not Use a TP Monitor)**」を選択します。

ステップ 2. 「使用する通信プロトコルの指定 (Specify the communications protocols you will use)」。

ステップ 3. 「トランザクション・マネージャー・データベースの指定 (Specify a Transaction Manager database)」。

このパネルでは、(IST\_CONN) に接続する最初のデータベースが省略時指定されています。この省略時値をそのままにしておくか、カタログ化されている別のデータベースを選択することができます。

ステップ 4. 「更新に関するデータベース・サーバーのタイプ、および TCP/IP を排他的に使用するかどうかの指定 (Specify the types of database servers involved in the update, and also whether or not TCP/IP is to be used exclusively)」。

ステップ 5. 「同期点管理プログラムの設定値の指定 (Specify the Syncpoint Manager settings)」。

このページが表示されるのは、前のページの設定で、複数サイト更新のシナリオで DB2 の同期点管理プログラムを使う必要があることが示されている場合だけです。

### 複数サイト更新機能のテスト

ステップ 1. インスタンスを右マウス・ボタンで選択し、ポップアップ・メニューから「複数サイト更新 (Multisite Update)」→「テスト (Test)」の順にメニュー・オプションを選択します。「複数サイト更新のテスト (Test Multisite Update)」ウィンドウが開きます。

ステップ 2. 「使用可能データベース (Available databases)」リスト・ボックスにある使用可能データベースから、テストするデータベースを選択します。選択したデータベースを「**選択済みデータベース**

**(Selected databases)** リスト・ボックスに入れたり出したりするには、2 つのリスト・ボックスの間にある矢印ボタンを使います。また、選択したユーザー ID およびパスワードを変更したい場合は、「**選択済みデータベース (Selected databases)**」リスト・ボックスで直接編集することが可能です。

ステップ 3. 選択が済んだら、ウィンドウの最下部にある「**了解 (OK)**」をクリックします。「複数サイト更新のテスト結果 (Multisite Update Test Result)」ウィンドウがオープンします。

ステップ 4. 「複数サイト更新のテスト結果 (Multisite Update Test Result)」ウィンドウには、選択したデータベースのうち、どれが更新のテストに成功し、また失敗したかが表示されます。このウィンドウには、失敗したデータベースの SQL コードとエラー・メッセージも表示されます。

---

## DRDA 関連のオンライン資料

以下に示すオンライン資料には、DRDA に関連した有用な情報が収められています。

**AS/400** に関する情報:

<http://www.as400.ibm.com/db2/v4r4book.htm>

**OS/390** に関する情報:

<http://www.ibm.com/software/data/db2/os390/library.html>

**DataJoiner** に関する情報:

<http://www.ibm.com/software/data/datajoiner/library.html>

**データベース / データ管理に関するオンライン資料:**

<http://www.ibm.com/software/data/pubs/>



---

## 第3章 DB2 コネクト使用のシナリオ

DB2 コネクトには、ホスト・データベースや AS/400 データベースのアクセス要件に応じた多様なソリューションが用意されています。ここでは、特定の要件や環境に適用できるいくつかのシナリオの概要を説明します。

*DB2 コネクト パーソナル・エディション* は、Windows 32 ビット・オペレーティング・システム、Linux、OS/2 の単一ワークステーションを、S/390 または AS/400 データベースに接続するのに使用します。DB2 コネクト パーソナル・エディションが最も適しているのは、ネイティブの TCP/IP サポートがデータベース・サーバーによって提供されており、配置するアプリケーションが従来の 2 層クライアント / サーバー・アプリケーションであるような環境です。

たとえば、従来の 2 層からなる VisualBasic および Microsoft Access アプリケーションを利用する場合は DB2 コネクト パーソナル・エディションが適しています。アプリケーションが中間層アプリケーション・サーバーを必要とする場合は、DB2 コネクト エンタープライズ・エディションを使わなければなりません。DB2 コネクト パーソナル・エディションを使った配置のシナリオについては、24 ページの『直接的なデータベース・アクセス』を参照してください。

*DB2 コネクト エンタープライズ・エディション* は多くの場合、DB2 クライアントをホストまたは AS/400 データベースに接続するために、中間サーバーにインストールされます。さらに、複数のローカル・ユーザーがホスト・サーバーまたは AS/400 サーバーに直接アクセスする必要があるマシンでも使われます。

たとえば、DB2 コネクト エンタープライズ・エディションは多数のローカル・ユーザーを有する大型マシンにインストールすることができます。さらに、Web サーバーや Transaction Processor (TP) モニター、およびローカルの SQL アプリケーション・プロセスやスレッドが複数ある他の 3 層アプリケーション・サーバーにインストールすることもできます。これらのケースでは、システムを単純にしておくため DB2 コネクト エンタープライズ・エディションを同一のマシンにインストールすることもできれば、CPU サイクルの負荷を減らすために別個のマシンにインストールすることも可能です。

DB2 コネクト エンタープライズ・エディションが最も適しているのは、以下に示すような環境です。

- ホストと AS/400 データベース・サーバーがネイティブの TCP/IP 接続をサポートしておらず、デスクトップ・ワークステーションからの SNA を介した直接接続が望ましくない。27ページの『接続サーバーとしての DB2 コネクト エンタープライズ・エディション』を参照してください。
- Web サーバーが Web ベースのアプリケーションを実行する。29ページの『DB2 コネクトと Web アプリケーション』を参照してください。
- Web サーバーが、データ指向の Java アプレットを使った Web ベースのアプリケーションを実行する。
- 中間層アプリケーション・サーバーが使われている。36ページの『DB2 コネクトをアプリケーション・サーバーと組み合わせて使う』を参照してください。
- TP モニター (CICS、Encina、Microsoft Transaction Server (MTS)、Tuxedo、Component Broker、MQSeries など) が使われている。40ページの『DB2 コネクトをトランザクション処理モニターと組み合わせて使う』を参照してください。

DB2 コネクト アンリミテッド・エディション は、DB2 コネクトを柔軟に配置することができ、製品の選択やライセンス入手も簡単に行える、ユニークなパッケージ製品です。この製品には DB2 コネクト パーソナル・エディションと DB2 コネクト エンタープライズ・エディションの両方が含まれており、ライセンス条件では、任意の DB2 コネクト製品を無制限に配置できるようになっています。ライセンス料金は、DB2 コネクトのユーザーが使用する System/390 の規模に応じて決まります。この新しいパッケージ製品は、OS/390 システム用だけが使用可能です。また、ライセンス供与は DB2 (OS/390 版) のデータ・ソースに対してのみ有効です。

---

## 直接的なデータベース・アクセス

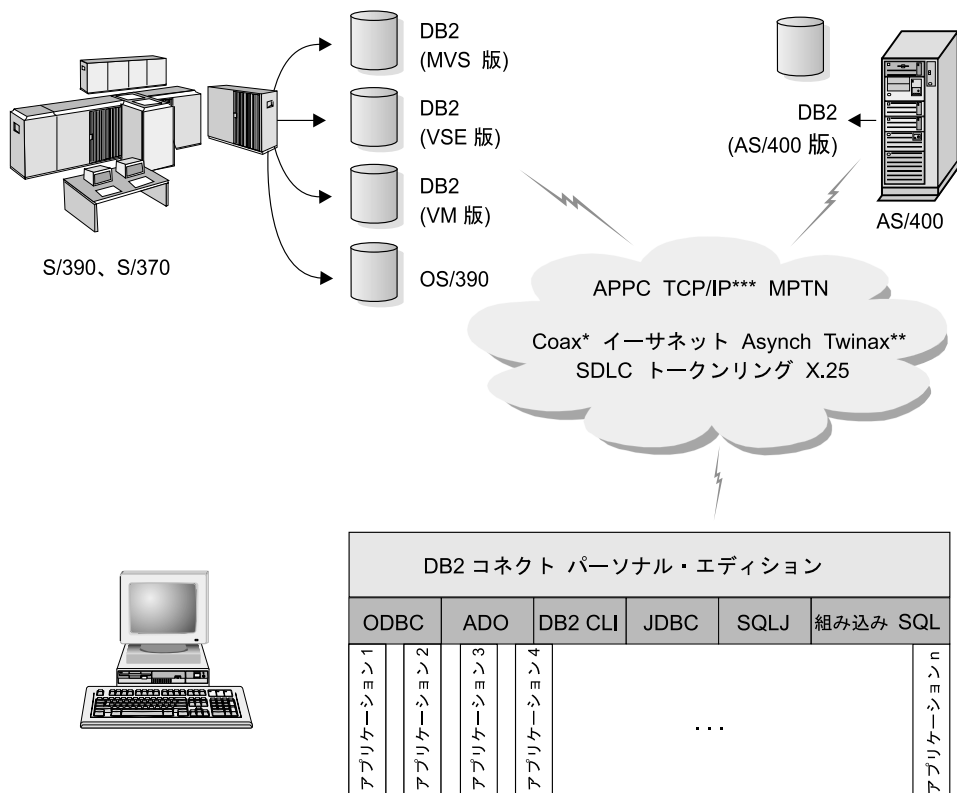
DB2 コネクトの基本機能は、Windows 32 ビット・システム、Linux、または OS/2 のワークステーション上で稼働しているデスクトップ・アプリケーションから、ホスト・データベースへの直接接続を提供することです。DB2 コネクト パーソナル・エディションを利用すれば、このソリューションを最も簡単に実現できます。

DB2 コネクト パーソナル・エディションがインストールされているそれぞれのワークステーションは、DB2 (OS/390 版)、DB2/400、DB2 UDB (Windows NT 版、Windows 2000 版、UNIX 版、OS/2 版) の各サーバーへの TCP/IP に

よる直接接続を確立できます。さらに、アプリケーションが同一トランザクション内にある DB2 ファミリーの複数のデータベースへ接続して更新することも可能です。このとき、2 フェーズ・コミットのプロトコルにより、データの整合性が完全に保たれます。

DB2 コネクト パーソナル・エディションは統合化された APPC サポートも提供しており、DB2 (MVS 版) や、APPC を必要とするその他のホスト・データベースや AS/400 データベースと通信することができます。ただし、ネイティブの TCP/IP サポートが利用できる場合には、SNA ではなく TCP/IP を使用することを強くお勧めします。

26ページの図3 では、ホストまたは AS/400 のデータベース・サーバーに直接接続されたワークステーションを示しています。それぞれのワークステーションには DB2 コネクト パーソナル・エディションがインストールされています。



すべてのプロトコルがすべてのプラットフォームでサポートされているわけではありません。

\* ホスト接続の場合のみ

\*\* AS/400 の場合

\*\*\* TCP/IP 接続には DB2 (OS/390 版) V5R1、DB2 (AS/400 版) V4R2、DB2 (VM 版) V6.1 のいずれかが必要

図 3. DB2 コネクトと、ホストまたは AS/400 のデータベース・サーバーとの直接接続

**注:**

1. 特定の DRDA AR とそれに対応するホストおよび AS/400 DRDA でどのプロトコルがサポートされるかについては、該当する **DB2 コネクト 概説** および **インストール** を参照してください。
2. DB2 ユニバーサル・データベースを DB2 コネクト・ワークステーションにインストールする必要はありません。DB2 コネクト・ワークステーションでの完全な関係データベース管理システムを希望される場合は、DB2 ユニバーサル・データベースをご注文ください。

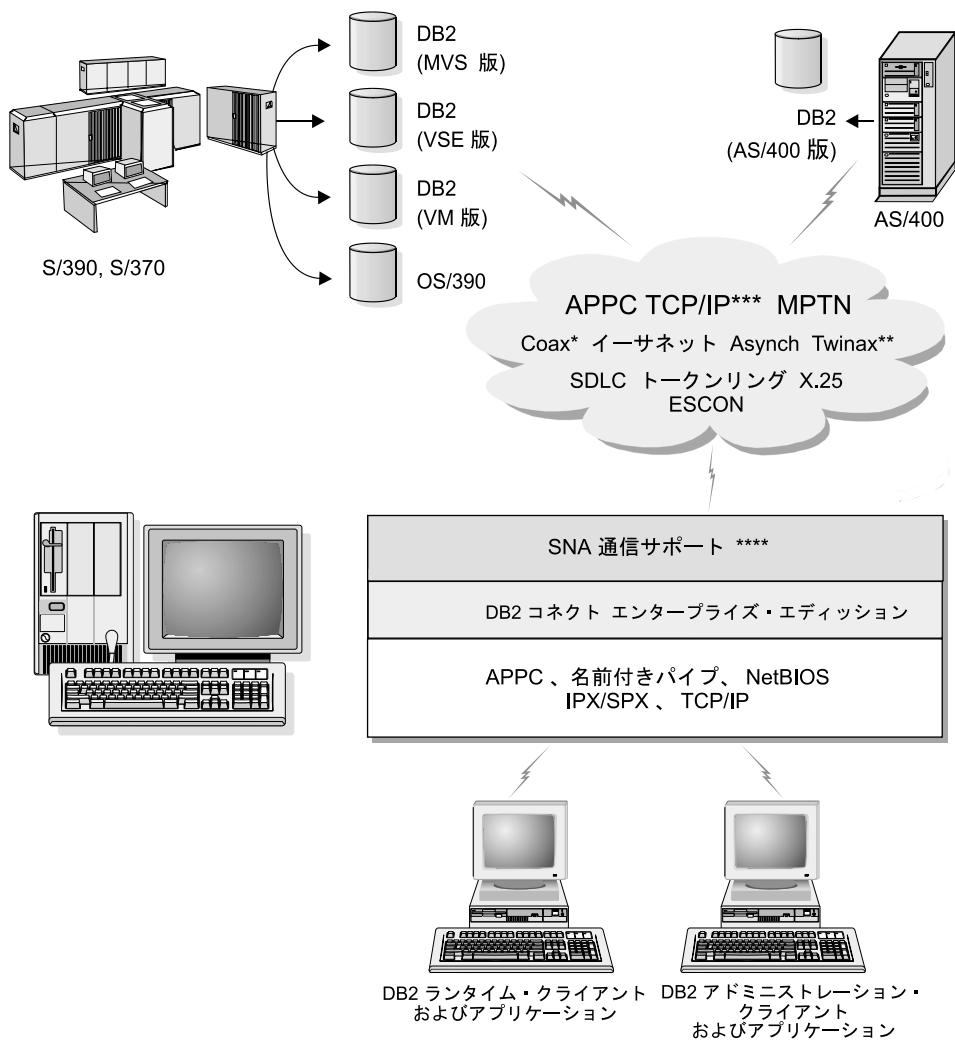
3. DB2 アプリケーション開発クライアントは現在、DB2 コネクトのパッケージに組み込まれており、アプリケーション開発でご利用になりたいときにインストールできます。さらに、DB2 (OS/390 版) のストアード・プロシージャの作成、テスト、配置に使用できるストアード・プロシージャ・ビルダーも DB2 コネクトに組み込まれています。
4. C プログラマーで、Microsoft の ODBC、OLE DB、ActiveX Data Objects (ADO) を使った Windows アプリケーションを開発している方は、*Microsoft Open Database Connectivity Software Development Kit* を利用してください。Java をプログラミング言語としてアプリケーションを開発しているプログラマーであれば、利用する Java 開発環境はどれでも構いません (IBM の VisualAge for Java など)。

---

### 接続サーバーとしての DB2 コネクト エンタープライズ・エディション

DB2 コネクト・サーバーを使用すれば、複数のクライアントがホストや AS/400 のデータに接続できます。また、企業データへのアクセスの確立や保守に要する労力を大幅に減らすことができます。28ページの図4 では、DB2 クライアントをホストまたは AS/400 のデータベース・サーバーへ DB2 コネクト エンタープライズ・エディション経由で間接接続させることのできる環境を実現する IBM のソリューションを示しています。

以下の例では DB2 コネクト・サーバーを、DB2 コネクト・サーバー・サポート (DB2 Connect Server Support) 構成要素がインストールされた、DB2 UDB のエンタープライズ・エディションまたはエンタープライズ拡張エディションに置き換えることができます。



すべてのプロトコルがすべてのプラットフォームでサポートされているわけではありません。

- \* ホスト接続の場合のみ
- \*\* AS/400 の場合
- \*\*\* TCP/IP 接続には、DB2 (OS/390 版) V5R1、DB2 (AS/400 版) V4R2、DB2 (VM 版) V6.1 のいずれかが必要
- \*\*\*\* SNA 通信サポートは各オペレーティング・システムに固有のものであり、TCP/IP 接続が利用できない場合にのみ必要

図4. DB2 コネクト エンタープライズ・エディション

---

## DB2 コネクトと Web アプリケーション

Web ブラウザーは急速な勢いで、オンライン・カタログからイントラネット・アプリケーションに至るあらゆるものの標準インターフェースになりつつあります。単純な Web アプリケーションであれば、Web サーバーだけで十分な場合もあります。それに対し、データベース・アクセスやトランザクション処理を必要とするような、大量データを処理するアプリケーションについては、DB2 コネクトを使って Web 上で膨大な数のトランザクションを同時に処理するソリューションが IBM によって提供されています。

ここでは、DB2 コネクトを使用することで効果を上げることができる Web ベースのビジネス・ソリューションについて解説します。

### 従来の CGI プログラミングの利点と限界

WWW 上の e-business アプリケーションは通常、ユーザーがバックエンドのデータベースを照会できるように CGI (Common Gateway Interface) を使っています。多くの企業も内部で Web アプリケーションを利用しており、この場合でも通常はバックグラウンドでデータベースが稼働しています。

ユーザーが Web ページ上のフォームに記入すると、それらのフォームは CGI 経由で Web サーバー上のアプリケーションまたはスクリプトに送られます。次に、そのスクリプトは提供されているデータベース API を使って、SQL 照会をホスト・データベースへ送信します。続いて、その同じスクリプトは照会の結果を表示する Web (HTML) ページを作成し、ユーザーの Web ブラウザーが表示できるようにそれを送り返します。たとえば、特定の商品やサービスが入手 (利用) 可能であるか、あるいはその現在の価格がいくらかをユーザーが照会できるオンライン・カタログはその一例です。

CGI アプリケーションは設計をシンプルにすることができ、保守するのも容易です。CGI はオペレーティング・システムと言語のいずれにも依存しない規格であるため、ほとんどすべてのコンピューティング・プラットフォームで利用できます。CGI プログラムは C++ で記述することもできれば、Perl などのスクリプト言語で記述することもできます。

このように、CGI は Web ベースのアプリケーションにおける理想的なソリューションのように見えますが、実は重大な欠点があります。まず、CGI のプログラミング環境は他の API の場合ほど洗練されていません。さらにスケーラビリティにも問題があり、大規模な e-commerce を運用するときに影響を及ぼします。CGI アプリケーションを 1 つ起動するたびに、Web サーバーでは新しいプロセスが 1 つ作成されます。各インスタンスはデータベースへの接続を各自のものとして確立する必要があり、それぞれが各自の照会を送信する

こととなります。大量のトランザクションの処理を行う環境では、このような制限によってパフォーマンスに重大な問題が生じることがあります。

DB2 コネクトを Web サーバーと組み合わせて使えば、大量のデータを扱う堅固な e-commerce アプリケーションを作成できます。DB2 コネクトには、Web ベースのアプリケーションのパフォーマンスを改善するいくつかのソリューションが用意されています。さらに、DB2 コネクトのユーザーはストアード・プロシージャ（『Web サーバー上の DB2 コネクト』）を利用することで、データベースに送られる照会の数を減らすことができます。

接続プール（31ページの『接続のプール』を参照）を利用すれば、データベースとの間の接続および切断の頻度を少なくすることができます。大量データを処理するときに CGI の制限が問題となる場合は、IBM の Net.Data（33ページの『Net.Data』を参照）および WebSphere（34ページの『IBM WebSphere』を参照）を使って、大量データを処理するエンタープライズ・アプリケーションへ非 CGI 接続を行うことが可能です。

## Web サーバー上の DB2 コネクト

DB2 コネクトのすべての製品（OS/2 版、UNIX 版、Windows NT 版、Windows 2000 版）には、HTTP（Web）サーバーが用意されています。さらに、DB2 コネクト エンタープライズ・エディションは Apache や Lotus Domino Go といった Web サーバーを独創的な方法でサポートしているだけでなく、Microsoft Internet Information Server や Netscape Enterprise Server など他の Web サーバーと組み合わせて利用することも可能です。

OS/390、AS/400、VM、VSE の各システム上で DB2 ファミリーのデータベースを利用している場合、Web サーバー側では DB2 コネクト エンタープライズ・エディションが必要になります。DB2 コネクト エンタープライズ・エディションには、Web サーバーがこれらのホスト・プラットフォームおよび AS/400 プラットフォームへアクセスすることを可能にするライブラリーと通信インターフェースが用意されています。Web サーバーと、OS/390、AS/400、VM、または VSE 上で稼働しているデータベースとの間の通信には、TCP/IP または SNA のいずれかを使用できます。

**注:** IBM の Web ソリューションでは、複数のデータベースを同一の CGI スクリプトで、あるいは特定の CGI スクリプトの同一トランザクション内で扱うことができます。



続く 2 つのセクションでは、DB2 データベースにアクセスする CGI アプリケーション側で実行可能な、パフォーマンスを向上させる方法について説明します。その後のセクションでは、標準の CGI に代わる手段 (Java など) について考えます。

### ストアード・プロシージャ

クライアント / サーバーの場合と同様、Web アプリケーションについて考慮すべき重要な点として、HTTP サーバーとバックエンド・データベースの間で発生する通信量を最小限に抑えることが挙げられます。これは、大多数の e-business アプリケーションで中心的な役割を果たす、大量データのトランザクション処理において特に重要な点です。

推奨されるアプローチは、CGI アプリケーションのプログラミングを、ストアード・プロシージャにカプセル化されたプログラミングおよびビジネス・ロジックと組み合わせる手法です。OS/2、UNIX、および Windows 上の DB2 ユニバーサル・データベースと、OS/390、AS/400、および VSE 上の DB2 はいずれも、ストアード・プロシージャを呼び出すときに同じパラメーター規則を使います。

通常の CGI の場合と同様、Web ブラウザーはフォームを Web サーバーに送信し、そこで CGI スクリプトが実行されます。ただし、個々の SQL ステートメントが DB2 データベースに送られる代わりに、ストアード・プロシージャを実行するようにとの要求が送られます。このストアード・プロシージャは、何もしなければ別々に実行されてしまういくつかの SQL ステートメントをカプセル化します。それで、ストアード・プロシージャを利用することによって、CGI スクリプトとバックエンド・データベースとの間でやり取りされるメッセージの数を減らすことができます。

ストアード・プロシージャの主な利点は、HTTP サーバーとバックエンドの DB2 データベースの間におけるネットワーク通信量が減少することです。ストアード・プロシージャについて詳しくは、アプリケーション開発の手引き、または DB2 ストアード・プロシージャ・ビルダーのオンライン・ヘルプを参照してください。

### 接続のプール

DB2 コネクト・サーバーからホストへの接続を確立するには、コンピューター・リソースと時間が必要です。数千ものクライアントが DB2 コネクト・サーバーを介してホストに対し接続と切断を頻繁に繰り返す環境では、接続の確立と切断に相当の処理時間を費やすことになります。

DB2 コネクトの接続プールは、そのような環境でのパフォーマンスを大幅に向上させるものです。DB2 コネクトは、データベースに対してオープンしている接続を使用可能なプールで保持し、クライアントが接続を要求すると、すでに使える状態になっている接続をこのプールから提供できます。接続プールによって、通常はそれらの接続をオープンしたりクローズしたりするのに費やされるオーバーヘッドが大幅に減少します。

接続プールの仕組みについて詳しくは、155ページの『接続のプール』を参照してください。

## Java アプリケーション・サーバーとしての DB2 コネクト

CGI の欠点の多くは、それをなるべく使用せず、代わりに Java を使うことで回避できます。IBM は、Web トランザクションのどの場面でも CGI の代わりに Java を利用できるようにするアプレットとアプリケーションの両方を提供しています。IBM が提供するソリューションでは、いくつかの技術を組み合わせるようになって利用できるようになっています。つまり、Net.Data や Microsoft Active Server Pages のようなスクリプトによるソリューションを DB2 と組み合わせたり、Java アプリケーション・サーバー (IBM WebSphere など) で実現されるより堅固なシステムへ移行できるということです。

Java プログラマーが使えるアプリケーション・プログラミング・インターフェース (API) は 2 つあります。その最初である JDBC は、Java を使ってデータ指向の Java アプレットや Java アプリケーション、さらに Java サーブレット、Java Server Pages (JSP)、Enterprise Java Beans (EJB) を開発するためにサポートされています。JDBC はコール・レベルまたはメソッド呼び出しの API です。もう一方の Java API は、SQLJ です。SQLJ では、SQL を Java プログラム内にインラインで指定することが可能です。DB2 はこの両方の API を、Web トランザクションのクライアント側かサーバー側で使用できます。

クライアント側では、アプレット、データ指向のアプレット、アプリケーションなどがサポートされています。データベース側では、Java を使用可能にする部分がデータベース・オブジェクト (ユーザー定義関数やストアド・プロシージャなど) で構成されています。

DB2 (OS/390 版)、DB2 (VSE および VM 版)、そして DB2 (AS/400 版) の場合、Java アプリケーションを配置する方法には 2 通りあります。DB2 コネクト パーソナル・エディションが提供する TCP/IP または SNA による直接接続を使用する方法と、DB2 コネクト エンタープライズ・エディション・サーバーを経由してメインフレームまたはバックエンドの AS/400 へ接続する方法がそれぞれです。

どちらの場合も、Web 上のユーザーはデータベースへアクセスするのに、標準的な Web ブラウザー以外の特別なソフトウェアを必要としません。また、インストールする必要があるのは、DB2 コネクト・サーバーといずれかの業界標準の Web サーバーだけです。Web サーバーと DB2 コネクトが同じ物理マシン上にない場合は、DB2 クライアントを Web サーバー上にインストールしなければなりません。

DB2 (OS/390 版) の場合、その主要な構成要素は中間層サーバーで稼働している DB2 コネクト エンタープライズ・エディションです。この構成要素が、DB2 (OS/390 版、VSE および VM 版、AS/400 版) サーバーへの接続に加えて、JDBC サーバーを使用可能にする部分を提供します。この場合でも、クライアントの Web ブラウザーに特別なソフトウェアは必要ありません。

IBM では、Java のアプリケーションとアプレットを開発するための多岐にわたるツールを用意しています。データベースの接続性に関しては DB2 Developer's Edition が、VisualAge for Java Professional Edition、WebSphere Application Server、Net.Data を含む完全なキットを提供しています。DB2 ユニバーサル・データベースと DB2 コネクトも評価用としてそれに含まれています。さらに、IBM VisualAge for Java Enterprise Edition にも、大規模なエンタープライズ・アプリケーション向けの開発ツールが同梱されています。他社製のツール、たとえば Borland JBuilder や Symantec Visual Cafe なども、IBM のデータベース・ソリューションで利用できます。

## Net.Data

DB2 ユニバーサル・データベースおよび DB2 コネクト・ファミリーの一部である Net.Data は、Web ベースのトランザクション・アプリケーションを作成または保守するときに役立つ、アプリケーション開発ツールのセットです。Net.Data を使用することで、DB2 UDB (OS/2 版、Windows NT 版、Windows 2000 版、UNIX 版、OS/390 版、VM 版、VSE 版、および AS/400 版) に格納されたデータにアクセスし、それを変更することができます。Net.Data で作成したアプリケーションは Web サーバー上に格納され、Web ブラウザーを介して活動化することが可能です。

Net.Data ではマクロ、またはテンプレートを利用して、HTML と SQL の基本しか理解していないユーザーでも非常に洗練された Web アプリケーションを構築できるようにしています。マクロは、Java、JavaScript、HTML タグ、組み込み関数などで構成されるテキスト・ファイルです。これらのマクロは、事前定義されたレイアウト、変数、関数を持つ動的な Web ページを生成するのに使われます。

基本的な Net.Data マクロには、次の 7 つの明確なセクションがあります。

- 「共通 (Common)」セクション。基本的にはプログラマーに役立つ情報が示されている部分です。
- 「定義 (Define)」セクション。変数の定義を指定する部分です。
- 「関数 (Function)」セクション。主要なプログラミング・ロジックを記述する部分です。
- 「レポート (Report)」セクション。Net.Data マクロの出力結果のフォーマット・ロジックを指定する部分です。
- 「HTML」セクション。Web ページで使われている HTML の大部分をここに記述します。
- 「インクルード (Include)」セクション。他のマクロが再利用できるこのマクロの共通部分を、ここで簡単に組み込むことができます。
- 「メッセージ (Message)」セクション。この部分にはエラー処理を記述します。

特に DB2 を対象にした Net.Data の主な特色として、クライアントの配置が不要である点が挙げられます。このような場合、クライアントは単なる Web ブラウザーで構いません。

Net.Data の処理プログラムは、DB2 ユニバーサル・データベースとともに Windows NT、Windows 2000、OS/2、または UNIX ワークステーション上にインストールされ、同時に Web サーバーにもインストールされます。DB2 (OS/390 版、VSE および VM 版、AS/400 版) への接続時には、Net.Data の基本部分のすべてが DB2 コネクト・サーバー上と Web サーバー上に配置されます。

## IBM WebSphere

IBM WebSphere は、従来の CGI プログラミングでの可能な範囲を超えた、より完全な e-business ソリューションを提供します。WebSphere Application Server は CGI のスクリプト機能を実行するだけでなく、Web を介した複雑で高度なサービスを、サーブレット、Active Server Pages、Enterprise JavaBeans の使用により可能にします。WebSphere を使えば以下のことが行えます。

- 業界標準を利用することで、開発のスピードを上げて相互運用性を最大にする
- 他社製ツールのテクノロジーやアプリケーション・フレームワークをプラグインする (組み込む)。
- Web サイトのコンテンツの性能と使用状況を分析する。

- より多くのユーザーに対応しつつスループットを維持するために、サイトの規模を簡単に調整できる。
- いくつもの主要オペレーティング・システム (IBM AIX、HP-UX、Linux、Novell NetWare、IBM OS/2、IBM OS/390、IBM OS/400、Sun Solaris、Microsoft Windows NT、および Windows 2000) にわたって配置を実行する。
- Apache、IBM、Netscape、Microsoft 製のものを含む、既存の Web サーバーを使用する。

WebSphere は単一の製品ではなく、3 つの異なるマーケットを対象にしたファミリー製品となっています。WebSphere によるソリューションの中核を成しているのは、WebSphere Application Server です。

WebSphere Application Server が提供する環境は、3 種類のオブジェクトを対象にしています。最初のもは Java Server Pages (JSP) であり、これは Active Server Pages (ASP) に相当するものです。2 番目は Java サーブレットで構成される構成要素であり、3 番目は Enterprise JavaBeans です。Enterprise JavaBeans は、大規模で強力なエンタープライズ・クラスのアプリケーションを配置する際の標準になりつつあります。

さらに、Data Access JavaBeans は、特に DB2 を対象にした、非常に洗練されたデータベース機能を提供します。DB2 は JDBC および SQLJ を介して直接アクセスすることもできます。COM+ と CORBA の両方もサポートされています。

WebSphere アプリケーションは、Web サーバーや DB2 ユニバーサル・データベースと同じプラットフォーム上に置くことができます。DB2 (OS/390 版、VM および VSE 版、AS/400 版) の場合、WebSphere は DB2 コネクト エンタープライズ・エディションと同じプラットフォーム上に配置されます。

WebSphere ソリューションは、Web Studio や WebSphere Performance Pack と同様にいくつか用意されています。WebSphere の 3 つのバージョンを以下に示します。

### スタンダード版

Web サイトの作成者を対象にしたこのサーバーは、Java サーブレットと JSP テクノロジーの利用を可能にし、静的ページから、カスタマイズされた動的 Web コンテンツに不可欠なソースに至るまで、Web サイトおよびポータルの変換を簡単に、かつ短時間で行うことができます。さらに、グループ間や企業間での情報とデータの共用を容易にする業界最先端の XML サポートと、Web サイトへの投資から最大限の

利益を上げるのに役立つパフォーマンスと利用状況の情報を提供する、組み込みのサイト分析テクノロジーもこれに含まれています。

### アドバンスド版

アプリケーション・プログラマーを対象にしたこの高性能な EJB サーバーは、EJB 構成要素を使用するビジネス・ロジックの配置を可能にします。スケーラビリティ、機密保護、接続性、Java サポートを提供しており、スタンダード版のすべての機能が含まれています。

### エンタープライズ版

エンタープライズ設計者を対象にしたこのサーバーは、組織内の異なるビジネス・システムを統合して堅固な e-business アプリケーションを構築し、リソースを最大限に再利用します。エンタープライズ版には、賞を獲得した IBM TXSeries や Component Broker のテクノロジーが組み込まれています。さらに、アドバンスド版とスタンダード版のすべての機能も含まれています。

---

## DB2 コネクトをアプリケーション・サーバーと組み合わせて使う

クライアント / サーバー型のアプリケーションが登場したとき、アプリケーション設計者は Windows や OS/2 といったプラットフォーム上でアプリケーションにグラフィカル・ユーザー・インターフェースを持たせることにより、ユーザビリティ（使いやすさ）を向上させてトレーニング・コストを削減することが可能になりました。同時に、多様なオペレーティング・システムとハードウェア・プラットフォーム上にある堅固なデータベース・サーバーに、データベース管理機能を柔軟な方法で送ることも可能になりました。

アプリケーション・ロジックがクライアント・ワークステーションに配布されるこのクライアント / サーバー・モデルのことを、通常は 2 層クライアント / サーバーと呼びます。この 2 層モデルでは、アプリケーションはクライアント層に配置され、データベース・サーバーはサーバー層もしくはバックエンド層に実装されます。24ページの『直接的なデータベース・アクセス』に図示されているように、DB2 コネクトは DB2 (OS/390 版)、DB2 (MVS/ESA 版)、DB2 (AS/400 版)、および DB2 (VM および VSE 版) がデータベース・サーバーとなっている 2 層クライアント / サーバー・アプリケーションを完全にサポートします。

クライアント / サーバー・アプリケーションのサイズが拡大するにつれて、2 層クライアント / サーバー・モデルには重大な限界があることが明らかになってきました。大量のビジネス・ロジックを何百もの、時には何千ものクライアント・ワークステーションに配布することは、変更管理の作業を複雑で費用のかかるものにしてしまいました。また、ビジネス・ルールに何か変更を加える

と、アプリケーションのクライアント部分を置き換えなければなりません。アプリケーションのそのような置き換えは多くの場合、ビジネス・ルールが一貫して適用されるようにするために、企業内のすべてのクライアント・ワークステーションで一斉に行うことが必要です。

システムの規模に関して明白になってきた 2 層クライアント / サーバー・モデルの別の欠点は、そのようなアプリケーションが消費するリソースの量です。何百もの、または何千ものファット・クライアント (太ったクライアント。2 層モデルにおけるクライアントはしばしばこのように呼ばれる) を配置することによって、各クライアント・ワークステーションにより高い処理能力と大きなディスク容量が要求されるようになりました。さらに、それぞれのクライアントが専用のデータベース接続を必要とし、そのような接続を保持することに関連したリソースも必要とされるため、データベース・サーバーの要件も非常に高くなっています。ビジネス・ロジックを配布することに対する 2 層クライアント / サーバーの依存性は、ストアード・プロシージャを広範囲にわたって使うことである程度軽減できますが、その他の欠点については、モデルを変更する以外に簡単に解決する方法はありません。

## アプリケーション・サーバーによるソリューション

2 層クライアント / サーバー・アプリケーションにかかるコストとその複雑さが増大するにつれ、大型アプリケーションの大部分は、多層からなるクライアント / サーバーの方向に発展しました。多層モデルでは、データベース層の役割は変わっていませんが、クライアント層には 1 つまたは複数の中間層が追加されます。追加される層は普通は 1 つであるため、このモデルは 3 層 という名前になっています。

3 層モデルでは、クライアントの処理はユーザーとの対話だけにとどまり、ビジネス・ロジックは何も組み込まれません。また、中間層は 1 つまたは複数のアプリケーション・サーバーで構成されます。アプリケーション・サーバーを利用する目的は、ビジネス・プロセスとビジネス・ルールの背後にあるロジックを、堅固かつ費用効率の高い方法で実装することにあります。2 層モデルの場合にそうであったように、ビジネス・ルールの実装には多くの場合、パフォーマンスを向上させるためにストアード・プロシージャが補足的に使われま

す。クライアント・ワークステーションは大量のアプリケーション・ロジックを実装せずに、ユーザーとの対話だけを処理すればよくなるため、クライアント層のリソース要件は大幅に軽減されます。そのような理由から、3 層モデルにおけるクライアント層はよくシン・クライアント (やせたクライアント) と呼ばれます。さらに、中心となるアプリケーション・サーバーがすべてのクライア

ントからの要求を処理するため、そのアプリケーション・サーバーはすべてのクライアント間のデータベース接続などのリソースを共用することが可能です。その結果、データベース・サーバーがアプリケーション・ユーザーごとに専用の接続を保持する必要がなくなります。

現在、業界には 3 層アプリケーション・サーバーの応用例がいくつも存在しています。ほとんどの ERP (エンタープライズ・リソース・プランニング) ベンダーは 3 層モデルを使って自社のアプリケーションを実装しています。これには、SAP R/3 や PeopleSoft V7 といったアプリケーションが含まれます。それ以外の例としては、Siebel や Vantive などの代表的なエンタープライズ・リレーションシップ・マネジメント・ベンダーが含まれています。

## アプリケーション・サーバーと DB2 コネクト

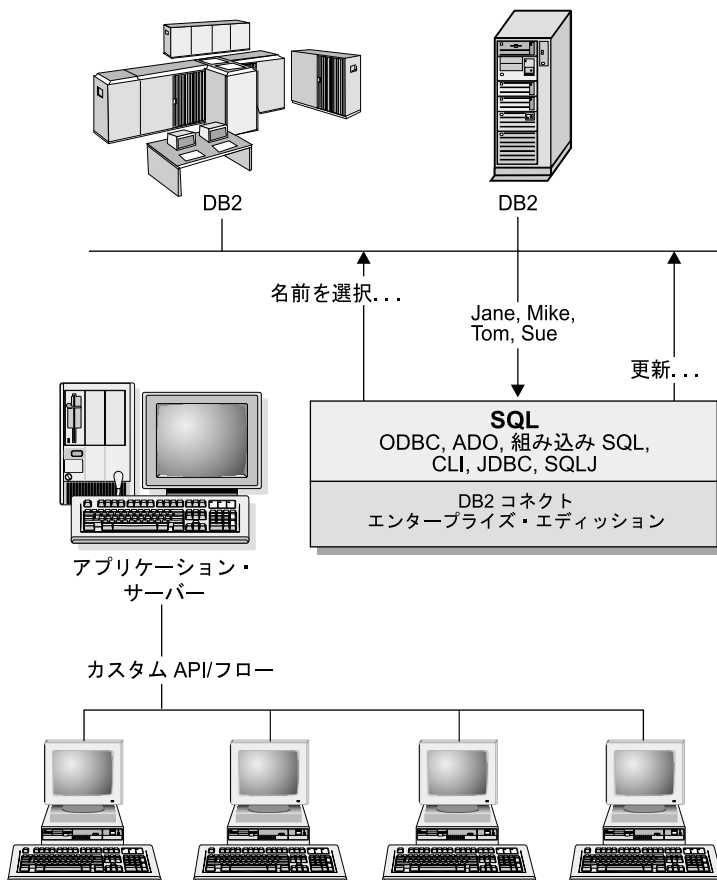
DB2 コネクト エンタープライズ・エディションのサーバーは、多層アプリケーションを総合的にサポートします。DB2 コネクトによるサポートは、アプリケーション・ロジックを開発するのに使用できる各種の API (ODBC、ADO、DB2 CLI、組み込み SQL、JDBC、SQLJ など) だけでなく、DB2 ファミリーのデータベース・サーバーと対話するための完全な通信基盤をも含んでいます。

DB2 コネクトはさらに、データベース層が DB2 ファミリーの複数のデータベース・サーバーで構成されているようなシステムもサポートします。これにより、アプリケーション・サーバーは複数のデータベース・サーバー上に置かれているデータを 1 回のトランザクションで更新するトランザクションを実装できます。

そのような分散トランザクションの整合性は、DB2 コネクトが提供する 2 フェーズ・コミットのプロトコル・サポートによって保証されます。たとえば、アプリケーションは同一トランザクション内で DB2 (OS/390 版) データベースと DB2 UDB (Windows NT 版) のデータを更新することが可能です。また、分散要求サポートがインストールされて有効になっている場合は、アプリケーションは同一トランザクション内で Oracle データベースの読み取りと DB2 ファミリーのデータベースの更新を行うことができます。

下記の図では、DB2 コネクト エンタープライズ・エディションにより、アプリケーション・サーバーとバックエンドのデータベース・サーバーとの間の接続機構および API が提供されています。





接続プール (155ページの『接続のプール』を参照) や接続コンセントレーター (157ページの『DB2 コネクトの接続コンセントレーター』を参照) といった DB2 コネクトの先進機能を利用すれば、アプリケーションのリソース要件は大幅に軽減され、アプリケーション・サーバーの実装も単純化できます。

## DB2 コネクトとアプリケーション・サーバーの構成

アプリケーション・サーバーを使用するには、DB2 コネクト エンタープライズ・エディション (単体の製品として、あるいは DB2 Connect Unlimited Edition 製品のパッケージの一部として入手可能) が必要です。DB2 コネクト パーソナル・エディションはアプリケーション・サーバーをサポートしておらず、それに必要なライセンスも含んでいません。さらに、アプリケーション・サーバーを実装する予定があれば、ご使用の DB2 コネクトの使用許諾条件をよく読み、入手する必要のあるライセンスの数を把握してください。アプリケ

ーション・サーバー環境では、DB2 コネクトを配置する方法が 2 つあります。DB2 コネクト エンタープライズ・エディションは次のいずれかにインストールされます。

- アプリケーション・サーバーのマシン
- 独立している通信サーバーのマシン

ほとんどのケースでは、DB2 コネクトをアプリケーション・サーバーと同じサーバー上にインストールすることをお勧めします。DB2 コネクトをアプリケーション・サーバー上にインストールすることで、アプリケーション・サーバーで実装しようとしているフェールオーバーおよびロード・バランシングの仕組みに DB2 コネクトを関与させることができます。このようにセットアップすると、DB2 コネクトを別のサーバー上にインストールした場合に必要とされる余分のネットワーク・ホップがないため、パフォーマンスの向上を期待できます。さらに、別のサーバーをもう 1 つインストールして保守することも不要であるため、管理作業も単純化されます。

DB2 コネクトを別のサーバーにインストールした方がよいケースもあります。アプリケーション・サーバーが稼働しているオペレーティング・システムやハードウェア・プラットフォームに DB2 コネクト エンタープライズ・エディションが対応していない場合がそうです。たとえば、アプリケーション・サーバーが Silicon Graphics (SGI) または SCO UnixWare のサーバー上に配置されている場合は、DB2 コネクト エンタープライズ・エディションがそれらのプラットフォームに対応していないため、DB2 コネクトを別のサーバーに配置する以外に方法はありません。

---

## DB2 コネクトをトランザクション処理モニターと組み合わせて使う

前の部分では、DB2 コネクトをアプリケーション・サーバーと組み合わせて使う方法について説明しました。アプリケーション・サーバーを利用すれば、多数のユーザーが最小限のシステム・リソースでアプリケーションを実行できます。

アプリケーション・サーバーを拡張して、調整されたトランザクションを、そのアプリケーション・サーバーが実行するアプリケーションから呼び出せるようにすることができます。このトランザクション調整機能は一般に、トランザクション処理 (TP) モニターとして知られています。TP モニターはアプリケーション・サーバーと連携して機能します。

トランザクションは、組織の日常業務を処理するときに生じる定型的なイベント（通常はサービス要求）とみなすことができます。トランザクションの規則正しい処理が、TP モニターで想定されている作業のタイプとなっています。

どの組織にも、どのようにそれが運営されるかを表現したルール（法則）と手順が存在します。これらのルールを具体化するユーザー・アプリケーションのことを、ビジネス・ロジックと呼ぶことができます。また、これらのビジネス・アプリケーションが実行するトランザクションのことを、しばしばトランザクション処理、あるいはオンライン・トランザクション処理（OLTP）と呼びます。

商用 OLTP の主要な特性は以下のとおりです。

#### **多数のユーザー**

非常に多くの人間がビジネスの現在の状態に影響するため、トランザクション処理は組織の人間の大多数によって利用されるのが普通です。

**反復性** コンピューターとの対話のほとんどは、何度も繰り返し実行される同一のプロセスであることが少なくありません。たとえば、注文の入力や支払いの処理などは毎日、何回も使われます。

#### **短時間の対話**

組織内の人間がトランザクション処理システムと行う対話のほとんどは短時間で済みます。

#### **共用データ**

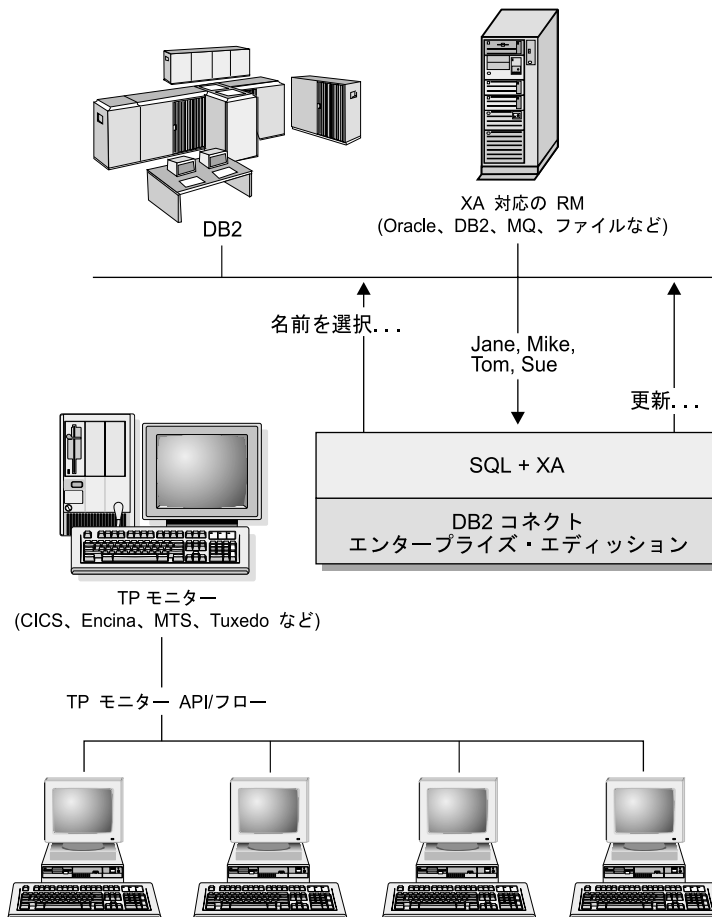
データは組織の状態を表すものであるため、データには 1 つのコピーしかありません。

#### **データ保全性**

データは組織の現在の状態を表していなければならない、内部的な矛盾があってはなりません。たとえば、どの注文も顧客レコードに関連付けられている必要があります。

#### **低コスト / トランザクション**

トランザクション処理はビジネスを行うときの直接コストを表しているため、システムのコストは最小でなければなりません。DB2 コネクトを使用すれば、UNIX、Windows NT、Windows 2000、または OS/2 上で稼働しているアプリケーション・サーバーの制御下にあるアプリケーションが、リモート LAN、ホスト、AS/400 のデータベース・サーバーに対してトランザクションを実行し、それらのトランザクションを TP モニターに調整させることが可能になります。



この図では、DB2 コネクト エンタープライズ・エディションが API とともに、アプリケーション・サーバーとバックエンドのデータベース・サーバーとの間の接続機構を提供しています。

## TP モニターの例

現在、市販されている代表的な TP モニターには以下のものがあります。

- IBM TxSeries CICS
- IBM TxSeries Encina Monitor
- BEA Tuxedo

Microsoft Transaction Server Remote S/390、AS/400、LAN などのデータベース・サーバーは、これらの TP モニターで調整されたトランザクション内で使用できます。

## Tuxedo と DB2 コネクト

DB2 コネクト バージョン 6 以前では、Tuxedo ベースのアプリケーションによる、ホストおよび AS/400 のデータベース・サーバーに対するアクセスが、読み取り専用アクセスに限定されていました。DB2 コネクト バージョン 7 ではこの制限がなくなっており、Tuxedo ベースのアプリケーションは、Tuxedo の調整トランザクション内にあるホストおよび AS/400 のデータベース・サーバーを更新できるようになりました。構成の面では、特別な要件と制約事項が適用されます。詳細については、157ページの『DB2 コネクトの接続コンセントレーター』を参照してください。

## X/Open 分散トランザクション処理 (DTP) モデル

単一のトランザクションで複数のリソースを更新するのに、ビジネス・ロジックを実行するアプリケーションが必要になることがあります。たとえば、ある口座から別の口座への送金を実現する銀行業務アプリケーションは、一方のデータベース (送金元口座) からの引き落とし処理と、もう一方のデータベース (送金先口座) への入金処理を必要とするかもしれません。

これら 2 つのデータベースが別々のベンダーのものである可能性もあります。たとえば、一方のデータベースが DB2 ユニバーサル・データベース (OS/390 版) で、もう一方が Oracle データベースになっている場合があります。このような場合、それぞれのデータベース・ベンダー独自のトランザクション・インターフェースを TP モニターごとに実装するのではなく、TP モニターと、アプリケーションがアクセスするリソースとの間に共通のトランザクション・インターフェースが定義されています。このインターフェースは XA インターフェースとして知られているものです。XA インターフェースを使用する TP モニターのことを XA 準拠トランザクション・マネージャー (TM) と呼びます。また、XA インターフェースを実装する更新可能なリソースのことを XA 準拠リソース・マネージャー (RM) と呼びます。

上記の TP モニターはすべて XA 準拠 TM です。リモート・ホスト、AS/400、および DB2 UDB の LAN ベースのデータベース・サーバーは、DB2 コネクト経由でアクセスを行うときは XA 準拠 RM になります。そのため、XA 準拠 TM を有する TP モニターであれば、トランザクションを実行するビジネス・アプリケーション内にある、ホスト、AS/400、および LAN ベースの DB2 UDB のデータベース・サーバーを使用できます。

## DB2 コネクトで XA 準拠トランザクション・マネージャーを使用する方法

ここでは、TP モニターで S/390 と AS/400 のデータベース・サーバーを使用するのに必要な構成手順について説明します。これは、すでに操作可能な TP モニターがあって、DB2 コネクトがインストールされていることを前提にして

います。また、ホストまたは AS/400 のデータベース・サーバーへの接続の構成とテストも済んでいなければなりません。詳細については、DB2 コネクト概説およびインストール を参照してください。

よく使われている TP モニターを構成するのに必要な手順は、管理の手引きに記載されています。LAN ベースの DB2 UDB データベース・サーバーへのアクセスと、ホストまたは AS/400 のデータベース・サーバーへのアクセスとでは、構成作業に違いはありません。以下の手順は、管理の手引き に明記されていない TP モニターの一般的な構成手順を示したものです。

DB2 コネクトが TP モニター内にある S/390 と AS/400 のデータベース・サーバーを使用するように構成するには、以下の手順に従ってください。

1. TP モニターが DB2 XA スイッチにアクセスできるように構成します。  
DB2 XA スイッチは、TP モニターに DB2 コネクトの XA API のアドレスを提供します。これを行う方法は TP モニターごとに異なっています。DB2 XA スイッチを TP モニターに提供する方法については、管理の手引き を参照してください。
2. TP モニターを DB2 の XA\_OPEN ストリングを使って構成します。これを行う方法は TP モニターごとに異なっています。DB2 コネクトの XA\_OPEN ストリングについては、管理の手引き を参照してください。また、DB2 の XA\_OPEN ストリングを TP モニターが使用するように構成する方法については、ご使用の TP モニターの資料を参照してください。
3. 必要に応じて、DB2 コネクト同期点管理プログラム (SPM) の省略時の構成パラメーターを変更します。ホストと AS/400 のデータベース・サーバーは現在のところ、XA インターフェースをサポートしていません。

SPM は DB2 コネクトの構成要素の 1 つで、XA の 2 フェーズ・コミット・プロトコルを、ホストと AS/400 のデータベース・サーバーが使用する 2 フェーズ・コミット・プロトコルにマップします。省略時の状態では、DB2 インスタンスに SPM 構成パラメーターの事前定義値が指定されています。最も重要なパラメーターは、データベース・マネージャーの構成パラメーター SPM\_NAME です。TCP/IP ホスト名の最初の 7 文字からとった名前が省略時値になっています。

TCP/IP を使って DB2 (OS/390 版) への接続を行うときは、省略時の設定値はどれも変更する必要はありません。この場合、SPM はすでに動作可能になっているため、SPM の構成作業は不要です。ホストまたは AS/400 のデータベース・サーバーへのアクセスに SNA を使用するときは、SPM\_NAME 値がネットワークの有効な SNA LU を表すようにしなければ

なりません。デフォルトの SPM\_NAME 値をそのまま使うことができない場合は、「複数サイト更新 (Multisite Update)」ウィザードでこの値を変更してください。





---

## 第4章 DB2 コネクト環境でのプログラミング

このセクションでは、DB2 コネクトを使用するアプリケーションの作成に関する情報を提供します。詳細については、コール・レベル・インターフェースの手引きおよび解説書、コマンド解説書、およびアプリケーション開発の手引きを参照してください。

---

### 分散環境でのプログラミング

DB2 コネクトにより、アプリケーション・プログラムは、System/390 サーバーと AS/400 サーバー上にある DB2 データベース内のデータにアクセスできます。たとえば、Windows の下で実行されているアプリケーションは DB2 ユニバーサル・データベース (OS/390 版) データベース内のデータにアクセスできます。新規のアプリケーションを作成することもできますし、既存のアプリケーションをホストまたは AS/400 環境で実行できるように修正することもできます。ある 1 つの環境でアプリケーションを開発し、それを別の環境に移植することもできます。

DB2 コネクトを使用すれば、DB2 ユニバーサル・データベース (OS/390 版) のようなホスト・データベース製品で以下の API が利用できるようになります。ただし使用できるのは、その項目がホスト・データベース製品によりサポートされている場合に限られます。

- 組み込み SQL (静的および動的の両方)
- DB2 コール・レベル・インターフェース
- Microsoft ODBC API
- JDBC

SQL ステートメントの中には、関係データベース製品によって相違のあるものもあります。使われる SQL ステートメントは以下のように分類できます。

- 標準に関係なく、使用するすべてのデータベース製品で同一のもの
- SQL 解説書に掲載されており、それゆえにすべての IBM 関係データベース製品で使用できるもの
- アクセスする 1 つのデータベース・システムで固有のもの

はじめの 2 つの類別の SQL ステートメントは可搬性が高いですが、3 番目の類別の SQL ステートメントはまず変更を行うことが必要な場合があります

す。一般的に、データ定義言語 (DDL) での SQL ステートメントは、データ操作言語 (DML) での SQL ステートメントほど可搬性が高くありません。

DB2 ユニバーサル・データベースではサポートされていない SQL ステートメントでも DB2 コネクトには受け入れられるものがあります。DB2 コネクトはこれらのステートメントをホストまたは AS/400 サーバーに渡します。最大列長のような別のプラットフォームの制限に関する情報は、*SQL 解説書* を参照してください。

CICS アプリケーションを OS/390 または VSE から他の CICS 製品 (たとえば CICS (AIX 版) ) の下で実行するよう移植する場合には、DB2 コネクトを使用して OS/390 または VSE のデータベースにアクセスすることもできます。詳細については、*CICS/6000 適用業務プログラミングの手引き* および *CICS カスタマイゼーションおよび操作* を参照してください。

ホストまたは AS/400 環境でプログラミングを行う場合は、以下に挙げる要因に特に注意してください。

- データ定義言語 (DDL) の使用
- データ操作言語 (DML) の使用
- データ制御言語 (DCL) の使用
- 接続および切断
- プリコンパイル
- 分類順序の定義
- 参照保全の管理
- ロッキング
- SQLCODE および SQLSTATE に関する相違点
- システム・カタログの使用
- 分離レベル
- ストアード・プロシージャ
- NOT ATOMIC 複合 SQL
- 分散作業単位
- DB2 コネクトでサポートされる SQL ステートメントと拒否される SQL ステートメント

## データ定義言語 (DDL) の使用

異なるシステムでは記憶域の扱われ方が異なるので、DDL ステートメントは IBM データベース製品により異なります。ホストまたは AS/400 サーバー・システムでは、データベースの設計と CREATE TABLE ステートメントの発行の間にいくつかのステップがあり得ます。たとえば、一連のステートメントによって、論理オブジェクトの設計を、それらのオブジェクトの記憶域内での物理的表現へと変換する場合があります。

ホストまたは AS/400 サーバー・データベースへのプリコンパイルを行うさいに、プリコンパイラはそのような多くの DDL ステートメントをホストまたは AS/400 サーバーに渡します。その同じステートメントは、アプリケーションが実行されているシステム上のデータベースに対してはプリコンパイルを行いません。たとえば、OS/2 アプリケーションにおいて、CREATE STORGROUP ステートメントは DB2 ユニバーサル・データベース (OS/390 版) データベースに対しては正常にプリコンパイルされますが、DB2 (OS/2 版) データベースに対しては正常にプリコンパイルされません。

## データ操作言語 (DML) の使用

一般的に、DML ステートメントには高い可搬性があります。SELECT、INSERT、UPDATE、および DELETE ステートメントなどは、IBM 関係データベース製品の間で類似しています。ほとんどのアプリケーションは主に DML SQL ステートメントを使用しています。それらは DB2 コネクト・プログラムによりサポートされています。

### 数値データ・タイプ

数値データが DB2 ユニバーサル・データベース に転送されると、データ・タイプが変わることがあります。(DB2 ユニバーサル・データベース (AS/400 版) によってサポートされる) 数値およびゾーン 10 進の SQLTYPE は、固定 (パック) 10 進の SQLTYPE に変換されます。

### 混合バイト・データ

混合バイト・データは、拡張 UNIX 文字セット (EUC) の文字、2 バイト文字セット (DBCS) の文字および 1 バイト文字セット (SBCS) の文字が同じ列の中で構成されています。データを EBCDIC で保管するシステム (OS/390、OS/400、VSE および VM) では、2 バイト・データの始まりと終わりはシフトアウト (SO) 文字およびシフトイン (SI) 文字で印付けられます。データを ASCII で保管するシステム (OS/2 および UNIX など) では、シフトイン文字およびシフトアウト文字は必須ではありません。

アプリケーションで混合データを ASCII システムから EBCDIC システムへ転送する場合は、シフト文字のための十分な余地を確保するようにしてください。SBCS データから DBCS データへ切り換わるたびに、データの長さにそれぞれ 2 バイトを追加してください。さらに可搬性を良くするためには、混合データを使用するアプリケーションには可変長ストリングを使用してください。

### 長フィールド

長フィールド (長さが 254 文字を超えるストリング) は、システムによって扱い方が異なります。ホストまたは AS/400 サーバーは、長フィールド用のスカラー関数のサブセットだけをサポートしています。たとえば、DB2 ユニバーサル・データベース (OS/390 版) では、長フィールドに合わせて **LENGTH** および **SUBSTR** 関数だけが許されます。また、ホストまたは AS/400 サーバーでは、ある種の SQL ステートメントには別の取り扱いが必要になります。たとえば DB2 (VSE および VM 版) では、INSERT ステートメントを用いる場合、ホスト変数、SQLDA、またはヌル値だけを使用することが必要です。

### ラージ・オブジェクト (LOB) データ・タイプ

LOB データ・タイプは DB2 コネクトでサポートされています。

### ユーザー定義タイプ (UDT)

DB2 コネクトではユーザー定義の具象タイプだけがサポートされています。抽象データ・タイプはサポートされていません。

### ROWID データ・タイプ

ROWID データ・タイプは、DB2 コネクトではビット・データ用の VARCHAR として処理されます。

### 64 ビット整数 (BIGINT) データ・タイプ

DB2 コネクトでは 8 バイト (64 ビット) 整数がサポートされています。BIGINT 整数データ・タイプは、データ精度を保ちながら大規模データベースのカーディナリティーをサポートするときに使用します。

## データ制御言語 (DCL) の使用

各 IBM 関係データベース管理システムでは、GRANT および REVOKE SQL ステートメントにいろいろなレベルの区分が用意されています。各データベース管理システムで使用するのに適した SQL ステートメントを確認するために、製品特定の出版物を調べてください。

## 接続および切断

DB2 は、パラメーターを指定していない `CONNECT` を、`CONNECT` ステートメントの `CONNECT TO` および `CONNECT RESET` バージョンと同様にサポートしています。アプリケーションが `CONNECT TO` ステートメントを先に明示的に実行せずに `SQL` ステートメントを呼び出した場合、省略時アプリケーション・サーバー (それが存在する場合) への暗黙接続が実行されます。

データベースに接続するときには、関係データベース管理システムを識別するための情報が `SQLCA` の `SQLERRP` フィールドに戻されます。アプリケーション・サーバーが `IBM` 関係データベースの場合、`SQLERRP` の先頭の 3 バイトは以下のいずれかを含みます。

**DSN** DB2 ユニバーサル・データベース (OS/390 版)

**ARI** DB2 (VSE および VM 版)

**QSQ** DB2 ユニバーサル・データベース (AS/400 版)

**SQL** DB2 ユニバーサル・データベース

DB2 コネクトを使用しているときに `CONNECT TO` または `null CONNECT` ステートメントを発行した場合、`SQLCA` の `SQLERRMC` フィールド内には国別コードまたは領域トークンがブランクとして戻されます。アプリケーション・サーバーの `CCSID` はコード・ページまたはコード・セット・トークンに戻されます。

明示的に切断を行うには、`CONNECT RESET` ステートメント (タイプ 1 接続の場合)、`RELEASE` および `COMMIT` ステートメント (タイプ 2 接続の場合)、または `DISCONNECT` ステートメント (どちらの接続タイプでも可、ただし TP モニター環境では不可) を使用します。

接続が明示的に切断されずにアプリケーションが正常終了した場合、DB2 コネクトは結果のデータを暗黙のうちにコミットします。

**注:** アプリケーションが正常終了したのに、エラーを示す `SQLCODE` を依然として受け取る場合があります。この場合、DB2 コネクトはデータをコミットします。データのコミットを行いたくない場合は、`ROLLBACK` コマンドを発行してください。

`FORCE` コマンドは、選択したユーザーまたはすべてのユーザーをデータベースから切断します。これはホストまたは AS/400 サーバー・データベースにもサポートされるので、ユーザーは DB2 コネクト・ワークステーションの強制切断を行うことができます。

## プリコンパイル

各種の IBM 関係データベース・システムの間で、プリコンパイルに関していくつかの相違点があります。DB2 ユニバーサル・データベース用のプリコンパイラーは、以下の点でホストまたは AS/400 サーバーのプリコンパイラーと異なります。

- アプリケーションをワン・パスするだけでプリコンパイルします。
- DB2 ユニバーサル・データベースのデータベースに対してバインドを行うときは、バインドが成功するためにはオブジェクトが存在していなければなりません。VALIDATE RUN はサポートされていません。

## ブロック化

DB2 コネクトは、DB2 データベース・マネージャーのブロック化バインド・オプションをサポートします。

### UNAMBIG

確定カーソルだけがブロック化されます (省略時値)。

**ALL** 未確定カーソルがブロック化されます。

**NO** カーソルはブロック化されません。

DB2 コネクト・プログラムは、DB2 データベース・マネージャー構成ファイルで定義されたブロック・サイズを RQRIOBLK フィールドに使用します。

DB2 コネクトの現行バージョンでは、ブロック・サイズを最大 32 767 までサポートします。それより大きな値が DB2 データベース・マネージャー構成ファイルで指定される場合、DB2 コネクトは 32 767 値を使用しますが、DB2 データベース・マネージャーをリセットしません。ブロック化は、動的 SQL と静的 SQL で同じブロック・サイズが使用され、同じ方法で処理されます。

**注:** ほとんどのホストまたは AS/400 サーバー・システムは動的カーソルを未確定とみなしますが、DB2 ユニバーサル・データベース・システムは一部の動的カーソルを確定カーソルとみなします。混乱を避けるため、DB2 コネクトでは BLOCKING ALL を指定することができます。

管理 API 解説書 およびコマンド解説書 でリストしているように、CLP、コントロール・センター、または API を使用して DB2 データベース・マネージャー構成ファイルにおいてブロック・サイズを指定してください。

## パッケージ属性

パッケージには以下の属性があります。

### 集合 ID

パッケージの ID。PREP コマンドに指定することができます。

**所有者** パッケージ所有者の許可 ID。PREP コマンドまたは BIND コマンドに指定することができます。

**作成者** パッケージをバインドするユーザー名。

**修飾子** パッケージ内のオブジェクトの暗黙修飾子。PREP コマンドまたは BIND コマンドに指定することができます。

それぞれのホストまたは AS/400 サーバー・システムで、これらの属性の使用に関する制限があります。

#### **DB2 ユニバーサル・データベース (OS/390 版)**

4 つの属性すべてが別々でもかまいません。異なる修飾子を使用するには、特別な管理特権が必要になります。これらの属性の使用条件について詳しくは、DB2 ユニバーサル・データベース (OS/390 版) のコマンド解説書を参照してください。

#### **DB2 (VSE および VM 版)**

すべての属性は同一でなければなりません。USER1 がバインド・ファイルを作成し (PREP 使用)、USER2 が実際のバインドを実行する場合、USER2 は、USER1 に代わってバインドを実行するために DBA 権限が必要になります。USER1 のユーザー名だけが属性に使用されます。

#### **DB2 ユニバーサル・データベース (AS/400 版)**

修飾子は集合名を示します。修飾子と所有権との関連は、オブジェクトにおける特権の授与や取り消しに影響を与えます。ユーザー名が集合 ID によって修飾されない限り、ログオンされるユーザー名が作成者および所有者になります。ユーザー名が集合 ID によって修飾されているときは、集合 ID が所有者になります。集合 ID は、修飾子として使用される前から存在していなければなりません。

#### **DB2 ユニバーサル・データベース**

4 つの属性すべてが別々でもかまいません。別の所有者を使用するには、管理権限が必要で、スキーマ (既存の場合) に対する CREATEIN 特権がバインダーになければなりません。

**注:** DB2 コネクトは、DB2 ユニバーサル・データベース (OS/390 版) および DB2 ユニバーサル・データベースで *SET CURRENT PACKAGESET* コマンドをサポートしています。

#### **C のヌル文字で終了するストリング**

CNULREQD バインド・オプションは、LANGLEVEL オプションを使用して指定したヌル文字で終了するストリングの処理をオーバーライドします。

LANGLEVEL オプションを MIA または SAA1 に設定した場合にヌル文字で終了するストリングがどのように処理されるかについては、アプリケーション開発の手引きを参照してください。

省略時には、CNULREQD は YES に設定されます。こうすることによって、ヌル文字で終了するストリングは MIA 標準に従って解釈されます。DB2 ユニバーサル・データベース (OS/390 版) サーバーに接続している場合には、CNULREQD を YES に設定するよう強くお勧めします。CNULREQD オプションを NO に設定して、(ヌル文字で終了するストリングに関して) SAA1 標準にコード化されたアプリケーションをバインドする必要があります。そうしなければ、ヌル文字で終了するストリングを SAA1 に設定した LANGLEVEL を使用して作成しても、MIA 標準で解釈される場合があります。

### スタンドアロン SQLCODE および SQLSTATE

ISO/ANS SQL92 で定義されているスタンドアロン SQLCODE および SQLSTATE 変数は、LANGLEVEL SQL92E プリコンパイル・オプションを介してサポートされます。SQL0020W 警告がプリコンパイル時刻に発行される場合は、LANGLEVEL がサポートされていないことを示しています。この警告は LANGLEVEL SQL92E のサブセットである コマンド解説書で LANGLEVEL MIA の下にリストされている機能にのみ適用されます。

## 分類順序の定義

EBCDIC と ASCII の差は、種々のデータベース製品間の分類順序の差を引き起こし、ORDER BY と GROUP BY 句にも影響を及ぼします。それらの差を最小限にとどめる方法の 1 つは、EBCDIC 分類順序に模倣するユーザー定義の照合順序を作成することです。照合順序の指定は、新規データベースを作成するときだけ行うことができます。詳細については、アプリケーション開発の手引き、管理 API 解説書、およびコマンド解説書を参照してください。

注: データベース表は ASCII 形式の DB2 ユニバーサル・データベース (OS/390 版) に保管できるようになりました。このことは DB2 コネクトと DB2 ユニバーサル・データベース (OS/390 版) 間のより速いデータの交換を可能にします。また、これによりデータを変換したり並べかえるのに使用するフィールド手順は必要なくなります。

## 参照保全の管理

システムが異なると参照制約の扱い方も異なります。



## DB2 ユニバーサル・データベース (OS/390 版)

外部キーは基本キーを使用して作成しますので、基本キーに関する索引を先に作成しておく必要があります。表はそれ自身を参照することができます。

## DB2 (VSE および VM 版)

外部キー用の索引は自動的に作成されます。表はそれ自身を参照することができません。

## DB2 ユニバーサル・データベース (AS/400 版)

外部キー用の索引は自動的に作成されます。表はそれ自身を参照することができます。

## DB2 ユニバーサル・データベース

DB2 ユニバーサル・データベースのデータベースの場合、固有限制(基本キーを含む)には 1 つの索引が自動的に作成されます。表はそれ自身を参照することができます。

他の規則については、関係する連鎖のレベルにより変化します。

## ロック

データベース・サーバーのロック方法は、アプリケーションにまで影響することがあります。たとえば、行レベルのロックで設計されたアプリケーション、およびカーソル固定の分離レベルは、ページ・レベルのロックを実行するシステムに直接移送することはできません。これらの潜在的な差のゆえに、アプリケーションを調整する必要がある場合があります。

DB2 ユニバーサル・データベース (OS/390 版) および DB2 ユニバーサル・データベース製品には、ロックをタイムアウトにして待機中のアプリケーションにエラー戻りコードを送る機能があります。

## SQLCODE および SQLSTATE に関する相違点

異なる IBM 関係データベース製品は、類似のエラーについて必ずしも同じ SQLCODE を発行するものではありません。この問題には、2 とおりの対処の仕方があります。

- ある特定のエラーの場合に、SQLSTATE を SQLCODE の代わりに使用する。  
SQLSTATE は、どのデータベース製品でもほぼ同一の意味を持ち、それらの製品は SQLCODE に対応する SQLSTATE を作成します。
- あるシステムから別のシステムへの SQLCODE のマップを使用する。

省略時には、DB2 コネクトは SQLCODE およびトークンを各 IBM ホストまたは AS/400 サーバー・システムからユーザーの DB2 ユニバーサル・データベースのシステムへマッピングします。省略時のマッピングを指定変更したい場合や、SQLCODE マッピングのないデータベース・サーバー (非 IBM データベース・サーバー) を使用している場合には、独自の SQLCODE マッピング・ファイルを指定することができます。SQLCODE マッピングをオフにすることもできます。

詳細については、137ページの『第11章 SQLCODE マッピング』を参照してください。

## システム・カタログの使用

システム・カタログは、IBM データベース製品により異なります。視点を使用することにより、多くの相違点をマスクすることができます。詳細については、使用しているデータベース・サーバーの資料を参照してください。

CLI にあるカタログ機能は、同じ API にサポートされることにより、この問題を克服し、DB2 ファミリー全体のカタログ照会を設定していきます。

## 検索割り当てにおける数値変換オーバーフロー

検索割り当てでのオーバーフローは、IBM 関係データベース製品によって異なった処理がなされる場合があります。たとえば、DB2 ユニバーサル・データベース (OS/390 版) および DB2 ユニバーサル・データベースから整数のホスト変数の中に浮動列を取り入れることを考慮してください。浮動値を整数値に変換する際に、変換オーバーフロー (桁あふれ) が生じる場合があります。省略時解釈では DB2 ユニバーサル・データベース (OS/390 版) は、アプリケーションに警告 SQLCODE およびヌル値を戻します。一方、DB2 ユニバーサル・データベースは変換オーバーフロー・エラーを戻します。適正なサイズのホスト変数を取り出すことにより、検索割り当てにおける数値変換オーバーフローを回避するようお勧めします。

## 分離レベル

アプリケーションを prep またはバインドするとき、DB2 コネクトは以下の分離レベルを受け入れます。

- RR** 反復可能読み取り
- RS** 読み取り固定
- CS** カーソル固定
- UR** 非コミット読み取り

## NC コミットなし

分離レベルは保護の大きい順にリストしてあります。指定した分離レベルをホストまたは AS/400 サーバーがサポートしていない場合、その次に高いサポートされているレベルが使用されます。

表2 は、各ホストまたは AS/400 アプリケーション・サーバーにおけるそれぞれの分離レベルの結果を示しています。

表2. 分離レベル

DB2 コネクト	DB2 ユニバーサル・データベース (OS/390 版)	DB2 (VSE および VM 版)	DB2 ユニバーサル・データベース (AS/400 版)	DB2 ユニバーサル・データベース
RR	RR	RR	注 1	RR
RS	注 2	RR	COMMIT(*ALL)	RS
CS	CS	CS	COMMIT(*CS)	CS
UR	注 3	CS	COMMIT(*CHG)	UR
NC	注 4	注 5	COMMIT(*NONE)	UR

### 注:

1. DB2 ユニバーサル・データベース (AS/400 版) 用には RR に一致する同等の COMMIT オプションはありません。DB2 ユニバーサル・データベース (AS/400 版) 用は表全体をロックすることにより RR をサポートします。
2. バージョン 3.1 では RR、APAR PN75407 付きバージョン 4.1 およびバージョン 5.1 では RS になります。
3. バージョン 3.1 では CS、バージョン 4.1 または 5.1 では UR になります。
4. バージョン 3.1 では CS、APAR PN60988 付きバージョン 4.1 およびバージョン 5.1 では UR になります。
5. DB2 (VSE および VM 版) では分離レベル NC はサポートされていません。

DB2 ユニバーサル・データベース (AS/400 版) では、分離レベルが UR でブロック化が ALL にセットされていてアプリケーションがバインドされているならば、もしくは分離レベルが NC にセットされているならば、ジャーナルされていない表にアクセスすることができます。

## ストアド・プロシージャ

### • 呼び出し

クライアント・プログラムは SQL CALL ステートメントを発行することにより、サーバー・プログラムを呼び出すことができます。この場合、各サーバーの機能の仕方は他のサーバーと若干異なります。

## OS/390

プロシージャー名の長さは 8 バイトを超えてはならず、サーバー上の SYSIBM.SYSPROCEDURES カタログに定義されていなければなりません。

## VSE または VM

プロシージャー名の長さは 18 バイトを超えてはならず、サーバー上の SYSTEM.SYSROUTINES カタログに定義されていなければなりません。

## OS/400

プロシージャー名は SQL 識別子でなければなりません。ストアード・プロシージャーを見つけるときの実際のパス名 (スキーマ名または集合名) を指定するために、`DECLARE PROCEDURE` または `CREATE PROCEDURE` ステートメントを使用することもできます。

REXX/SQL から DB2 (AS/400 版) に対する `CALL` ステートメントはすべて、`CALL USING DESCRIPTOR` 形式の REXX/SQL マップで設定された `CALL` ステートメントとしてアプリケーションにより動的に作成され、実行される必要があります。

SQL `CALL` ステートメントの構文については、*SQL 解説書* を参照してください。アプリケーション・プログラムの作成の際のストアード・プロシージャーの使用法に関しては、*アプリケーション開発の手引き* を参照してください。

DB2 ユニバーサル・データベース上のサーバー・プログラムは、DB2 ユニバーサル・データベース (OS/390 版)、DB2 ユニバーサル・データベース (AS/400 版)、または DB2 (VSE および VM 版) で使用されるサーバー・プログラムと同じパラメーター規則を指定して起動できます。DB2 ユニバーサル・データベースのストアード・プロシージャーを起動する方法について詳しくは、*アプリケーション開発の手引き* を参照してください。特定のプラットフォーム用のパラメーター規則に関する詳細は、そのプラットフォームに関する DB2 製品資料を参照してください。

ストアード・プロシージャー内のすべての SQL ステートメントは、クライアント SQL プログラムによって開始される SQL 作業単位の一部として実行されます。

- 特殊な意味のある標識値を、ストアード・プロシージャーに対して、またはストアード・プロシージャーから渡さないでください。

DB2 ユニバーサル・データベース相互間では、ユーザーが標識変数に何を入力しても、システムはすべて渡します。ただし、DB2 コネクトを使用する場合、標識変数のうち 0、-1、および -128 だけを渡すことができます。

- パラメーターを定義して、サーバー・アプリケーションによって生じるエラーや警告をすべて返す必要があります。

DB2 ユニバーサル・データベース上のサーバー・プログラムは、SQLCA を更新してエラーや警告をすべて返すことができますが、DB2 ユニバーサル・データベース (OS/390 版) または DB2 ユニバーサル・データベース (AS/400 版) 上のストアード・プロシージャーにはそのような機能はありません。ユーザーのストアード・プロシージャーからエラー・コードを返したい場合には、パラメーターとして渡す必要があります。SQLCODE および SQLCA は、システム検出エラー用にサーバーだけが設定します。

- 現時点では、ストアード・プロシージャーの結果セットを返すことができるホストまたは AS/400 アプリケーション・サーバーは、DB2 (VSE および VM 版) バージョン 7 以上または DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以上だけです。

### ストアード・プロシージャー・ビルダー

DB2 ストアード・プロシージャー・ビルダーは、ストアード・プロシージャーの作成、インストール、テストのための使いやすい開発環境です。これを使うことで、DB2 サーバー上におけるストアード・プロシージャーの登録、構築、インストールといった細かい作業に注意を奪われることなく、ストアード・プロシージャーのロジックを作成することに集中できます。さらに、ストアード・プロシージャー・ビルダーを使えば、あるオペレーティング・システム上でストアード・プロシージャーを開発し、別のサーバー・オペレーティング・システム上でそれらを構築することも可能になります。

ストアード・プロシージャー・ビルダーはグラフィカルなアプリケーションであり、迅速な開発をサポートしています。ストアード・プロシージャー・ビルダーを使えば、以下の作業を行うことができます。

- 新規のストアード・プロシージャーを作成する。
- ローカルおよびリモートの DB2 サーバー上でストアード・プロシージャーを構築する。
- 既存のストアード・プロシージャーを変更して再構築する。
- インストールしたストアード・プロシージャーの実行をテストしてデバッグする。

ストアード・プロシージャ・ビルダーは「DB2 ユニバーサル・データベース (DB2 Universal Database)」プログラム・グループから単独のアプリケーションとして起動できます。あるいは、以下のいずれかの開発用アプリケーションから起動することも可能です。

- Microsoft Visual Studio
- Microsoft Visual Basic
- IBM VisualAge for Java

さらに、DB2 (OS/390 版) のコントロール・センターからストアード・プロシージャ・ビルダーを起動することもできます。ストアード・プロシージャ・ビルダーは単独のプロセスとして、コントロール・センターの「ツール (Tools)」メニュー、ツールバー、「ストアード・プロシージャ (Stored Procedure)」フォルダーなどから開始することが可能です。また、「ストアード・プロシージャ・ビルダー・プロジェクト (Stored Procedure Builder Project)」ウィンドウからは、DB2 (OS/390 版) サーバーに組み込まれた SQL ストアード・プロシージャを 1 つ以上選択して、コマンド行プロセッサ (CLP) 内で実行できる指定されたファイルへエクスポートできます。

ストアード・プロシージャ・ビルダーは作業結果を管理するのにプロジェクトを使用します。それぞれのストアード・プロシージャ・ビルダー・プロジェクトには、特定のデータベース (DB2 (OS/390 版) サーバーなど) への接続が保管されます。また、フィルターを作成して各データベースごとにストアード・プロシージャのサブセットを表示させることも可能です。新規の、あるいは既存のストアード・プロシージャ・ビルダー・プロジェクトをオープンするときに、ストアード・プロシージャが名前、スキーマ、言語、集合 ID (OS/390 のみ) に基づいて表示されるよう、ストアード・プロシージャにフィルターをかけることができます。

接続情報はストアード・プロシージャ・ビルダー・プロジェクトに保管されます。そのため既存のプロジェクトをオープンすると、そのデータベースのユーザー ID とパスワードを入力するよう自動的に要求されます。「SQL ストアード・プロシージャの挿入 (Inserting SQL Stored Procedure)」ウィザードを使用すれば、SQL ストアード・プロシージャを DB2 (OS/390 版) サーバー上で構築できます。DB2 (OS/390 版) サーバーに組み込まれた SQL ストアード・プロシージャの場合、コンパイル、プリリンク、リンク、バインド、実行時、WLM 環境、外部セキュリティといったオプションを具体的に設定できます。

さらに、その SQL ストアード・プロシージャに関する SQL コスト情報を入手することもできます。これには、その SQL ストアード・プロシージャ

を実行しているスレッドの CPU 時間に関する情報やその他の DB2 コスト情報も含まれます。特に、ラッチ / ロック競合の待ち時間、getpage の数、読み取り入出力の回数、書き込み入出力の回数などに関するコスト情報を入手できます。

コスト情報を入手するにあたって、ストアード・プロシージャ・ビルダーは DB2 (OS/390 版) サーバーに接続し、SQL ステートメントを実行し、ストアード・プロシージャを呼び出す (DSNWSPM) ことにより、その SQL ストアード・プロシージャが使った CPU 時間の量を調べます。

## NOT ATOMIC 複合 SQL

複合 SQL では、複数の SQL ステートメントを単一の実行可能ブロックとしてグループにまとめることができます。このようにすればネットワーク・オーバーヘッドを減らすことができ、応答時間が改善されます。

DB2 コネクトは NOT ATOMIC 複合 SQL をサポートしています。このことは、複合 SQL の処理はエラーが生じた後にも続けられることを意味します。(DB2 コネクトではサポートされていない ATOMIC 複合 SQL を使用した場合、エラーが生じたときにはその複合 SQL グループ全体がロールバックされます。)

ステートメントは、アプリケーション・サーバーによって終了させられるまで実行し続けます。一般的に言って、複合 SQL ステートメントの実行が停止するのは重大エラーが発生した場合だけです。

NOT ATOMIC 複合 SQL は、サポートされているホストまたは AS/400 アプリケーション・サーバーすべてで使用することができます。

複数の SQL エラーが発生した場合、最初から 7 番目までの失敗したステートメントの SQLSTATE が、SQLCA の SQLERRMC フィールドに戻されます。その際、複数のエラーが発生したことを示すメッセージも一緒に戻されます。詳細については、SQL 解説書を参照してください。

## DB2 コネクトによる複数サイト更新

DB2 コネクトを使って、複数サイト更新 (2 フェーズ・コミットともいう) を実行することができます。複数サイト更新とは、単一の分散作業単位 (DUOW) 内で複数のデータベースを更新することです。ユーザーがこの機能を使用するかどうかはいくつかの要因によって決まります。

- アプリケーション・プログラムは CONNECT 2 および SYNCPOINT TWOPHASE オプションを指定してプリコンパイルしなければなりません。

- SNA ネットワーク接続の場合、sync コネクト エンタープライズ・エディション バージョン 7 (AIX 版、OS/2 版、および Windows NT 版) の DB2 同期点管理プログラム機能で提供される、2 フェーズ・コミット・サポートを使用することができます。このサポートを使用すると、以下のホスト・データベース・サーバーが 1 つの分散作業単位に参加できるようになります。
  - DB2 (AS/400 版) バージョン 3.1 以上 (ただし、以前の名称は DB2 AS/400 版)
  - DB2 (MVS/ESA 版) バージョン 3.1 以上
  - DB2 (OS/390 版) バージョン 5.1 以上
  - DB2 (VSE および VM 版) バージョン 5.1 以上

このサポートはネイティブの DB2 UDB アプリケーション、および外部のトランザクション処理 (TP) モニター (IBM TXSeries、CICS for Open Systems、BEA Tuxedo、Encina Monitor、および Microsoft Transaction Server など) に適用されます。

**注:** BEA Tuxedo については、40ページの『DB2 コネクトをトランザクション処理モニターと組み合わせて使う』を参照してください。XA コンセントレーターについては、157ページの『DB2 コネクトの接続コンセントレーター』を参照してください。

- TCP/IP ネットワーク接続を利用する場合、DB2 (OS/390 版) V5.1 以上のサーバーが 1 つの分散作業単位に参加することができます。IBM TXSeries、CICS for Open Systems、Encina Monitor、または Microsoft Transaction Server などのトランザクション処理モニターを使ってアプリケーションを制御している場合は、同期点管理プログラムを使用する必要があります。

TCP/IP 接続でホスト・データにアクセスするのにネイティブ DB2 アプリケーションと TP モニター・アプリケーションの両方が共通 DB2 コネクト エンタープライズ・エディション サーバーを使用しているなら、同期点管理プログラムは必ず使用しなければなりません。

1 つの DB2 コネクト エンタープライズ・エディション サーバーから SNA と TCP/IP ネットワーク・プロトコルの両方を使ってホスト・データにアクセスし、2 フェーズ・コミットが必要な場合は、同期点管理プログラムを使用しなければなりません。このことは DB2 アプリケーションと TP モニター・アプリケーションの両方に当てはまります。



## DB2 コネクトがサポートするホストまたは AS/400 サーバー SQL ステートメント

以下に挙げるステートメントは、ホストまたは AS/400 サーバーでの処理用に正常にコンパイルできますが、DB2 ユニバーサル・データベース・システムでの処理用には正常にコンパイルできません。

- ACQUIRE
- DECLARE (modifier.(qualifier.)table\_name TABLE ...
- LABEL ON

上記のステートメントは、コマンド行プロセッサでもサポートされていません。

以下に挙げるステートメントは、ホストまたは AS/400 サーバーでの処理用にサポートされていますが、バインド・ファイルやパッケージに付加されることはなく、コマンド行プロセッサによるサポートもありません。

- DESCRIBE statement\_name INTO descriptor\_name USING NAMES
- PREPARE statement\_name INTO descriptor\_name USING NAMES FROM ...

プリコンパイラーは以下のことを仮定します。

- ホスト変数が入力変数であること。
- ステートメントに固有セクション番号が割り当てられていること。

## DB2 コネクトが拒否するホストまたは AS/400 サーバー SQL ステートメント

以下の SQL ステートメントは、DB2 コネクトでもコマンド行プロセッサでもサポートされていません。

- COMMIT WORK RELEASE
- DECLARE state\_name, statement\_name STATEMENT
- DESCRIBE statement\_name INTO descriptor\_name USING xxxx (ここで、xxxx は ANY、BOTH、または LABELS)
- PREPARE statement\_name INTO descriptor\_name USING xxxx FROM :host\_variable (ここで、xxxx は ANY、BOTH、または LABELS)
- PUT ...
- ROLLBACK WORK RELEASE
- SET :host\_variable = CURRENT ...

DB2 (VSE および VM 版) により拡張された動的 SQL ステートメントは、-104 および構文エラー SQLCODE で拒否されます。

## DB2 ユニバーサル・データベース (OS/390 版) でのチャージバック会計の実装

多くの DB2 ユニバーサル・データベース (OS/390 版) インストール・システムは、リソースのモニター手法を実装しており、これによってシステム管理者は、リソースの利用を各ユーザーのアクセスに関連付けることができます。これは、リソースを消費する個々のユーザーまたはその部門に支払い請求するために使用されます。この手法を一般に、チャージバック会計と言います。

DB2 コネクト製品を使用すると、システム管理者は、DB2 コネクト経由でデータベースにアクセスしているユーザーが消費するメインフレーム・リソースをモニターすることができます。会計ストリングを使用して会計データを DB2 コネクトからデータベース・サーバーの DB2 へ送信することができます。会計ストリングは、システム生成データとユーザー提供データとを組み合わせます。このデータによって、システム管理者は、リソース利用を各ユーザーのアクセスと関連付け、それに従ってユーザーに請求することができます。

会計ストリングは、DRDA パラメーター PRDDTA を使用して送信されます。このパラメーターの内容は DRDA では体系化されていないため、ユーザーのアプリケーション・サーバーがそのデータを会計データとして認識する保証はありません。現時点では、PRDDTA がサポートされるのは MVS システムと OS/390 システムだけです。ストリングは会計記録として保管されます。

会計ストリングは、DB2 コネクトで生成された 56 バイト (接頭部) とそれに続くユーザー指定の最大 199 バイト (接尾部) より成り、最大長は 255 バイトです。

表3 は、システム生成のフィールドを示しています。これらフィールドのおのおのは、右方にブランクが埋め込まれています。

表3. DB2 コネクトによって生成される会計ストリング・フィールド

フィールド名	長さ	説明
acct_str_len	1	会計ストリングの長さマイナス 1 を示す 16 進数の値 (X'3C' など)。

表 3. DB2 コネクトによって生成される会計ストリング・フィールド (続き)

フィールド名	長さ	説明
client_prdid	8	クライアント・ソフトウェアの製品 ID。たとえば、DB2 ユニバーサル・データベース バージョン 7 の製品 ID は SQL07010 です。
client_platform	18	クライアントが乗っているプラットフォーム。たとえば、AIX、OS/2、DOS、または Windows。
client_appl_name	20	ユーザーのアプリケーション名の最初の 20 文字。たとえば、payroll。
client_authid	8	ユーザー・アプリケーションの authid。たとえば、SMITH。
suffix_len	1	ユーザー提供の接尾部の長さを示す 16 進数の値。X'00' は、ユーザー提供の接尾部がないことを意味します。

ユーザー定義の接尾部は、次のいずれかです。

- sqlesact() API を有するアプリケーションによって指定される値
- DB2ACCOUNT 環境変数の値
- DFT\_ACCOUNT\_STR (省略時会計ストリング) 構成パラメーターの値
- ヌル・ストリング

接尾部が 199 文字より長い場合は切り捨てられます。会計ストリングがホストまたは AS/400 データベース・サーバーへ送信される時、会計ストリングが正しく変換されていることを確認してください。文字 A~Z、0~9、および下線 ( ) だけを使用します。

会計ストリングの設定について API 方式が推奨されます。ユーザーのアプリケーションは、データベースに接続する前に API を呼び出してください。アプリケーション内の会計ストリングを変更したい場合 (たとえば、異なるデー

データベースへ接続するときに異なるストリングを送信したい場合)、再度 API を呼び出します。 そのようにしなければ、その値はアプリケーションの終了まで有効のままです。

sqlsact() API が 1 番目のデータベース接続要求の前に呼び出されなかった場合は、DB2ACCOUNT 環境変数が読み取られます。この値は、アプリケーションの終了またはバックグラウンドでのコマンド行プロセッサ・プロセスの終了まで有効のままです。 1 番目のデータベース接続の後で新規の会計ストリング接尾部を指定するには、sqlsact() API を使用するか、またはアプリケーションもしくはバックグラウンドの CLP プロセスを終了させてから DB2ACCOUNT をその新規の値にセットして再始動します。

DB2ACCOUNT の値が存在しない場合には、DFT\_ACCOUNT\_STR システム構成パラメーターの値が使用されます。この省略時値は、会計ストリングを DB2 コネクトに送る能力のないデータベース・クライアントにとって有用になり得ます。これが存在しない場合は、ヌル・ストリングが使用されます。

以下に会計ストリングの例をいくつか挙げます。

x'3C'SQL070100S/2	cheque	SMITH	x'05'DEPT1
x'37'SQL070100S/2	cheque	SMITH	x'00'

1 番目の例は、ユーザー定義の接尾部が DEPT1 です。 2 番目の例では、それがヌル・ストリングです。

---

## 会計情報を DB2 (OS/390 版) サーバーに送信する

メインフレームの多くの顧客にとって、さまざまなアプリケーションで使用されるリソースの詳細な会計を作成することは、操作手順の重要な部分です。DB2 (OS/390 版) には会計レポートを作成する広範囲な機能があり、情報システム部門はこれを使用して、メインフレームのリソースの使用状況に応じて個々のユーザー部門に課金できます。このプロセスはしばしば、チャージバック会計と呼ばれています。DB2 コネクト製品では、既存の会計レポートと会計手順を使用して、PC と UNIX のアプリケーションに使用されたホスト・リソースの正確な会計を作成できます。

DB2 コネクトでは以下のものが備えられており、この機能に柔軟性を持たせています。

- 特定の DB2 コネクト エンタープライズ・エディション・サーバーで生成されたすべての使用状況に関する省略時の会計ストリング。

- 個々のユーザーまたはアプリケーションが、自分の使用状況に応じて課金する会計を指定するための 2 種類のメカニズム。

## 会計ストリングの設定

省略時の会計ストリングは、DB2 コネクト・ワークステーションの `dft_account_str` 構成パラメーターで設定されています。この省略時メカニズムは、会計ストリングを DB2 コネクトに送る機能のないデータベース・クライアント (たとえば、バージョン 2 以前の製品で開発されたアプリケーション) にとって有効です。

クライアント・アプリケーションで、DB2 コネクト・サーバーに設定されている省略時の会計ストリングを上書きする方法には、以下の 2 つがあります。

- 会計ストリング設定 API である `sqlesact()` を使用する。

この `sqlesact()` API は、アプリケーションがデータベースに接続する前に呼び出されます。この方式の利点は次のとおりです。

- API を呼び出すと、レジストリー値を読み取るコストがかからない。
- 将来の接続要求のために新しい会計ストリングを使用しない限り、この API を再呼び出しする必要がない。レジストリー値を使用している場合は、アプリケーション・プロセスを終了し、DB2ACCOUNT を再定義してから、このプロセスを再始動する必要があります。

この API の使用方法について詳しくは、*管理 API 解説書* を参照してください。

- クライアント・ワークステーションで DB2ACCOUNT レジストリー値を使用する。

`sqlesact()` API が 1 番目のデータベース接続要求の前に呼び出されなかった場合は、DB2ACCOUNT レジストリー値が読み取られます。会計ストリングは、これ以降のすべてのデータベース接続要求に使用されます。

**注:** 会計ストリングを定義する際には、以下の規則に注意してください。

1. 文字 A~Z、0~9、または `'_'` (下線) を使用する。
2. 会計ストリングを 199 バイトに制限する。これより長いストリングは切り捨てられます。

---

## 参考資料

以下の資料は、分散環境で実行するアプリケーションの開発に役立ちます。

- 特定のデータベース製品についてのアプリケーション・プログラミング資料は、製品ごとに異なる情報を掲載しています。

- 特定のデータベース製品についての SQL 解説書は、アプリケーションが、正確な構文を持ち、サポートされる SQL ステートメントだけを含むことを確認するのに役立ちます。
- DB2 (OS/390 版) リモート DRDA リクエスター / サーバー解説書 は、DB2 ユニバーサル・データベース (OS/390 版) ユーザーのチャージバック会計に関する最新情報を提供します。
- SQL 解説書 は、IBM 関係データベース製品間の差異についてのレベルの高い論議と、いくつかの特定の差異を取り扱う方法についての説明を記載しています。
- DRDA の出版物は、DRDA 環境における計画、接続性、プログラミング、および問題判別についての情報を提供します。表題および資料番号のリストについては、22ページの『DRDA 関連のオンライン資料』を参照してください。

---

## 第5章 独自のアプリケーションの実行

さまざまなタイプのアプリケーションで、DB2 データベースにアクセスすることができます。

- DB2 アプリケーション開発クライアント (組み込み SQL、API、ストアード・プロシージャ、ユーザー定義機能、または DB2 CLI に対する呼び出しを含む) を使用して開発されたアプリケーション。
- ODBC アプリケーション (Lotus Approach など)。
- JDBC アプリケーションおよびアプレット。
- Net.Data マクロ (HTML および SQL を含む)。

DB2 クライアント上のアプリケーションは、物理的な位置を知ることなしにリモート・データベースにアクセスできます。DB2 クライアントは、データベースの位置を判別し、データベース・サーバーに対する要求の伝送を管理して、その結果を戻します。

通常、データベース・クライアント・アプリケーションを実行するには、以下のステップに従います。

ステップ 1. サーバーが構成されて実行されていることを確かめます。

アプリケーション・プログラムの接続先のデータベース・サーバーでデータベース・マネージャーが開始済みであることを確かめます。そうでない場合、アプリケーションを開始する前に、サーバーで **db2start** コマンドを発行する必要があります。

ステップ 2. アプリケーションが使用するデータベースに接続できることを確認します。

ステップ 3. ユーティリティとアプリケーションをデータベースにバインドします。ユーティリティのバインドについては、70ページの『データベース・ユーティリティのバインド』を参照してください。

ステップ 4. アプリケーション・プログラムを実行します。

---

## データベース・ユーティリティーのバインド

データベース・ユーティリティー (インポート、エクスポート、REORG、コマンド行プロセッサ) や DB2 CLI バインド・ファイルをデータベースとともに使用できるようにするには、その前にそれらをそのデータベースにバインドする必要があります。ネットワーク環境では、異なるオペレーティング・システムで稼働したり DB2 のバージョンやサービス・レベルが異なっていたりする複数のクライアントを使用する場合に、それぞれのオペレーティング・システムと DB2 バージョンの組み合わせごとに 1 度ずつユーティリティーをバインドする必要があります。

ユーティリティーをバインドすると、パッケージが作成されます。これは、単一のソース・ファイルから特定の SQL ステートメントを処理するのに必要なすべての情報を含むオブジェクトです。

バインド・ファイルは、インストール・ディレクトリーの下にある bnd ディレクトリーで、さまざまな .lst ファイルと一緒にグループ化されます。各ファイルはサーバーごとに固有です。

---

## CLI/ODBC プログラムの実行

DB2 コール・レベル・インターフェース (CLI) 実行時環境と DB2 CLI/ODBC ドライバーは、インストール中に任意選択のコンポーネントとして DB2 クライアントに組み込まれます。

このサポートを使用すると、ODBC や DB2 CLI API を使って開発したアプリケーションは、任意の DB2 サーバーと一緒に動作できるようになります。DB2 CLI アプリケーション開発サポートは、DB2 サーバーのパッケージにある DB2 アプリケーション開発クライアントによって提供されます。

DB2 CLI または ODBC アプリケーションが DB2 にアクセスするには、サーバーに DB2 CLI パッケージをバインドする必要があります。ユーザーがパッケージをバインドする必須権限を持っている場合、これは最初の接続時に自動的に行われますが、サーバーにアクセスする各プラットフォーム上のそれぞれのバージョンで、管理者の方がまずこのことを行うようお勧めします。

DB2 CLI および ODBC が DB2 データベースにアクセスするには、クライアント・システムで、以下に示す一般的なステップに従う必要があります。ここで示す指示は、有効なユーザー ID とパスワードを使用して、DB2 に正常に接続したことを前提としています。プラットフォームによっては、これらのステップの多くは自動化されています。完全な詳細については、ご使用のプラットフォームについて特に扱っているセクションを参照してください。



- ステップ 1. (クライアント・マシンとサーバー・マシンが別々の場合) クライアント構成アシスタント (CCA) を使用してデータベースを追加し、そのインスタンスとデータベースをコントロール・センターが認識できるようにしてから、そのシステムのインスタンスとデータベースを追加します。このプログラムにアクセスできない場合には、コマンド行プロセッサで **catalog** コマンドを使用することができます。
- ステップ 2. DB2 CLI/ODBC ドライバーは、Windows プラットフォームで DB2 クライアントをインストールするときに任意に選択できる構成要素です。インストール時にこのドライバーを選択したことを確認してください。OS/2 の場合は、「**ODBC ドライバーのインストール (Install ODBC Driver)**」アイコンを使用して、DB2 CLI/ODBC ドライバーと ODBC ドライバー・マネージャーの両方をインストールする必要があります。UNIX プラットフォームの場合、DB2 CLI/ODBC ドライバーは自動的にクライアントとともにインストールされます。
- ステップ 3. ODBC から DB2 データベースにアクセスするには、以下のようになります。
- a. ODBC ドライバー・マネージャー (Microsoft または他のベンダーが提供するもの) がインストールされている必要があります (32 ビット Windows システムの場合にのみ、これは DB2 のインストール中に省略時設定で行われます)。
  - b. DB2 データベースは、ODBC データ・ソースとして登録されている必要があります。ODBC ドライバー・マネージャーは、DB2 カタログ情報を読み取る代わりに、独自のデータ・ソース・リストを参照します。
  - c. DB2 表に固有索引が入っていない場合、多くの ODBC アプリケーションはその表を読み取り専用として開きます。固有索引は、ODBC アプリケーションによって更新される DB2 表ごとに 1 つずつ作成する必要があります。SQL 解説書で **CREATE INDEX** ステートメントを参照してください。コントロール・センターで、表の設定を変更してから、「**基本キー (Primary Key)**」タブを選択して、利用可能な列リストから基本キー列リストへと、1 つまたは複数の列を移動します。基本キーの一部として選択する列はすべて、NOT NULL として定義する必要があります。

ステップ 4. 必要であれば、さまざまな CLI/ODBC 構成キーワードを設定し、DB2 CLI/ODBC や、DB2 CLI/ODBC を使うアプリケーションの動作を変更することができます。

上記のステップに従って ODBC サポートをインストールし、DB2 データベースを ODBC データ・ソースとして追加したら、ODBC アプリケーションはそれらにアクセスできるようになります。

## CLI/ODBC のアクセスに関するプラットフォーム特有の詳細



DB2 CLI および ODBC アプリケーションが DB2 にアクセスできるようにする方法に関するプラットフォーム固有の詳細については、以下のカテゴリに分けて説明します。

- 『CLI/ODBC による Windows 32 ビット オペレーティング・システムから DB2 へのクライアント・アクセス』
- 75ページの『CLI/ODBC による OS/2 から DB2 へのクライアント・アクセス』

### CLI/ODBC による Windows 32 ビット オペレーティング・システムから DB2 へのクライアント・アクセス

DB2 CLI および ODBC アプリケーションが Windows クライアントから DB2 データベースに正常にアクセスできるようにするには、その前にクライアント・システムで以下のステップを実行します。

ステップ 1. DB2 データベース (およびリモート・データベースの場合にはノードも) をカタログ化する必要があります。これを行うには、CCA (またはコマンド行プロセッサ) を使用します。

詳しくは、CCA のオンライン・ヘルプ (またはコマンド解説書で **CATALOG DATABASE** および **CATALOG NODE** コマンド) を参照してください。

ステップ 2. Microsoft ODBC Driver Manager および DB2 CLI/ODBC ドライバーがインストールされていることを確かめます。Windows 32 ビット オペレーティング・システムでは、インストール中に ODBC 構成要素を手動で選択解除していなければ、両方とも DB2 とともにインストールされています。新しいバージョンの Microsoft ODBC Driver Manager が見つかった場合、DB2 はそれを上書きしません。

それらが両方ともマシンに存在しているかどうかを調べるには、以下のようにします。

- a. コントロール・パネルで「Microsoft ODBC データ ソース」アイコンを開始するか、コマンド行から **odbcad32.exe** コマンドを実行します。
- b. 「ドライバ」タブをクリックします。
- c. リストに「IBM DB2 ODBC ドライバー (IBM DB2 ODBC DRIVER)」が表示されていることを確かめます。

Microsoft ODBC Driver Manager または IBM DB2 CLI/ODBC ドライバーのいずれかがインストールされていない場合には、Windows 32 ビット オペレーティング・システムで DB2 のインストールをやり直し、ODBC 構成要素を選択します。

ステップ 3. DB2 データベースをデータ・ソースとして ODBC ドライバー・マネージャーに登録します。Windows 32 ビット オペレーティング・システムでは、システムのすべてのユーザーがデータ・ソースを利用できるようにするか (システム・データ・ソースにする)、現行のユーザーだけが利用できるようにする (ユーザー・データ・ソースにする) ことができます。いずれかの方法を使用して、データ・ソースを追加します。

- CCA を使用する場合:
  - a. データ・ソースとして追加する DB2 データベース別名を選択します。
  - b. 「プロパティ (Properties)」押しボタンをクリックします。「データベースのプロパティ (Database Properties)」ウィンドウが表示されます。
  - c. 「このデータベースを ODBC 用に登録 (Register this database for ODBC)」チェック・ボックスを選択します。
  - d. Windows 32 ビット オペレーティング・システムでは、ラジオ・ボタンを使用して、データ・ソースをユーザー・データ・ソースまたはシステム・データ・ソースのいずれかとして追加することができます。
- Microsoft 32 ビット ODBC 管理ツール (コントロール・パネルのアイコンから、またはコマンド行で **odbcad32.exe** を実行することによりアクセス可能) を使用する場合:
  - a. Windows 32 ビット オペレーティング・システムでは、省略時設定でユーザー・データ・ソース・リストが表示されます。システム・データ・ソースを追加する場合には、「システム DSN」ボタンか「システム DSN」タブをクリックします (プラットフォームによって異なります)。

- b. 「追加」押しボタンをクリックします。
  - c. リストにある「IBM DB2 ODBC ドライバー (IBM DB2 ODBC Driver)」をダブルクリックします。
  - d. 追加する DB2 データベースを選択して、「OK」をクリックします。
- Windows 32 ビット オペレーティング・システムには、DB2 データベースを ODBC ドライバー・マネージャーにデータ・ソースとして登録するコマンド (コマンド行プロセッサで発行可能) があります。管理者は、必要なデータベースを登録するコマンド行プロセッサ・スクリプトを作成することもできます。その場合、このスクリプトを、ODBC を介して DB2 データベースにアクセスする必要があるすべてのマシンで実行することができます。

CATALOG コマンドに関する詳しい情報は、コマンド解説書に説明されています。

```
CATALOG [ user | system ] ODBC DATA SOURCE
```

ステップ 4. CCA を使用して DB2 CLI/ODBC ドライバーを構成します (任意選択)。

- a. 構成する DB2 データベース別名を選択します。
- b. 「プロパティ (Properties)」押しボタンをクリックします。「データベースのプロパティ (Database Properties)」ウィンドウが表示されます。
- c. 「設定 (Settings)」押しボタンをクリックします。「CLI/ODBC の設定 (CLI/ODBC Settings)」ウィンドウが開きます。
- d. 「拡張機能 (Advanced)」押しボタンをクリックします。開いたウィンドウで構成キーワードを設定することができます。これらのキーワードはデータベースの別名に関連付けられ、そのデータベースにアクセスするすべての DB2 CLI/ODBC アプリケーションに影響を及ぼします。オンライン・ヘルプには、すべてのキーワードについての説明があります。この説明は、インストールおよび構成 補足 というオンライン・マニュアルにもあります。

ステップ 5. (上記のように) ODBC アクセスをインストールし終えたら、ODBC アプリケーションを使用して DB2 データにアクセスできるようになります。ODBC アプリケーションを起動して、「開く (Open)」ウィンドウに移動します。「ODBC データベース

**(ODBC databases)**」ファイル・タイプを選択します。 ODBC データ・ソースとして追加した DB2 データベースをリストから選択することができます。 固有索引が存在しないかぎり、多くの ODBC アプリケーションは表を読み取り専用として開きます。

### CLI/ODBC による OS/2 から DB2 へのクライアント・アクセス

DB2 CLI および ODBC アプリケーションが OS/2 クライアントから DB2 データベースに正常にアクセスできるようにするには、その前にクライアント・システムで以下のステップを実行します。

1. DB2 データベース (およびリモート・データベースの場合にはノードも) をカタログ化する必要があります。これを行うには、CCA (またはコマンド行プロセッサ) を使用します。

詳しくは、CCA のオンライン・ヘルプ (またはコマンド解説書 で

**CATALOG DATABASE** および **CATALOG NODE** コマンド) を参照してください。

2. ODBC アプリケーションを使用して DB2 データにアクセスする場合には、以下のステップを実行してください。 (CLI アプリケーションだけを使用する場合は、このステップをスキップして次のステップに進んでください。)

- a. ODBC ドライバー・マネージャーがインストールされていることを確かめます。 ODBC ドライバー・マネージャーは DB2 と一緒にインストールされることはありません。 ODBC アプリケーションに付属のドライバー・マネージャーを使用するようお勧めします。 DB2 CLI/ODBC ドライバーがインストールされていることも確かめます。

- 1) ODBC 管理ツールを、マニュアルで説明されているとおりに実行します。通常これは、以下の 2 つの方法のいずれかで行います。

- OS/2 で、「ODBC」フォルダーをダブルクリックして、「ODBC アドミニストレーター (ODBC Administrator)」アイコンをダブルクリックします。

- コマンド行から **odbcadm.exe** を実行します。

「データ・ソース (Data Sources)」ウィンドウが開きます。

- 2) 「**ドライバー (Drivers)**」押しボタンをクリックします。「ドライバー (Drivers)」ウィンドウが開きます。

- 3) リストに「IBM DB2 ODBC ドライバー (IBM DB2 ODBC DRIVER)」が表示されていることを確かめます。

ODBC ドライバー・マネージャーがインストールされていない場合は、ODBC アプリケーションに付属しているインストールに関する指示に従

ってください。 IBM DB2 CLI/ODBC ドライバーがインストールされていない場合は、 DB2 フォルダの「**ODBC ドライバーのインストール (Install ODBC Driver)**」アイコンをダブルクリックして、 DB2 CLI/ODBC ドライバーをインストールします。

- b. 次のいずれかの方法を使用して、 DB2 データベースをデータ・ソースとして ODBC ドライバー・マネージャーに登録します。
  - CCA を使用する場合:
    - 1) データ・ソースとして追加する DB2 データベース別名を選択します。
    - 2) 「**プロパティ (Properties)**」押しボタンをクリックします。
    - 3) 「**このデータベースを ODBC 用に登録 (Register this database for ODBC)**」チェック・ボックスを選択します。
  - ODBC ドライバー・マネージャーを使用する場合:
    - 1) ODBC ドライバー・マネージャーを、マニュアルで説明されているとおりに実行します。通常これは、以下の 2 つの方法のいずれかで行います。
      - OS/2 で、「**ODBC**」フォルダをダブルクリックして、「**ODBC アドミニストレーター (ODBC Administrator)**」アイコンをダブルクリックします。
      - コマンド行から **odbcadm.exe** を実行します。
    - 2) 「データ・ソース (Data Source)」ウィンドウで、「**追加 (Add)**」押しボタンをクリックします。「データ・ソースの追加 (Add Data Source)」ウィンドウが開きます。
    - 3) リストにある「**IBM DB2 ODBC ドライバー (IBM DB2 ODBC DRIVER)**」をダブルクリックします。
    - 4) 追加する DB2 データベースを選択して、「**OK**」をクリックします。
3. CCA を使用して DB2 CLI/ODBC ドライバーを構成します (任意選択).
  - a. 構成する DB2 データベース別名を選択します。
  - b. 「**プロパティ (Properties)**」押しボタンをクリックします。「データベースのプロパティ (Database Properties)」ウィンドウが表示されます。
  - c. 「**設定 (Settings)**」押しボタンをクリックします。「CLI/ODBC の設定 (CLI/ODBC Settings)」ウィンドウが開きます。
  - d. 「**拡張機能 (Advanced)**」押しボタンをクリックします。表示されたウィンドウで構成キーワードを設定することができます。これらのキーワ

ードはデータベースの別名に関連付けられ、そのデータベースにアクセスするすべての DB2 CLI/ODBC アプリケーションに影響を及ぼします。オンライン・ヘルプには、すべてのキーワードについての説明があります。この説明は、 および インストールおよび構成 補足 にもあります。

4. (上記のように) ODBC アクセスをインストールし終えたら、ODBC アプリケーションを使用して DB2 データにアクセスできるようになります。ODBC アプリケーションを起動して、「開く (Open)」ウィンドウに移動します。「**ODBC データベース (ODBC databases)**」ファイル・タイプを選択します。ODBC データ・ソースとして追加した DB2 データベースをリストから選択することができます。固有索引が存在しないかぎり、多くの ODBC アプリケーションは表を読み取り専用として開きます。

## 詳細な構成情報

72ページの『CLI/ODBC のアクセスに関するプラットフォーム特有の詳細』というセクションで、必要な情報はすべて入手できます。インストールおよび構成 補足 には、DB2 CLI および ODBC アプリケーションのセットアップと使用に関する追加情報が含まれています。(オンラインの インストールおよび構成 補足 資料は、`x:\doc\en\html` というディレクトリーに置かれています。ここで、`x:` はユーザーの CD-ROM を表す文字、`en` はユーザーの言語を表す 2 文字の国別コード (たとえば、英語の場合は `en`) です。) この情報は、DB2 ツール・サポートが利用できない場合に役に立ちます。また、さらに詳しい情報を必要としている管理者にとっても有用です。

インストールおよび構成 補足 オンライン資料では、以下のトピックが扱われています。

- DB2 CLI/ODBC ドライバーをデータベースにバインドする方法
- CLI/ODBC 構成キーワードを設定する方法
- `db2cli.ini` の構成

---

## Java プログラムの実行

AIX、HP-UX、Linux、OS/2、DYNIX/ptx、Silicon Graphics IRIX、Solaris Operating Environment、または Windows 32 ビット オペレーティング・システムで、適切な Java 開発者キット (JDK) を使用して、DB2 にアクセスする Java プログラムを作成することができます。JDK には、Java 用の動的 SQL API である Java Database Connectivity (JDBC) が含まれています。

DB2 JDBC サポートを使用するには、DB2 クライアントのインストール時に DB2 Java Enablement 構成要素を含める必要があります。DB2 JDBC サポートを使用すると、JDBC アプリケーションおよびアプレットを作成して実行することができます。JDBC アプリケーションおよびアプレットには動的 SQL のみが含まれており、Java 呼び出しインターフェースを使用して、SQL ステートメントを DB2 に渡します。

DB2 アプリケーション開発クライアントにより、Java Embedded SQL (SQLJ) のサポートが提供されています。DB2 SQL サポートおよび DB2 JDBC サポートを使用すると、SQLJ アプリケーションおよびアプレットを作成して実行することができます。SQLJ アプリケーションおよびアプレットには静的 SQL が含まれており、DB2 にバインドされている組み込み SQL ステートメントを使用します。

Java をサーバー上で使用して、JDBC および SQLJ ストアド・プロシージャおよびユーザー定義関数 (UDF) を作成することもできます。

さまざまなタイプの Java プログラムを作成して実行するには、DB2 のさまざまな構成要素からのサポートが必要です。

- JDBC アプリケーションを作成するには、DB2 クライアントとともに DB2 Java Enablement 構成要素をインストールする必要があります。JDBC アプリケーションを実行するには、DB2 Java Enablement 構成要素がインストールされた DB2 クライアントで、DB2 サーバーに接続する必要があります。
- SQLJ アプリケーションを作成するには、DB2 Application Development Client および DB2 アドミニストレーション・クライアントとともに DB2 Java Enablement 構成要素をインストールする必要があります。SQLJ アプリケーションを実行するには、DB2 Java Enablement 構成要素がインストールされた DB2 クライアントで、DB2 サーバーに接続する必要があります。
- JDBC アプレットを作成するには、DB2 クライアントとともに DB2 Java Enablement 構成要素をインストールする必要があります。JDBC アプレットを実行する場合、クライアント・マシンに DB2 構成要素は必要ありません。
- SQLJ アプレットを作成するには、DB2 Application Development Client および DB2 アドミニストレーション・クライアントとともに DB2 Java Enablement 構成要素をインストールする必要があります。SQLJ アプレットを実行する場合、クライアント・マシンに DB2 構成要素は必要ありません。



JDBC および SQLJ プログラムの作成と実行に関する詳細については、アプリケーション構築の手引きを参照してください。Java での DB2 プログラミングについては、アプリケーション開発の手引きを参照してください。この資料では、JDBC および SQLJ アプリケーション、アプレット、ストアド・プロシージャ、および UDF の作成や実行について扱っています。

更新された最新の DB2 Java 情報については、以下の Web サイトにアクセスしてください。 <http://www.ibm.com/software/data/db2/java>

## 環境の構成

DB2 Java プログラムを作成するには、適切なバージョンの Java 開発者キット (JDK) を開発マシンにインストールして構成する必要があります。DB2 Java アプリケーションを実行するには、適切なバージョンの Java Runtime Environment (JRE) または JDK を開発マシンにインストールして構成する必要があります。以下のリストには、ご使用の開発マシンに適した JDK のバージョンがリストされています。

**AIX** IBM AIX Developer Kit, Java Technology Edition バージョン 1.1.8。  
AIX システムに JDK がインストールされていない場合、この JDK が DB2 Application Development Client とともにインストールされます。

### HP-UX

Hewlett Packard が提供する HP-UX Developer's Kit for Java リリース 1.1.8。

**Linux** IBM Developer Kit for Linux, Java Technology Edition バージョン 1.1.8。

**OS/2** IBM Java 開発者キット for OS/2 バージョン 1.1.8。製品 CD-ROM に入っています。

### DYNI/ptx

IBM 提供の DYNI/ptx/JSE バージョン 1.2.1。

### SGI IRIX

SGI 提供の Java 2 Software Development Kit for SGI IRIX バージョン 1.2.1。

### Solaris Operating Environment

Sun Microsystems 提供の Java 開発者キット for Solaris バージョン 1.1.8。

### Windows 32 ビット オペレーティング・システム

IBM Developer Kit for Windows 32 ビット オペレーティング・システム, Java Technology Edition バージョン 1.1.8。 DB2 Application

Development Client をインストールするときに、この JDK は `sqllib¥java¥jdk` ディレクトリーに自動的にインストールされます。

上記の JDK のインストールおよび構成について詳しくは、以下の URL を参照してください。 <http://www.ibm.com/software/data/db2/java>

サポートされているすべてのプラットフォームについて、DB2 クライアントとともに DB2 Java Enablement 構成要素をインストールして構成する必要があります。SQLJ プログラムをデータベースにバインドするには、DB2 アドミニストレーション・クライアントとともに DB2 Java Enablement 構成要素をインストールして構成する必要があります。

DB2 Java ストアード・プロシージャまたは UDF を実行するには、DB2 データベース・マネージャーの構成を変更して、開発マシン上で JDK バージョン 1.1 がインストールされている位置へのパスを含める必要もあります。これを行うには、コマンド行で以下のコマンドを入力します。

#### Windows および OS/2 プラットフォームの場合:

```
db2 update dbm cfg using JDK11_PATH C:¥sqllib¥java¥jdk
```

ここで、`C:¥sqllib¥java¥jdk` は、JDK がインストールされている位置へのパスを表します。

DB2 データベース・マネージャーの構成を検査して、`JDK11_PATH` フィールドの値が正しいかどうかを調べるには、次のコマンドを入力します。

```
db2 get dbm cfg
```

出力をファイルにパイプ接続すれば、表示を見やすくすることができます。`JDK11_PATH` フィールドは、出力の最初の方で表示されます。これらのコマンドの詳細については、[コマンド解説書](#) を参照してください。



Solaris Operating Environment の場合、一部の Java 仮想マシンが、「setuid」環境で実行されるプログラムで正常に動作しません。Java インタープリターを含む共用ライブラリー (`libjava.so`) が、正常にロードできない場合もあります。これを回避するため、以下のようなコマンドを使用して、`/usr/lib` 内にある必要なすべての JVM 共用ライブラリーへのシンボリック・リンクを作成することができます。

```
ln -s /opt/jdk1.1.3/lib/sparc/native_threads/*.so /usr/lib
```

この回避方法や他の方法に関する詳細については、以下の URL にアクセスしてください。 <http://www.ibm.com/software/data/db2/java>

Java プログラムを実行するため、OS/2 および Windows オペレーティング・システムに DB2 をインストールする際、および UNIX プラットフォームでインスタンスを作成する際に、以下の環境変数が自動的に更新されます。

#### UNIX プラットフォームの場合:

- CLASSPATH に、「.」および sqllib/java/db2java.zip ファイルが含まれます。
- AIX、Linux、DYNIX/ptx、Silicon Graphics IRIX、および Solaris Operating Environment の場合、LD\_LIBRARY\_PATH に sqllib/lib ディレクトリーが含まれます。
- HP-UX の場合、SHLIB\_PATH に sqllib/lib ディレクトリーが含まれます。
- Solaris Operating Environment の場合のみ、THREADS\_FLAG が「native」に設定されます。

#### Windows および OS/2 プラットフォームの場合:

- CLASSPATH に、「.」および %DB2PATH%\%java%\db2java.zip ファイルが含まれます。

SQLJ プログラムを作成および実行するため、こうしたファイルが含まれるよう CLASSPATH も自動的に更新されます。

#### UNIX プラットフォームの場合:

- sqllib/java/sqlj.zip (SQLJ プログラムの作成に必要)
- sqllib/java/runtime.zip (SQLJ プログラムの実行に必要)

#### Windows および OS/2 プラットフォームの場合:

- %DB2PATH%\%java%\sqlj.zip (SQLJ プログラムの作成に必要)
- %DB2PATH%\%java%\runtime.zip (SQLJ プログラムの実行に必要)

## Java アプリケーション

実行可能プログラムで Java インタープリターを実行し、デスクトップまたはコマンド行からアプリケーションを開始するには、以下のコマンドを実行します。

```
java prog_name
```

ここで、prog\_name はプログラムの名前です。

DB2 JDBC ドライバーは、アプリケーションからの JDBC API 呼び出しを処理し、DB2 クライアントを使用してサーバーに要求を送り、その結果を受け取ります。SQLJ アプリケーションは、実行する前にデータベースにバインドする必要があります。

## Java アプレット

Java アプレットは Web 上で送信されるため、ご使用の DB2 マシン (サーバーまたはクライアント) に Web サーバーをインストールする必要があります。

アプレットを実行するには、.html ファイルが正しく構成されていることを確かめる必要があります。.html ファイルで指定された TCP/IP ポートで JDBC アプレット・サーバーを開始してください。たとえば、次のように指定した場合、

```
param name=port value='6789'
```

以下のように入力します。

```
db2jstrt 6789
```

作業ディレクトリーがご使用の Web ブラウザーからアクセスできることを確かめる必要があります。アクセスできない場合、アプレットの .class および .html ファイルを、アクセス可能なディレクトリーにコピーします。SQLJ アプレットの場合、プロファイル .class および .ser ファイルもコピーする必要があります。

sqllib/java/db2java.zip ファイルをこれら他のファイルと同じディレクトリーにコピーします。SQLJ アプレットの場合、sqllib/java/runtime.zip ファイルもこのディレクトリーにコピーします。それから、クライアント・マシンで、JDK 1.1 をサポートする Web ブラウザーを起動して、.html ファイルをロードします。

アプレットが DB2 に接続するために JDBC API を呼び出すと、JDBC ドライバーは DB2 サーバーに存在する JDBC アプレット・サーバーを介して、DB2 データベースに対する別個の通信を確立します。SQLJ アプレットは、実行する前にデータベースにバインドする必要があります。

---

## 第2部 リファレンスおよびトラブルシューティング



---

## 第6章 データベース・ディレクトリーの更新

DB2 コネクトでは、接続先のデータベースについての情報を管理するために、以下のディレクトリーを使用します。

- ノード・ディレクトリー。このディレクトリーには、DB2 コネクトがアクセスするすべてのホストまたは AS/400 データベース・サーバーのネットワーク・アドレスや通信プロトコル情報が含まれます。
- データベース接続サービス (DCS) ディレクトリー。このディレクトリーには、ホストまたは AS/400 データベース・サーバー・データベースに固有の情報が含まれます。
- システム・データベース・ディレクトリー。DB2 コネクトがアクセスするすべてのデータベースの名前とロケーションの情報が含まれます。

### 注:

1. これらのディレクトリーを更新する前に、ホストまたは AS/400 データベース・サーバー上とワークステーション上で通信を構成しておく必要があります。詳細については、インストールおよび構成 補足 を参照してください。
2. OS/2 と Windows 32 ビット オペレーティング・システムの場合、データベース・ディレクトリーは DB2 ユニバーサル・データベースのクライアント構成アシスタント (CCA) を使って更新することができます。  
その他のプラットフォームでは、データベース・ディレクトリーの更新は DB2 コマンド行プロセッサ (CLP) を使って行います。
3. 97ページの『ディレクトリーの更新』には、コマンド構文の例があります。詳細については、コマンド解説書 を参照してください。
4. DCE を使用している場合、接続先のホストまたは AS/400 データベース・サーバー・データベースごとに、これらのディレクトリーを更新するか、同等の情報をグローバル DCE ディレクトリーに保管する必要があります。DCE の詳細については、223ページの『付録D. DCE ディレクトリー・サービスの使用』および管理の手引き を参照してください。ここでは、DCE ディレクトリー・サービスを利用していない ことを前提にしています。

---

## 情報の収集

217ページの『付録B. ディレクトリー・カスタマイズ・ワークシート』には、収集する必要がある情報が示されています。そのワークシートのコピーをとって、ご使用のシステム値を記入しておくとう便利です。

### ノード・ディレクトリー

ノード・ディレクトリーには、次の情報を指定することができます。

#### ノード名

リモート・データベースがあるホストまたは AS/400 データベース・サーバー・システムのニックネーム。この名前は、ユーザー定義です。同じノード名を「ノード・ディレクトリー・パラメーター」表および「システム・データベース・ディレクトリー・パラメーター」表の両方に記入します。

形式は、1～8桁の単一バイト英数字です。(番号記号(#)、アットマーク(@)、ドル記号(\$)、および下線(\_))が含まれます。)下線または数字で始めることはできません。

#### プロトコル

APPC または TCPIP を指定できます。

#### 記号宛先名

APPC ノードを定義する場合、「CPI 通信サイド情報」表に指定されている記号宛先名を使用します(たとえば、Microsoft の SNA サーバーを使用する場合は、CPI-C 記号宛先プロパティの名前)。この値は、SNA をインストールまたは構成(あるいはこれら両方)を行った人から入手する必要があります。記号宛先名には大文字小文字の区別があります(大文字と小文字のミスマッチがあると、SQL1338 戻りコードが戻されることがあります)。

#### 機密保護タイプ

実行される機密保護検査のタイプ。APPC ノードの場合、有効なオプションは SAME、PROGRAM、および NONE です。TCP/IP ノードでは、SECURITY SOCKS は、ノードで SOCKS を使用できる状態にするように指定するオプションです。この場合、SOCKS を使用可能にするには、SOCKS\_NS および SOCKS\_SERVER 環境変数を設定しなければなりません。詳細については、125ページの『第10章 機密保護』およびコマンド解説書を参照してください。

#### TCP/IP リモート・ホスト名または IP アドレス

TCP/IP ノードを定義している場合、リモート TCP/IP ホスト名またはリモート TCP/IP アドレス。ホスト名が指定される場合、DB2 コネク



ト・ワークステーションで、ドメイン・ネーム・サーバー (DNS) の検索を介して、またはローカル TCP/IP ホスト・ファイル内の項目によって解決されなければなりません。

DB2 (OS/390 版) リモート・ホストの場合、ホスト名は、分散データ機能 (DDF) の起動時に DSNL004I メッセージ (DOMAIN=hostname) に示されます。

### **TCP/IP サービス名またはポート番号**

TCP/IP ノードを定義している場合、リモート TCP/IP サービス名またはポート番号。これは、リモート・ホストで TCP/IP に定義しなければなりません。ポート番号 446 を DRDA 用の省略時ポート番号として登録しておく必要があります。

DB2 (OS/390 版) リモート・ホストの場合、ポート番号はブートストラップ・データ・セット (BSDS) で定義され、分散データ機能 (DDF) の起動時に出される DSNL004I メッセージ (TCPPORT=portnumber) で提供されます。

**注:** TCP/IP 接続での 2 フェーズ・コミット再同期操作で用いられる 2 番目のポートは、サーバーにより割り当てられます。たとえば、DB2 ユニバーサル・データベース (OS/390 版) ブートストラップ・データ・セットの場合、DB2 ユニバーサル・データベース (OS/390 版) へのインバウンド接続の再同期でのみ使用されるポート番号 (RESPORT) が割り当てられます。サービス名を定義する必要はありません。

## **DCS ディレクトリー**

DCS ディレクトリー内では、以下の情報を指定することができます。

### **データベース名**

ホストまたは AS/400 データベース・サーバーのユーザー定義のニックネーム。同じデータベース名を「DCS ディレクトリー・パラメーター」表および「システム・データベース・ディレクトリー・パラメーター」表の両方に記入します。

形式は、1~8 桁の単一バイト英数字です。(番号記号 (#)、アットマーク (@)、ドル記号 (\$)、および下線 (\_) が含まれます。) 下線または数字で始めることはできません。

### **宛先データベース名**

ホストまたは AS/400 データベース・サーバー・システム上のデータベース。以下のとおりです。

## MVS/ESA

DB2 ユニバーサル・データベース (OS/390 版) サブシステム。その LOCATION NAME によって識別される。

LOCATION NAME は、TSO にログインした後で、利用可能ないずれかの照会ツールを使って以下に示す SQL 照会を発行することで判別できます。

```
select current server from sysibm.sysdummy1
```

さらに、LOCATION NAME は MVS/ESA ブートストラップ・データ・セット (BSDS) でも定義されており、DSNL004I メッセージ (LOCATION=location) に表示されます。これは、分散データ機能 (DDF) の起動時に書き込まれます。

## OS/390

DB2 ユニバーサル・データベース (OS/390 版) サブシステム。その LOCATION NAME によって識別される。

LOCATION NAME は、TSO にログインした後で、利用可能ないずれかの照会ツールを使って以下に示す SQL 照会を発行することで判別できます。

```
select current server from sysibm.sysdummy1
```

さらに、LOCATION NAME はブートストラップ・データ・セット (BSDS) でも定義されており、DSNL004I メッセージ (LOCATION=location) に表示されます。これは、分散データ機能 (DDF) の起動時に書き込まれます。

## VSE または VM

データベース名 (DBNAME)

## OS/400

関係データベース名 (RDBNAME)

その他 OS/2、Windows NT、Windows 2000、および UNIX ベースのシステムの場合は、データベース・ディレクトリーにあるデータベース別名。

## アプリケーション・リクエスター名

SQL 要求を DRDA アプリケーション・サーバーへ送るアプリケーション・リクエスター・ライブラリーの名前。このアプリケーション・リクエスターは、アプリケーション・プログラムに代わって要求を処理します。

書式: AR <application\_requester\_name>

省略時値は、DB2 コネクトのアプリケーション・リクエスターです。

### パラメーター・ストリング

省略時値を変更したいときは、次のパラメーターの一部または全部を次の順序で指定します。パラメーター・ストリングはクライアント構成アシスタントを使って設定することができません。また、CLP の使用時には、パラメーター・ストリングを単一引用符 (OS/2 や Windows NT などの場合)、あるいは二重引用符 (AIX などの場合) で囲まなければなりません。

#### マッピング・ファイル

省略時の SQLCODE マッピングを上書きする SQLCODE マッピング・ファイルの名前。SQLCODE マッピングをオフにするときは、**NOMAP** を指定します。詳細については、137ページの『第11章 SQLCODE マッピング』を参照してください。

**,D** これは 2 番目の定位置パラメーターです。このパラメーターを指定すると、以下の SQLCODES のいずれかが戻されるときにアプリケーションはホストまたは AS/400 データベース・サーバー・データベースから切断します。

SQL30000N  
SQL30040N  
SQL30050N  
SQL30051N  
SQL30053N  
SQL30060N  
SQL30070N  
SQL30071N  
SQL30072N  
SQL30073N  
SQL30074N  
SQL30090N

**,D** を指定してもしなくても、次の SQLCODE の場合は切断を起こします。

SQL30020N  
SQL30021N  
SQL30041N  
SQL30061N  
SQL30081N

これらのコードの説明については、メッセージ解説書を参照してください。

注: エラーのために DB2 コネク트가切断すると、自動的にロールバックされます。

### **„,INTERRUPT\_ENABLED**

これは 3 番目の定位置パラメーターです。DB2 コネク트가ワークステーションの DCS ディレクトリーで INTERRUPT\_ENABLED が構成されている場合、ホストまたは AS/400 データベース・サーバーに接続している状態でクライアント・アプリケーションが割り込みを発行すると、DB2 コネク트는、接続を除去し、作業単位をロールバックすることによって割り込みを実行します。この割り込み動作は、AIX、OS/2、Windows NT、および Windows 2000 上でサポートされます。

そのアプリケーションは、サーバーへの接続が終了したことを示す sqlcode (-30081) を受け取ります。アプリケーションが追加のデータベース要求を処理するには、その後でホストまたは AS/400 データベース・サーバーへの新しい接続を確立する必要があります。

AIX V4.1 以上、SNA Server V3.1 以上、OS/2、Windows NT、および Windows 2000 以外のプラットフォームでは、アプリケーションが割り込み要求を受信したときの、DB2 コネク��による自動切断オプションはサポートされていません。

注: このサポートは、任意のプラットフォームでの TCP/IP 接続に対して機能します。クライアントがソケットを kill する場合がありますが、サーバー処理系によって未解決の受信になるかどうかが決まります。DB2 ユニバーサル・データベース (OS/390 版) は非同期のソケット呼び出しを利用するので、接続の切断を検出し、進行中の長時間実行 SQL ステートメントをロールバックすることができます。

### **„„„,SYSPLEX**

これは 6 番目の定位置パラメーターで、特定のデータベースに対して DB2 コネク�� SYSPLEX サポートを明示的に使用可能にするときに使用します。

ワークステーション・レベルで SYSPLEX サポートを使用不可にする、新しいプロファイル (環境またはレジストリー) 変数 DB2SYSPLEX\_SERVER が用意されています。

**,,,,,LOCALDATE=<value>**

これは 7 番目の定位置パラメーターで、DB2 コネクトの日付形式サポートを使用可能にするときに使用します。これは、<value> に対する日付マスクにより実装されます。以下のとおりです。

たとえば、次の CLP (コマンド行プロセッサ) ステートメントを発行したとします。

```
catalog appc node nynode remote nycpic security program
catalog dcs database nydb1 as new_york
catalog database nydb1 as newyork1 at node nynode
authentication dcs
```

ホスト・データベースにアクセスするためにデータベース別名 *newyork1* を使用します。日付マスクは指定されていないので、データ変換は行われません。

ここで新しい日付形式サポートを使用すると、以下の CLP コマンドを発行することができます。この場合、CLP が使用されており、パラメーター・ストリング自体が二重引用符を使って指定されているため、LOCALDATE 値は 2 組の二重引用符に囲まれています。また、LOCALDATE 指定から二重引用符が外されないようオペレーティング・システムのエスケープ文字 *¥* (円記号) が使用されていることにも注目してください。95ページの『パラメーター・ストリングの指定』も参照してください。

```
catalog dcs database nydb2 as new_york
parms ¥",,,,,,LOCALDATE=¥"¥"YYYYMMDD¥"¥"¥"
catalog database nydb2 as newyork2 at node nynode
authentication dcs
```

データベース別名 *"newyork2"* は同じホスト・データベースへのアクセスを提供しますが、日付形式マスクも指定します。この例は、日付形式マスクが LOCALDATE キーワードを使って指定されること、また日付形式マスクが DCS ディレクトリー・エントリーの PARMS フィールドの中で 7 番目の定位置パラメーターであることを示しています。

日付マスクを有効にするには、以下の条件をすべて満たしていなければなりません。

1. Y、M、および D の数列はそれぞれ 1 回しか認められていません。ここで Y は年、M は月、また D は日を表す数字です。
  2. 1 つの数列における Y の最大数は 4 個です。
  3. 1 つの数列における M の最大数は 2 個です。
  4. 1 つの数列における D の最大数は 2 個です。
- たとえば、以下の日付マスクはすべて有効です。

```
"YYyyMmDd" - Y, M, and D digits are case-insensitive  
"MM+DD+YYYY" - OK to have a mask longer than 10 bytes  
                and to have characters other than Y, M,  
                and D in the mask  
"abcYY+MM" - OK not to have a sequence of D's
```

以下の日付マスクはすべて無効です。

```
"YYYYMMDD" - invalid there are 5 Y's in a sequence  
"YYYYMDDM" - invalid there are 2 sequences of M's
```

日付形式マスクが無効でもエラーは出されません。マスクが無視されるだけです。日付マスクが有効であっても、そのマスクが必ず使用されるとは限りません。有効な日付マスクに基づく日付形式変換は、以下の条件をすべて満たしている場合にのみ実行されます。

1. SQL エラーがない。
2. 出力が、ISO 準拠形式 (ISO および JIS) の日付値である。
3. 出力データ域の長さが最大 10 バイトある。この長さは、データ形式変換を実行しない場合でも、データ値を出力データ域に保管するのに最低必要なサイズです。この要件は、日付形式マスクが 10 バイトより短いときにも当てはまります。
4. DCS ディレクトリー・エントリーで有効な日付形式マスクが指定されており、そのマスクが出力データ領域に保管可能である。

**,,,,,,CHGPWD\_SDN=<name>**

これは 8 番目の定位置パラメーターで、パスワード有効期限管理 (PEM) で使用する記号宛先名を指定します。<name> に指定される値では、大文字小文字が区別されます。

131ページの『MVS パスワードの変更』には、次のように CHGPWD\_SDN を使って DCS データベース・ディレクトリーをカタログ化する例が収められています。

```
catalog dcs database db1 as dsn_db_1 parms  
",,,,,,,,,CHGPWD_SDN=pempgm"
```

**,,,,,,,,,BIDI=<ccsid>**

これは 9 番目の定位置パラメーターで、省略時のサーバー・データベース BiDi CCSID の上書きに使用する両方向 (BiDi) CCSID を指定します。以下に例を示します。

```
",,,,,,,,,BIDI=xyz"
```

xyz は上書きする CCSID を表しています (95ページの 1 を参照)。

サポートされている BiDi CCSID のリスト、およびそれらのストリング・タイプについては、[管理の手引き](#) を参照してください。

BiDi データをさまざまなプラットフォームで正しく処理するには、以下の BiDi 属性が必要です。

- 数字の形状 (ARABIC または HINDI)
- 方向 (RIGHT-TO-LEFT または LEFT-TO-RIGHT)
- 形状 (SHAPED または UNSHAPED)
- 対称性 (YES または NO)
- テキストの形式 (LOGICAL または VISUAL)

プラットフォームによって省略時値が異なるため、DB2 データをあるプラットフォームから別のプラットフォームに送信するときに問題が生じます。たとえば、Windows プラットフォームでは LOGICAL UNSHAPED データを使用しますが、MVS および OS/390 では通常 SHAPED VISUAL データ形式を使用します。そのため、BiDi 属性のサポートがないと、MVS 版または OS/390 版の DB2 から Windows 版の DB2 コネクトに送信されるデータは正しく表示されません。

DB2 コネクトとサーバー上のデータベースとの間でデータを交換する場合、通常、着信データの変換を実行するのは受信側です。普通は、同じ変換が、通常のコード・ページ変換の拡張である、BiDi のレイアウト変換にも適用されます。ただし、現在のところ BiDi 特有の CCSID または BiDi レイアウト変換をサポートしているホスト DB2 製品はありません。そのため、DB2 コネクト

では、サーバー・データベースから受信したデータに加えて、サーバー・データベースに送信するデータに対しても BiDi レイアウト変換を実行するオプション機能が追加されました。

サーバー・データベースに送信するデータに対して DB2 コネクトが BiDi レイアウト変換を実行するには、サーバー・データベースの BiDi CCSID を上書きする必要があります (95 ページの 2 を参照)。このことは、サーバー・データベースに対応する DCS データベース・ディレクトリー・エントリーの PARMS フィールドで、BIDI パラメーターを使って行います。

この機能については、例を使って説明します。

CCSID 62213 (BiDi ストリング・タイプ 5) を実行しているヘブライ語の DB2 クライアントが、CCSID 424 (BiDi ストリング・タイプ 4) を実行している DB2 ホスト・データベースにアクセスするとします。ただし、DB2 ホスト・データベースに格納されているデータは CCSID 8616 (BiDi ストリング・タイプ 6) であることが分かっています。

この場合、2 つの問題が生じます。まず、CCSID 424 と 8616 の間で BiDi ストリング・タイプがどのように異なるかを DB2 ホスト・データベースは理解していません。次に、DB2 ホスト・データベースは 62213 という DB2 クライアント CCSID を識別できません。DB2 ホスト・データベースがサポートするのは、CCSID 62213 と同じコード・ページに基づく CCSID 862 だけです。

DB2 ホスト・データベースに送信されるデータが BiDi ストリング・タイプ 6 形式になっていることを確認する必要があります。また、DB2 ホスト・データベースから受信するデータに対して BiDi レイアウト変換を実行する必要があることを DB2 コネクトに知らせなければなりません。DB2 ホスト・データベースに対しては次のカタログを使用します。

```
catalog dcs database nydb1 as TELAVIV parms ",,,,,,,BIDI=8616"
```

ここでは、DB2 ホスト・データベース CCSID 424 を 8616 に上書きするよう DB2 コネクトに指示します。この上書きには以下の処理が関係しています。

1. DB2 コネクトが CCSID 862 を使って DB2 ホスト・データベースに接続する。
2. DB2 ホスト・データベースに送信するデータに対して、DB2 コネクトが、CCSID 62213 (BiDi ストリング・タイプ 5) から CCSID 62221 (BiDi ストリング・タイプ 6) への BiDi レイアウト変換を実行する。



3. DB2 ホスト・データベースから受信したデータに対して、DB2 コネクトが、CCSID 8616 (BiDi ストリング・タイプ 6) から CCSID 62213 (BiDi ストリング・タイプ 5) への BiDi レイアウト変換を実行する。

**注:**

1. BIDI パラメーターを有効にするには、環境変数またはレジストリー値 DB2BIDI を YES に設定しなければなりません。
2. DB2 ホスト・データに送信するデータに対して DB2 コネクトがレイアウト変換を実行するようにするには、たとえ CCSID を上書きする必要がない場合でも、DCS データベース・ディレクトリーの PARMS フィールドに BIDI パラメーターを追加しなければなりません。その場合、CCSID には省略時の DB2 ホスト・データベース CCSID を提供してください。
3. 状況によっては、両方向の CCSID を使うことにより SQL 照会自体が変更され、DB2 サーバーで認識されなくなることがあります。特に、他のストリング・タイプが使用可能な場合、IMPLICIT CONTEXTUAL および IMPLICIT RIGHT-TO-LEFT CCSID の使用は避けてください。SQL 照会に引用符付き文字列が含まれている場合、CONTEXTUAL CCSID を使用すると予期せぬ結果を招くことがあります。SQL ステートメントでは引用符付き文字列の使用を避け、可能であれば代わりにホスト変数を使用してください。

特定の両方向 CCSID を使用したときにこれらの推奨処置では修正できない問題が生じる場合、環境変数またはレジストリー値 DB2BIDI を NO に設定する必要があります。

### パラメーター・ストリングの指定

指定できるパラメーター・ストリングの例をいくつか示します。

たとえば、以下のいずれかを指定することができます。“¥” (円記号) はオペレーティング・システムのエスケープ文字を表します。

AIX の場合:

```
NOMAP
/u/username/sql1lib/map/dcs1new.map,D
,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE=¥"¥"YYMMDD¥"¥",,
```

OS/2、Windows NT、Windows 2000 の場合:

```
NOMAP
d:¥sql1lib¥map¥dcs1new.map,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE=¥"¥"YYMMDD¥"¥",,
```

別の方法として、パラメーター・ストリングを指定せずに省略時値を受け入れることもできます。

**注:** パラメーター・ストリングに LOCALDATE マスクを指定するときは 2 組の二重引用符を指定する必要があるため、オペレーティング・システムのエスケープ文字 “¥” (円記号) を使用しなければなりません。例を示します。

```
db2 catalog dcs db x as y parms ¥",,,,,,LOCALDATE=¥"¥"YYMMDD¥"¥"¥"
```

結果として、次の DCS ディレクトリー・エントリーになります。

```
DCS 1 entry:
Local database name           = X
Target database name         = Y
Application requestor name   =
DCS parameters               = ,,,,,,LOCALDATE="YYMMDD"
Comment                      =
DCS directory release level  = 0x0100
```

## システム・データベース・ディレクトリー

システム・データベース・ディレクトリーには、次の情報を指定することができます。

### データベース名

「DCS ディレクトリー・パラメーター (DCS Directory Parameter)」表に記入したのと同じ値。

### データベース別名

ホストまたは AS/400 データベース・サーバーの別名。この名前は、データベースにアクセスするアプリケーション・プログラムによって使用されます。省略時設定では、データベース名について指定した値が使われます。

形式は、1~8 桁の単一バイト英数字です。(番号記号 (#)、アットマーク (@)、ドル記号 (\$)、および下線 (\_) が含まれます。) 下線または数字で始めることはできません。

### ノード名

「ノード・ディレクトリー・パラメーター (Node Directory Parameter)」表に記入したのと同じ値。

**認証** ユーザー名およびパスワードの検証をどこで行うかを指定します。有効なオプションは、SERVER、SERVER\_ENCRYPT、CLIENT、DCE、DCS、および DCS\_ENCRYPT です。詳細については、125ページの『第10章 機密保護』を参照してください。

## 同一のデータベースへの複数項目の定義

各データベースについて、3つのディレクトリー（ノード・ディレクトリー、DCS ディレクトリー、およびシステム・データベース・ディレクトリー）のおのおのに少なくとも1つの項目を定義する必要があります。場合によっては、データベースに複数の項目を定義したいこともあります。

たとえば、ホストまたは AS/400 データベース・サーバーから移植されたアプリケーションについては SQLCODE マッピングをオフにして、クライアント / サーバー環境用に開発されたアプリケーションについては省略時のマッピングを受け入れることができます。これは次のように行います。

- ノード・ディレクトリー内に1つの項目を定義します。
- DCS ディレクトリー内に、異なるデータベース名を持つ2つの項目を定義します。一方の項目には、パラメーター・STRINGに NOMAP を指定します。
- システム・データベース・ディレクトリーに、異なるデータベース別名を持つ2つの項目を定義します。

両方の別名とも同じデータベースにアクセスし、一方は SQLCODE マッピングを用いて、他方は SQLCODE マッピングなしです。

---

## ディレクトリーの更新

CATALOG コマンドは DB2 コネクト・システムであればどれも、また CCA の「データベースの追加 (Add Database)」ウィザードは OS/2 および Windows 32 ビット オペレーティング・システムで使用できます。DB2 アプリケーション開発クライアントをお持ちの場合は、アプリケーション・プログラムを作成して項目をカタログすることもできます。API の詳細については、*管理 API 解説書* および *コマンド解説書* を参照してください。

**注:** データベースをカタログ作成するには、*sysadm* または *sysctrl* 権限を持っていなければなりません。

コマンド行プロセッサを使用してディレクトリーを更新するには、次のようにします。

1. 次のコマンドを使用して、ノード・ディレクトリーを更新します。
- APPC 接続のあるノードの場合、

```
db2 CATALOG APPC NODE nodename  
REMOTE symbolic_destination_name SECURITY security_type
```

以下に例を示します。

```
db2 CATALOG APPC NODE DB2NODE REMOTE DB2CPIC SECURITY PROGRAM
```

- TCP/IP 接続を利用している DB2 ユニバーサル・データベース (OS/390 版) V5.1 または DB2 ユニバーサル・データベース (AS/400 版) V4.2 の場合、

```
db2 CATALOG TCPIP NODE nodename  
REMOTE hostname または IP address  
SERVER service_name または port_number  
SECURITY security_type
```

以下に例を示します。

```
db2 CATALOG TCPIP NODE MVSIPNOD REMOTE MVSHOST SERVER DB2INSTC
```

TCP/IP 接続の省略時の DRDA ポート番号は 446 です。

2. 次のコマンドを使用して、DCS ディレクトリーを更新します。

```
db2 CATALOG DCS DATABASE database_name AS target_database_name  
[AR application_requester]  
[PARMS "parameter string"]
```

以下に例を示します。

```
db2 CATALOG DCS DATABASE DB2DB AS NEW_YORK3
```

OS/2、Windows NT、Windows 2000 の場合、

```
db2 CATALOG DCS DATABASE DB2DB AS NEW_YORK3 PARMS "NOMAP,D"
```

AIX の場合、

```
db2 CATALOG DCS DATABASE DB2DB AS NEW_YORK3 PARMS '"NOMAP,D"'
```

**注:** パラメーター・ストリングに LOCALDATE マスクを指定するときにはオペレーティング・システムのエスケープ文字を使用する方法については、95ページの『パラメーター・ストリングの指定』を参照してください。

3. 次のコマンドを使用して、システム・データベース・ディレクトリーを更新します。

```
db2 CATALOG DATABASE database_name  
AS alias AT NODE nodename  
AUTHENTICATION authentication_type
```

以下に例を示します。

```
db2 CATALOG DATABASE DB2DB AS NYC3 AT NODE DB2NODE AUTHENTICATION DCS
```

リモート・クライアントがある場合は、各リモート・クライアントについてもディレクトリーを更新する必要があります。詳細については、該当する *DB2 コネクト 概説* および *インストール* を参照してください。



---

## 第7章 アプリケーションとユーティリティーのバインド

組み込み SQL を使用して開発されたアプリケーション・プログラムは、それが作動するとき使用する各データベースとバインドされていなければなりません。 コマンド・センターおよびクライアント構成アシスタントが使用可能なプラットフォームでは、これらの機能を使用してこれを行うことができます。

バインドは、各データベースにつき、アプリケーションごとに 1 度実行されなければなりません。 バインド・プロセスの間に、データベース・アクセス・プランは、実行される各 SQL ステートメントごとに保管されます。 これらのアクセス・プランは、アプリケーション開発者によって提供され、(プリコンパイルの間に作成された) バインド・ファイル に入っています。 バインドとは、ホストまたは AS/400 データベース・サーバーによりそれらのバインド・ファイルを処理するプロセスを指します。 バインドの詳細については、アプリケーション開発の手引き を参照してください。

DB2 コネクトで提供されているいくつかのユーティリティーは組み込み SQL を使って開発されているため、ホストまたは AS/400 データベース・サーバーにバインドするまでそれらをシステムで使用することはできません。 105ページの表4 でリストされている DB2 コネクト・ユーティリティーおよびインターフェースを使用しない場合、それらを各ホストまたは AS/400 データベース・サーバーにバインドする必要はありません。 これらのユーティリティーに必要なバインド・ファイルのリストは、次のファイルに含まれています。

### **ddcsmvs.lst**

MVS または OS/390 用

### **ddcsvse.lst**

VSE 用

### **ddcsvm.lst**

VM 用

### **ddcs400.lst**

OS/400 用

これらのファイルの 1 つをデータベースにバインドすることは、個々のユーティリティーをそのデータベースへバインドすることになります。

DB2 コネクト エンタープライズ・エディション がインストールされている場合は、それぞれのクライアント・プラットフォーム・タイプから DB2 コネクト・ユーティリティーを各ホストまたは AS/400 データベース・サーバーにバインドしないと、それらをシステムで使用することはできません。

たとえば、10 の OS/2 クライアントと、10 の Windows クライアントと、そして 10 の AIX クライアントがそれぞれ、DB2 コネクト エンタープライズ・エディション (Windows NT 版) サーバー経由で DB2 ユニバーサル・データベース (OS/390 版) に接続している場合は、次のようにします。

1. ddcsmvs.lst を Windows クライアントの 1 つからバインドします。
2. ddcsmvs.lst を OS/2 クライアントの 1 つからバインドします。
3. ddcsmvs.lst を AIX クライアントの 1 つからバインドします。
4. ddcsmvs.lst を DB2 コネクト・サーバーからバインドします。

**注:** これは、すべてのクライアントが同じサービス・レベルにあることを前提としています。そうでない場合、特定のサービス・レベルの各クライアントからバインドする必要があります。DB2 バージョン 2.1 以前のクライアントがある場合は、235ページの『付録E. バック・レベル・クライアント用ユーティリティーのバインド』を参照してください。

DB2 コネクト・ユーティリティーに加えて、組み込み SQL を使用するその他のアプリケーションについても、その作業のときに使用する各データベースへバインドする必要があります。バインドされていないアプリケーションは、実行する時、通常は SQL0805N エラー・メッセージを出します。バインドを必要とするすべてのアプリケーションについて追加的なバインド・リスト・ファイルを作成したいときがあるかもしれません。

バインド先のホストまたは AS/400 データベース・サーバーごとに、以下の手順を実行します。

1. ホストまたは AS/400 データベース・サーバー管理システムに対して十分な権限があることを確認します。

#### **MVS または OS/390**

必要な許可は以下のとおりです。

- SYSADM または
- SYSCTRL または
- BINDADD および CREATE IN COLLECTION NULLID



**注:** BINDADD および CREATE IN COLLECTION NULLID 特権は、パッケージがまだない場合、たとえば、はじめてパッケージを作成するときの**のみ**十分な権限であるといえます。

パッケージがすでに存在しており、それらを再度バインドしている場合は、タスクを完成させるために必要な権限は、誰がオリジナル・バインドを行ったかにかかってきます。

**A** もし同一人がオリジナル・バインドを行い、再度のバインドも行っている場合は、上に挙げた権限のどれでも有することになり、バインドを完了させることができます。

**B** 最初のバインドを行ったのとは違う人が、2番目のバインドを行う場合は、バインドを完了するのに SYSADM または SYSCTRL 権限が必要です。BINDADD および CREATE IN COLLECTION NULLID 権限を有するだけでは、バインドを完了させることができません。ただし、SYSADM または SYSCTRL 権限のいずれかがあれば、パッケージの作成は可能です。この状況では、置き換えようとする既存のパッケージのおおのについて、BIND 特権が必要になります。

## VSE または VM

DBA 権限が必要です。バインド・コマンドについて GRANT オプションを使用しようとする場合 (各 DB2 コネクト・パッケージに対するアクセス権を個別に付与することを避けるために)、NULLID ユーザー ID は、以下の表についての権限を他のユーザーに付与する権限を持たなければなりません。

- system.syscatalog
- system.syscolumns
- system.sysindexes
- system.systabauth
- system.syskeycols
- system.syssynonyms
- system.syskeys
- system.syscolauth

VSE または VM システムでは、次の命令を発行することができます。

```
grant select on table to nullid with grant option
```

## OS/400

NULLID コレクションにある \*CHANGE 権限またはそれ以上の権限。

2. 以下と同様のコマンドを発行します。

```
db2 connect to DBALIAS user USERID using PASSWORD
db2 bind path@ddcsmvs.lst blocking all
      sqlerror continue messages ddcsmvs.msg grant public
db2 connect reset
```

ここで、*DBALIAS*、*USERID*、および *PASSWORD* はホストまたは AS/400 のデータベース・サーバーに適用されます。 *ddcsmvs.lst* は MVS 用のバインド・リスト・ファイル、*path* はバインド・リスト・ファイルのロケーションです。

たとえば、*drive*:%sqllib%*bnd*% はすべての Intel オペレーティング・システムに、*INSTHOME*/sqllib/*bnd*/ はすべての UNIX オペレーティング・システムに適用されます。ここで、*drive* は DB2 コネクトがインストールされている論理ドライブ、*INSTHOME* は DB2 コネクト・インスタンスのホーム・ディレクトリーです。

**bind** コマンドの *grant* オプションを使用することにより、PUBLIC に対し、または特定のユーザー名またはグループ ID に対し、EXECUTE 特権を付与することができます。 **bind** コマンドの *grant* オプションを使用しない場合は、GRANT EXECUTE (RUN) を個別に与えなければなりません。

バインド・ファイルについてのパッケージ名を見つけるには、次のコマンドを入力します。

```
ddcspkgn @bindfile.lst
```

以下に例を示します。

```
ddcspkgn @ddcsmvs.lst
```

出力は次のようになります。

Bind File	Package Name
f:%sqllib% <i>bnd</i> %db2ajgrt.bnd	SQLAB6D3

参照のために、表4 には、DB2 コネクトの各種構成要素で使用されるバインド・ファイルおよびパッケージ名が示されています。ある場合には、異なるオペレーティング・システムでは異なるバインド・ファイルおよびパッケージが使用されています。

表4. バインド・ファイルおよびパッケージ

構成要素	バインド・ファイル	パッケージ	MVS または OS/390	VSE	VM	OS/400
バインダー (GRANT バインド・オプションで使用される)	db2ajgrt.bnd	sqlabxxx	あり	あり	あり	あり
<b>DB2 コール・レベル・インターフェース</b>						
分離レベル CS	db2clics.bnd	sql11xxx	あり	あり	あり	あり
分離レベル RR	db2clirr.bnd	sql12xxx	あり	あり	あり	あり
分離レベル UR	db2cliur.bnd	sql13xxx	あり	あり	あり	あり
分離レベル RS	db2clirs.bnd	sql14xxx	あり	あり	あり	あり
分離レベル NC	db2clinc.bnd	sql15xxx	なし	なし	なし	あり
MVS 表名の使用	db2clims.bnd	sql17xxx	あり	なし	なし	なし
OS/400 表名の使用 (OS/400 3.1 またはそれ以降)	db2clias.bnd	sql1axxx	なし	なし	なし	あり
VSE/VM 表名の使用	db2clivm.bnd	sql18xxx	なし	あり	あり	なし
<b>コマンド行プロセッサ</b>						
分離レベル CS	db2clpcs.bnd	sqlc2xxx	あり	あり	あり	あり
分離レベル RR	db2clpr.rnd	sqlc3xxx	あり	あり	あり	あり
分離レベル UR	db2clpur.bnd	sqlc4xxx	あり	あり	あり	あり
分離レベル RS	db2clprs.bnd	sqlc5xxx	あり	あり	あり	あり
分離レベル NC	db2clpnc.bnd	sqlc6xxx	なし	なし	なし	あり
<b>REXX</b>						
分離レベル CS	db2arxcs.bnd	sqla1xxx	あり	あり	あり	あり
分離レベル RR	db2arxrr.bnd	sqla2xxx	あり	あり	あり	あり
分離レベル UR	db2arxur.bnd	sqla3xxx	あり	あり	あり	あり
分離レベル RS	db2arxrs.bnd	sqla4xxx	あり	あり	あり	あり
分離レベル NC	db2arxnc.bnd	sqla5xxx	なし	なし	なし	あり
<b>ユーティリティ</b>						
エクスポート	db2uexpm.bnd	sqlubxxx	あり	あり	あり	あり
インポート	db2uimpb.bnd	sqlufxxx	あり	あり	あり	あり

DB2 コネクトの場合にこれらの値を判別するには、次のように *ddcspkgn* ユーティリティを実行します。

```
ddcspkgn @ddcsmvs.lst
```

任意に、このユーティリティを使用して、個々のバインド・ファイルのパッケージ名を判別することができます。たとえば、次のようにします。

```
ddcspkgn bindfile.bnd
```

DB2 (MVS/ESA 版) システムが APAR PN60988 をインストール済みである (またはそれがバージョン 3 リリース 1 以降のものである) 場合、分離レベル NC 用のバインド・ファイルを *ddcsmvs.lst* ファイルに追加することができます。

バインド・オプションについて詳しくは、*コマンド解説書* を参照してください。

**注:**

- a. バインド・オプション `sqlerror continue` は必ず使用しなければなりません。ただし、DB2 ツールやコマンド行プロセッサを使ってアプリケーションをバインドする場合、このオプションは自動的に指定されます。このオプションを指定すると、バインド・エラーが警告に変わります。それで、エラーのあるファイルをバインドしても、やはりパッケージを作成することになります。言い換えると、別の SQL 構文が無効であると特定のサーバー処理系がフラグを付ける場合でさえ、複数のサーバーに対して 1 つのバインド・ファイルを使用することができます。そのため、特定のホストまたは AS/400 データベース・サーバーに対して任意のリスト・ファイル *ddcsxxx.lst* をバインドする場合、警告が出されることがあります。たとえば、DB2 (VM 版) に対してバインドすると、DB2 (VM 版) はカーソルが "WITH HOLD" と宣言されることを許可しないため、非常に多くの警告メッセージが出されます。
  - b. DB2 コネクト経由で DB2 ユニバーサル・データベースに接続する場合、バインド・リスト *db2ubind.lst* を使用してください。`sqlerror continue` はホストまたは AS/400 データベース・サーバーに接続するときだけ有効なので指定しないでください。また、DB2 ユニバーサル・データベースに接続する場合は、DB2 コネクトではなく DB2 で提供される DB2 クライアントを使用するようにお勧めします。
3. 各アプリケーションまたはアプリケーションのリストをバインドするには、類似のステートメントを使用します。
  4. DB2 の以前のリリースからのリモート・クライアントがある場合には、これらのクライアント上にあるユーティリティを DB2 コネクトにバインド

する必要があります。詳細については、235ページの『付録E. バック・レベル・クライアント用ユーティリティのバインド』を参照してください。

---

## BIND コマンド

DB2 **bind** コマンドは、アプリケーションを特定のデータベースにバインドします。プリコンパイルとバインドを別個の操作で行う場合、バインドの際に指定したオプションは、プリコンパイル段階で指定したオプションを上書きします。

コマンド解説書 には、DB2 コネクト経由でホストまたは AS/400 データベース・サーバーにアプリケーションをバインドするときの、**BIND** コマンドの構文が記述されています。DRDA 特有の記述は必ず参照してください。

**注:** **BIND** コマンドのいくつかのパラメーターは、ホストまたは AS/400 データベース・サーバーでサポートされていない場合があります。詳細については、使用するホストまたは AS/400 データベース・サーバー RDBMS に付属の資料を参照してください。

---

## 再バインド

アプリケーションのバインド（およびホストまたは AS/400 データベース・サーバーでのパッケージの作成）が終わると、そのパッケージを再作成する必要があります。このことは、オリジナル・バインド・ファイルがなくても行えます。それにはコマンド行プロセッサ **REBIND PACKAGE** コマンドを使用するか、または対応する **API** を使用します。

このコマンドを使用する利点は、以下のとおりです。

- オリジナル・バインド・ファイルなしで再最適化して新規のパッケージ・セクションを構築することによりシステム変更の利点を取り入れることができます。
- 作動不能または無効にされたパッケージを再作成することができます。
- 移行によって無効にされたパッケージを再作成することができます。
- 暗黙の再バインドまたはバインドを使用するより、明示の再バインドを使用することによってパフォーマンスを改善することができます。
- 特性は変更することができます。たとえば、DB2 ユニバーサル・データベース (OS/390 版) の場合、テストや移行の目的のために非修飾表の修飾子を変更することができます。

プログラム、バインド・オプション、または所有者の情報を修正したい場合は、**BIND** コマンドを使用する必要があります。また、パッケージがデータベース内に存在しない場合、または (最初に検出されたエラーだけでなく) すべてのバインド・エラーを見たい場合も、**BIND** コマンドを使用する必要があります。

このコマンドを実行するには、ホストまたは AS/400 データベース・サーバーに必要な権限レベルが必要です。データベースに接続しない場合は、(接続特権があるときは) 省略時のデータベースに暗黙の接続がなされます。

コマンド行プロセッサ・コマンドの構文については、**コマンド解説書** で説明されています。

バインド・ファイルについてのパッケージ名を見つけるには、コマンド **ddcspkgn bindfile.bnd** を入力します。

---

## 第8章 データベース・システム・モニター

この章では、DB2 コネクト・ユーザー用の DB2 システム・モニター機能を要約します。DB2 コネクト バージョン 7 では、以下のシステム・モニター拡張機能が用意されています。

- スナップショット・モニター機能。システムのスナップショットからは、特定の時点における情報を知ることができます。スナップショットは、特定のオブジェクトまたはオブジェクト・グループのデータベース・マネージャーにおける現在の活動状態を撮った写真のようなものです。モニターから入手できる DCS データベース情報には、基本的に 5 種類のスナップショットがあります。
- LIST DCS APPLICATIONS CLP コマンドと同じ働きをするグラフィカル・ユーザー・インターフェース。LIST コマンドはスナップショットに比べて、より簡潔な読み取り結果を表示します。DB2 コマンド・センターでは LIST をグラフィカルに表示することができます。117ページの『LIST DCS APPLICATIONS EXTENDED』を参照してください。

データベース・システム・モニターの詳細については、システム・モニター 手引きおよび解説書 を参照してください。

---

### リモート・クライアント接続のモニター

DB2 コネクト エンタープライズ・エディションと共にデータベース・システム・モニターを使用すると、リモート・クライアントの接続をモニターすることができます。DB2 コネクト・サーバーにとってローカルであり、そのサーバー自体の上で稼働しているクライアントをモニターするには、次の環境変数を設定する必要があります。

```
db2set DB2CONNECT_IN_APP_PROCESS=NO
```

たとえば、ホストまたは AS/400 システムでエラーが発生したとき、システム管理者は、問題が DB2 コネクト・ワークステーションで起きたものかどうかを判別することができます。データベース・システム・モニターは、以下のものを相互に関連付けます。

- DRDA 相関トークン (CRRTKN) (無保護会話用)
- 作業論理単位識別子 (LUWID) (SNA 同期点管理プログラムによって保護される 2 フェーズの会話用)

- 作業単位 ID (UOWID) (DRDA-3 同期点管理プログラムによって保護される 2 フェーズの接続用 (TCP/IP 接続上などで使用される))
- DB2 コネクト接続識別子 (アプリケーション ID)

この情報は、どの DB2 コネクト接続が問題を起こしたかを示し、それによってシステム管理者は、DB2 コネクト接続を使用するその他のクライアントに影響を与えずに個別のクライアント・アプリケーションをそのシステムから強制切断することができます。

## DB2 コネクト用のモニター・スイッチをオンにする

システム・モニターは常時活動しています。しかし、GET SNAPSHOT 出力で詳細を得たい場合は、対応するモニター・スイッチをオンにする必要があります。DB2 コネクトに関係したモニター・スイッチは、STATEMENT (ステートメント・レベルの情報) と UOW (トランザクション・レベルの情報) です。

モニター・スイッチを変更するには、**db2 update monitor switches** コマンドを使用します。このコマンドの構文については、**コマンド解説書** を参照してください。以下に例を示します。これは、作業単位 (UOW) について DB2 システム・モニター統計を作成します。

```
db2 update monitor switches using uow on
```

## モニター・スイッチ状況のリスト

モニター・スイッチの状況をリストするには、**db2 get monitor switches** コマンドを発行してください。

---

## GET SNAPSHOT コマンドの使用

DB2 モニターは、有用なシステム情報を絶えず記録しています。GET SNAPSHOT コマンドを発行すれば、システム状況の要約情報をいつでも入手できます。モニター・スナップショットは、モニターしたいデータベース・マネージャーのインスタンスに対して SYSMAINT、SYSCTRL、SYSADM のいずれかの権限があればとることができます。

DCS 情報をモニターするのに役立つスナップショット・コマンドは 5 つあります。それらのコマンドは以下のとおりです。

- GET SNAPSHOT FOR ALL DCS DATABASES
- GET SNAPSHOT FOR ALL DCS APPLICATIONS
- GET SNAPSHOT FOR DCS APPLICATION ...
- GET SNAPSHOT FOR DCS DATABASE ON db\_alias



- GET SNAPSHOT FOR DCS APPLICATIONS ON db\_alias

スナップショット・コマンドは要求した領域に関する詳細なレポートを生成します。

たとえば、GET SNAPSHOT FOR DCS DATABASE ON DCSDB を発行すると、以下に示すレポートが生成されます。

```

DCS Database Snapshot
DCS database name           = DCSDB
Host database name          = GILROY
First database connect timestamp = 12-15-1999 10:28:24.596495
Most recent elapsed time to connect = 0.950561
Most recent elapsed connection duration = 0.000000
Host response time (sec.ms) = 0.000000
Last reset timestamp        =
Number of SQL statements attempted = 2
Commit statements attempted   = 1
Rollback statements attempted = 0
Failed statement operations   = 0
Total number of gateway connections = 1
Current number of gateway connections = 1
Gateway conn. waiting for host reply = 0
Gateway conn. waiting for client request = 1
Gateway communication errors to host = 0
Timestamp of last communication error = None
High water mark for gateway connections = 1
Rows selected                = 0
Outbound bytes sent          = 140
Outbound bytes received     = 103

```

このレポートは、データベース接続、パフォーマンス、SQL 要求のエラーやスループットなどに関する情報を提供します。しかし実際には、DB2 モニター・スナップショットにはさらに多くの詳細情報が含まれています。たとえば、GET SNAPSHOT FOR ALL DCS APPLICATIONS コマンドを発行すると、以下のようなレポートが表示されます。

```

DCS Application Snapshot
Client application ID       = 09150F74.B6A4.991215152824
Sequence number            = 0001
Authorization ID           = SMITH
Application name           = db2bp
Application handle         = 1
Application status         = waiting for request
Status change time         = 12-15-1999 10:29:06.707086
Client node                 = sys143
Client release level       = SQL06010
Client platform            = AIX
Client protocol            = TCP/IP
Client codepage            = 850
Process ID of client application = 49074
Client login ID            = smith
Host application ID        = G9150F74.B6A5.991215152825

```

```

Sequence number = 0000
Database alias at the gateway = MVSDB
DCS database name = DCSDB
Host database name = GILROY
Host release level = DSN05012
Host CCSID = 500
Outbound communication address = 9.21.21.92 5021
Outbound communication protocol = TCP/IP
Inbound communication address = 9.21.15.116 46756
First database connect timestamp = 12-15-1999 10:28:24.596495
Host response time (sec.ms) = 0.000000
Time spent on gateway processing = 0.000000
Last reset timestamp =
Rows selected = 0
Number of SQL statements attempted = 2
Failed statement operations = 0
Commit statements = 1
Rollback statements = 0
Inbound bytes received = 404
Outbound bytes sent = 140
Outbound bytes received = 103
Inbound bytes sent = 287
Number of open cursors = 0
Application idle time = 1 minute and 32 seconds
UOW completion status =
Previous UOW completion timestamp = 12-15-1999 10:28:25.592631
UOW start timestamp = 12-15-1999 10:29:06.142790
UOW stop timestamp =
Elapsed time of last completed uow (sec.ms) = 0.034396
Most recent operation = Execute Immediate
Most recent operation start timestamp = 12-15-1999 10:29:06.142790
Most recent operation stop timestamp = 12-15-1999 10:29:06.707053
Statement = Execute Immediate
Section number = 203
Application creator = NULLID
Package name = SQLC2C07
SQL compiler cost estimate in timerons = 0
SQL compiler cardinality estimate = 0
Statement start timestamp = 12-15-1999 10:29:06.142790
Statement stop timestamp = 12-15-1999 10:29:06.707053
Host response time (sec.ms) = 1.101612
Elapsed time of last completed stmt(sec.ms) = 0.564263
Rows fetched = 0
Time spent on gateway processing = 0.013367
Inbound bytes received for statement = 220
Outbound bytes sent for statement = 130
Outbound bytes received for statement = 49
Inbound bytes sent for statement = 27
SQL statement text:
create table t12 (col1 int, col2 char)

```

GET SNAPSHOT コマンドとその他の役立つ DB2 モニター・コマンドについて詳しくは、システム・モニター 手引きおよび解説書 を参照してください。

---

## DCS アプリケーション状況のリスト

DB2 コネクト バージョン 5.2 の場合、システム・モニターは 3 つの書式で LIST DCS APPLICATIONS コマンドを提供しています。次のとおりです。

- LIST DCS APPLICATIONS
- LIST DCS APPLICATIONS SHOW DETAIL
- LIST DCS APPLICATIONS EXTENDED

### LIST DCS APPLICATIONS

アプリケーション・レベルでモニターが提供する情報を表示するには、DB2 LIST DCS APPLICATIONS コマンドを発行してください。APPC 接続について以下の情報が戻されます (DB2 コネクト エンタープライズ・エディション バージョン 7 から DB2 ユニバーサル・データベース (OS/390 版) への接続)。

Auth Id	Application Name	Appl. Handle	Host Application ID
USERID	db2bp_41	0	CAIBMOML.OMXT4H0A.A79EAA3C6E29

TCP/IP 接続について以下の情報が戻されます (DB2 コネクト エンタープライズ・エディション バージョン 7 から DB2 ユニバーサル・データベース (OS/390 版) への接続)。

Auth Id	Application Name	Appl. Handle	Host Application ID
USERID	db2bp_41	2	0915155C.9704.1517172201BE

### 許可 ID

ホストまたは AS/400 データベース・サーバーへのログオンに使用された許可 ID。これは、だれがアプリケーションを実行しているかを識別します。

### アプリケーション名

DB2 コネクトに認識されるとおりクライアントで実行中のアプリケーションの名前。最後のパス分離文字の後の最初の 20 バイトが利用可能です。アプリケーション名は、DB2 (OS/2 版) バージョン 1 上で実行するアプリケーションについては利用不能です。

### アプリケーション・ハンドル

DB2 コネクト・ワークステーション上で実行中のエージェント。この

要素を使用して、データベース・システム・モニターの情報をその他の診断情報とリンクさせることができます。(たとえば、186ページの『トレース・ユーティリティー (ddstrc)』をご覧ください。) エージェント ID は、FORCE USERS コマンドまたは API を使用するときも必要となります。

### ホスト・アプリケーション ID

次のいずれかです。

- DRDA 関連トークン (CRRTKN) (無保護会話用)
- 作業論理単位識別子 (LUWID) (SNA 同期点管理プログラムによって保護される 2 フェーズの会話用)
- 作業単位 ID (UOWID) (DRDA-3 同期点管理プログラムによって保護される 2 フェーズの接続用 (TCP/IP 接続上などで使用される))

この固有の識別子は、アプリケーションがホストまたは AS/400 データベース・サーバーに接続するときに生成されます。この要素をアプリケーション ID と一緒に使用することにより、アプリケーション情報のクライアント部分とサーバー部分を相関させることができます。

## LIST DCS APPLICATIONS SHOW DETAIL

DB2 LIST DCS APPLICATIONS SHOW DETAIL コマンド形式が指定される場合、以下の情報を含む追加情報が表示されます。

Auth Id	Application Name	Appl. Handle	Client Application Id	Seq#	Client DB Alias
NEWTON	db2bp	0	09151251.07D3.980925183850	0001	MVSDB
Client Node	Client Release Codepage	Host Application Id	Seq# Host DB Name		
antman	SQL05020 819	G9151251.67D4.980925183851	0000 GILROY		
Host Release					
DSN05011					

このレポートは不定様式なので、117ページの『LIST DCS APPLICATIONS EXTENDED』のレポートの方が便利な場合があります。

### クライアント・アプリケーション ID

DB2 コネクト・ワークステーションに接続されたアプリケーションを固有に識別します。アプリケーション ID については各種の形式があり、それらはクライアントと DB2 コネクト・ワークステーションとの間の通信プロトコルによって変わります。その形式の詳細については、[管理の手引き](#) を参照してください。

この値によって、クライアントから DB2 コネクト・ワークステーションへの接続、および DB2 コネクト・ワークステーションからホストまたは AS/400 データベース・サーバーへの接続を相互に関連させることができます。

#### クライアント順序番号 (Seq#)

クライアント順序番号は、トランザクションの順序番号です。クライアント順序番号は、異なるシステム上に広がるトランザクションを関連付けるのに役立ちます。

#### クライアント DB 別名

このデータベースの別名は、そのデータベースに接続するアプリケーションにより与えられます。この要素は、アプリケーションが実際にアクセスしているデータベースを識別するために使用できます。この名前とデータベース名との間のマッピングは、クライアント・ノードおよびデータベース・マネージャー・サーバー・ノードにあるデータベース・ディレクトリーを使用して行うことができます。

#### クライアント NNAME (ノード)

これはクライアント・アプリケーションが実行しているノードを識別します。この情報は、使用中のクライアント・プロトコルによって変わります。たとえば、NetBIOS 接続されたクライアントの場合、これは NNAME データベース・マネージャーの構成パラメーターの値です。TCP/IP 接続されたクライアントの場合、これはホスト名です。

#### クライアント製品 ID (クライアント)

クライアント上で実行している製品およびバージョン。クライアント製品 ID は以下のとおりになります。

- SQL01010 (DB2 (OS/2 版) バージョン 1 用)
- SQL01011 (UNIX ベースの DB2 製品 バージョン 1 および DB2 クライアント・アプリケーション・イネーブラー用)
- SQL02010 (DB2 製品 バージョン 2 および DB2 クライアント・アプリケーション・イネーブラー用)
- SQL02020 (DB2 製品 バージョン 2.1.2 およびクライアント・アプリケーション・イネーブラー用)
- SQL05000 (DB2 ユニバーサル・データベースおよび DB2 コネクト製品バージョン 5.0 およびそれらのクライアント用)
- SQL05020 (DB2 ユニバーサル・データベースおよび DB2 コネクト製品バージョン 5.2 およびそれらのクライアント用)
- SQL06010 (DB2 ユニバーサル・データベースおよび DB2 コネクト製品バージョン 6.1 およびそれらのクライアント用)

- SQL07010 (DB2 ユニバーサル・データベースおよび DB2 コネクト製品のバージョン 7 およびそれらのクライアント用)

### コード・ページ ID

モニターされるアプリケーションが開始するノードでのコード・ページ識別子。

この情報を使用して、アプリケーション・コード・ページとデータベース・コード・ページ (ホストまたは AS/400 データベース・サーバー・データベースの場合は、ホストまたは AS/400 データベース・サーバー CCSID) との間のデータ変換がサポートされているかどうかを確認することができます。

アプリケーション・コード・ページが、データベース・システム・モニター実行中のコード・ページとは異なる場合、このコード・ページ要素はアプリケーションから渡され、データベース・システム・モニターによって表示されたデータを手動で変換する際の助けとなります。たとえば、アプリケーション名の変換を助けるためにそれを使用することができます。

### アウトバウンド順序番号

これは、アウトバウンド順序番号を表しています。アウトバウンド順序番号は、異なるシステム上でトランザクションを関連させるために使用されます。

### ホスト・データベース名

アプリケーションが接続されるデータベースの実名。DCS ディレクトリー内では、これはターゲット・データベース名です。

### ホスト製品 ID

サーバー上で実行している製品およびバージョン。その形式は *PPPVVRRM* で、各文字は以下のとおりです。

**PPP** ホストまたは AS/400 データベース・サーバー製品を識別します (たとえば、DSN は DB2 ユニバーサル・データベース (OS/390 版)、ARI は DB2 (VSE および VM 版)、QSQ は DB2 ユニバーサル・データベース (AS/400 版) など)。

**VV** 2 桁のバージョン番号です。たとえば、01。

**RR** 2 桁のリリース番号。

**M** 1 桁の修正レベルです。

## LIST DCS APPLICATIONS EXTENDED

LIST DCS APPLICATIONS コマンドにオプション EXTENDED を指定して使用し、拡張レポートを生成することができます。拡張レポートには、コマンドで SHOW DETAIL オプションを指定したときにリストされるすべてのフィールドと、9つの新規フィールドがリストされます。

- DCS アプリケーション状況
- 状況変更時刻
- クライアント・プラットフォーム
- クライアント・プロトコル
- ホスト・コード化文字セット識別子 (CCSID)
- クライアント・ログイン ID
- クライアント・アプリケーションのプロセス ID
- ゲートウェイでのデータベース別名
- DCS データベース名

既存のコマンド・オプションはアプリケーションごとに1行を使ってフィールドを横にリストしましたが、新しいオプションはフィールドごとに1行を使ってフィールドを縦にリストします。

新しいコマンド構文は次のとおりです。

```
LIST DCS APPLICATIONS [SHOW DETAIL | EXTENDED ]
```

このコマンドで新しいオプション EXTENDED を指定したときの出力例を示します。

```
                List of DCS Applications - Extended Report
Client application ID      = 09151251.0AA7.981015204853
Sequence number           = 0001
Authorization ID          = NEWTON
Application name          = db2bp
Application handle        = 1
Application status        = waiting for request
Status change time       = 10-15-1998 16:50:29.489160
Client node               = antman
Client release level      = SQL05020
Client platform           = AIX
Client protocol           = TCP/IP
Client codepage           = 819
Process ID of client application = 39324
Client login ID           = smith
Host application ID       = G9151251.GAA8.981015204854
Sequence number           = 0000
Database alias at the gateway = MVSDB
DCS database name        = DCSDB
```

Host database name	= GILROY
Host release level	= DSN05011
Host CCSID	= 500

アプリケーション状況フィールドには、次の 3 つの値のいずれかが含まれます。

1. **connect pending - outbound**。これは、ホスト・データベースへの接続要求が発行され、DB2 コネクトが接続の確立されるのを待っていることを意味しています。
2. **waiting for request**。これは、ホスト・データベースへの接続が確立され、DB2 コネクトがクライアント・アプリケーションからの SQL ステートメントを待っていることを意味しています。
3. **waiting for reply**。これは、SQL ステートメントがホスト・データベースに送信されたことを意味しています。

状況変更時刻は、処理中にシステム・モニター UOW スイッチがオンに変更された場合にのみレポートに表示されます。それ以外の場合は、“Not Collected”が表示されます。

## DB2 コントロール・センターを使用して拡張 DCS アプリケーション情報をリストする

DB2 バージョン 7 のコントロール・センターを使用して、DB2 コネクトのゲートウェイのモニターを実行できます。このセクションでは、コントロール・センターを使用して、**list dcs applications extended** コマンドの場合と同じレポートを表示する方法を示します。

任意のアプリケーションの拡張レポートを表示するには、以下のようにします。

1. コントロール・センターの「システム (**systems**)」アイコンの下にあるツリーを展開して、「システム (**Systems**)」→「インスタンス (**Instances**)」→「ゲートウェイ接続 (**Gateway Connections**)」を表示します。ゲートウェイ接続 (**Gateway Connections**) フォルダーの下のインスタンスを右マウス・クリックすると、ポップアップ・メニューが表示されます。このメニューから、「アプリケーション (**Applications...**)」を選択してください。「アプリケーション (**Applications**)」ウィンドウが表示されます。このウィンドウにはタブ形式のノートブックが表示され、「アプリケーション (**Applications**)」という表題のタブがあります。インスタンス中にゲートウェイ・アプリケーションがあると、「ゲートウェイ・アプリケーション (**Gateway Applications**)」という表題の 2 つ目のタブが表示されます。



2. 各ページのメイン・ウィンドウには、LIST DCS APPLICATIONS EXTENDED レポートのフィールドに対応する情報の列があります。ウィンドウに表示される最初の 6 列には、次のようなデータが含まれています。

```
Client Node
Application Name
Client Application ID
Host Application ID
Database Alias at Gateway
Status
```

ウィンドウの下部にある水平スクロール・バーを移動すると、レポートの残りのフィールドを表示できます。

この視点には、LIST DCS APPLICATIONS EXTENDED コマンドを実行した場合にリストされるすべてのフィールドがリストされます。

---

## Windows パフォーマンス・モニタの使用

Windows NT と Windows 2000 には、DB2 アプリケーションのパフォーマンスをモニターする上で役立つツールがあります。それは Windows 管理ツールの 1 つであるパフォーマンス・モニタです。このツールは、システムのパフォーマンスをグラフに表示します。モニターの対象としてさまざまなシステム、データベース、通信関連の項目を選び、それらをまとめてグラフ表示にマップすることができます。

たとえば、**GET SNAPSHOT FOR ALL DCS DATABASES** コマンドや **GET SNAPSHOT FOR ALL DCS APPLICATIONS** コマンドで取得できるレポートを、このモニターを使ってリアルタイムにグラフ化し、CPU 使用率といった値を直接比較できます。また、種々の設定がデータベースや通信のパフォーマンスに与える影響も直接比較できます。設定値の個々の構成内容は PMC ファイルに保管して、後で取り出すことができます。

たとえば、以下に示す図では CPU 使用率を軸に、いくつかの DB2 測定値をグラフ化しています。ここでグラフ化されている値の集合は、ファイル db2chart.pmc に保管されたものです。PMC ファイルはいくつでも保管することができ、それぞれのファイルにシステム・パフォーマンスの横断図を反映させることが可能です。

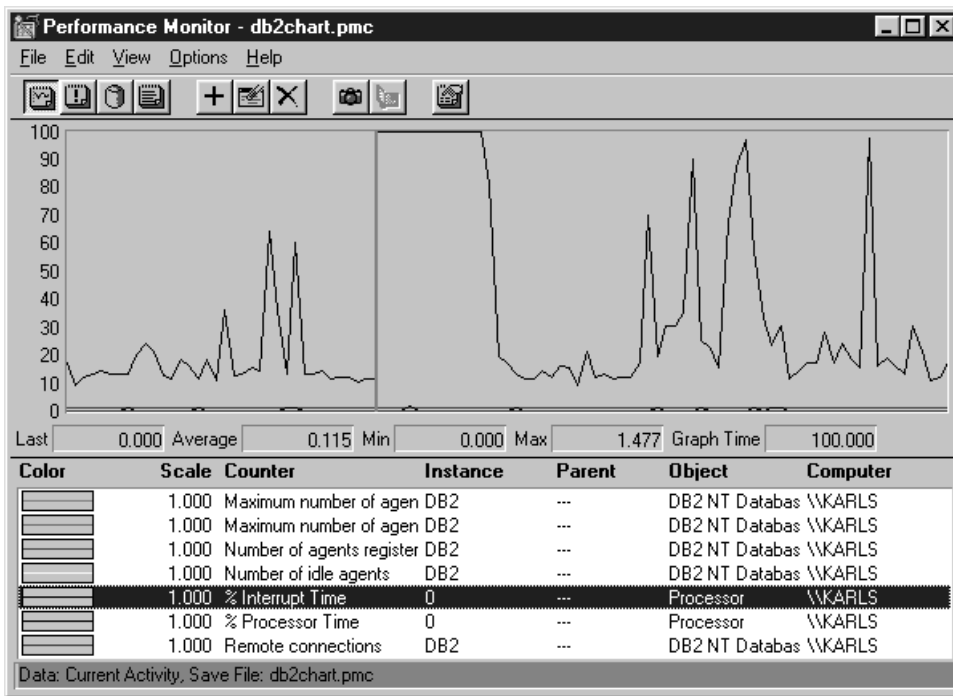


図 5. パフォーマンス・モニタ

ローカル・アプリケーションのモニターを有効にするには、  
DB2CONNECT\_IN\_APP\_PROCESS 環境変数をオフにする必要があります。

さらに多くの DCS APPLICATIONS コマンドについて知りたい場合、あるいは、Windows パフォーマンス・モニタを DB2 UDB および DB2 コネクトと組み合わせて使う方法について詳しくは、[管理の手引き](#) を参照してください。

---

## 第9章 管理ユーティリティー

この章では、管理作業を行う上で役立つユーティリティーについて説明します。ここには、以下のセクションが含まれています。

- 『コマンド行プロセッサ』
- 122ページの『インポートおよびエクスポート・ユーティリティーの使用』

---

### コマンド行プロセッサ

コマンド行プロセッサを用いることにより、ホストまたは AS/400 データベース・サーバーに対して、db2 を接頭部を持つ SQL ステートメントを発行することができます。ホストおよび AS/400 SQL と、DB2 コネクト SQL の違いについては、63ページの『DB2 コネクトがサポートするホストまたは AS/400 サーバー SQL ステートメント』 および 63ページの『DB2 コネクトが拒否するホストまたは AS/400 サーバー SQL ステートメント』を参照してください。

コマンド行プロセッサを対話入力モードで起動するには、以下のようにします。

**OS/2** 「OS/2 Warp」をクリックして「IBM DB2」→「コマンド行プロセッサ (Command Line Processor)」の順に選択するか、db2 コマンドを入力します。

**Windows 32 ビット オペレーティング・システム**  
「スタート」をクリックして、「プログラム」→「IBM DB2」→「コマンド行プロセッサ (Command Line Processor)」の順に選択します。

注: さらに、コマンド行プロセッサを対話入力モードで起動するには、MS-DOS プロンプトで db2 コマンドの後に db2cmd コマンドを入力します。

**UNIX** コマンド行プロセッサで db2 コマンドを入力します。

対話入力モードでは、プロンプトが次のように表示されます。

```
db2 =>
```

対話入力モードでは、DB2 コマンドの最初に db2 を付けて入力する必要はなく、DB2 コマンドをそのまま入力します。以下に例を示します。

```
db2 => list database directory
```

対話モードでオペレーティング・システムのコマンドを入力するときは、そのコマンドの最初に感嘆符 (!) を付けます。以下に例を示します。

```
db2 => !dir
```

1 行で収まらない長いコマンドを入力しなければならないときは、スペースに続けて行連結文字 ¥ を入力してください。そして行の終わりで **Enter** キーを押し、次の行でコマンドの続きを入力します。以下に例を示します。

```
db2 => select empno, function, firstname, lastname, birthdate, from ¥
db2 (cont.) => employee where function='service' and ¥
db2 (cont.) => firstname='Lily' order by empno desc
```

対話入力モードを終了するには、**quit** コマンドを入力します。

コマンド行プロセッサを使用する前には、101ページの『第7章 アプリケーションとユーティリティのバインド』に記載されているとおりに、コマンド行プロセッサをバインドしなければなりません。

**注:** コマンド行プロセッサを UNIX ベースのシステムで、対話式入力モード以外で使用するときは、SQL ステートメントを発行する際に、特殊文字 (たとえば、\* および ?) の前後に二重引用符を付ける必要があります。

SQL コマンドについて詳しくは、**コマンド解説書** を参照してください。

---

## インポートおよびエクスポート・ユーティリティの使用

インポートおよびエクスポート・ユーティリティは、DB2 コネクト ワークステーション上のファイルと、S/390 ホストまたは AS/400 データベースの間でデータを移動することができます。他の任意のアプリケーションや、このインポート / エクスポート形式をサポートする RDBMS で、このデータを使用することができます。たとえば、DB2 (OS/390 版) のデータを区切り ASCII ファイルにエクスポートし、後でそれを DB2 UDB (Windows 2000 版) データベースにインポートすることができます。

データベース・クライアントまたは DB2 コネクト ワークステーションからエクスポートおよびインポート機能を実行することができます。

**注:**

1. インポートおよびエクスポートするデータは、双方のデータベースのサイズとデータ・タイプ制限に従う必要があります。
2. インポートのパフォーマンスを改善するため、複合 SQL を使用することができます。インポート API または CLP *filetype-mod* スtring・パラメーターにおいて COMPOUND=*number* を指定することにより、指定した数の SQL ステートメントをブロックとしてグループ化してください。このようにすればネットワーク・オーバーヘッドを減らすことができ、応答時間が改善されます。
3. コマンド行プロセッサからのインポートおよびエクスポート・ユーティリティの構文についての詳細は、コマンド解説書 をご覧ください。

## **ワークステーションから S/390 または AS/400 データベース・サーバーへのデータの移動**

S/390 または AS/400 データベース・サーバーへのエクスポートは次のように行います。

1. 行単位の情報を DB2 UDB (UNIX 版、Windows NT 版、Windows 2000 版) の表から PC/IXF ファイルにエクスポートします。
2. DRDA サーバー・データベースに、インポートされる情報と互換性のある属性を記述した表がないなら、互換性のある表を作成します。
3. INSERT オプションを使って、PC/IXF ファイルを DRDA サーバー・データベースの表にインポートします。

## **DRDA サーバーからワークステーションへのデータの移動**

DRDA サーバー・データベースからのデータのインポートは次のように行います。

1. 行単位の情報を DRDA サーバーのデータベース表から PC/IXF ファイルにエクスポートします。
2. PC/IXF ファイルを使って DB2 表にインポートします。

### **制約事項**

DB2 コネクト・プログラムを使用するとき、インポートまたはエクスポートの操作は以下の条件に合致している必要があります。

- ファイル・タイプは PC/IXF でなければなりません。
- 索引定義はエクスポートにおいて保管されず、インポートにおいて使用されません。

- それらのデータと互換性のある属性の表が、インポートする前に存在している必要があります。INSERT はサポートされている唯一のオプションなので、DB2 コネクト・プログラムを通してのインポートでは表を作成することはできません。
- インポートの際は、コミット・カウント間隔を指定しないでください。

これらの条件が満たされない場合、操作は失敗に終わり、エラー・メッセージが表示されます。

## 1 バイトおよび 2 バイト・データの混合

混合データ (1 バイトと 2 バイト・データの両方を含む列) のインポートおよびエクスポートを行う場合は、以下の点を考慮に入れてください。

- データを EBCDIC で保管するシステム (MVS、OS/390、OS/400、VM、および VSE) では、2 バイト・データの始まりと終わりはシフトアウト (SO) 文字およびシフトイン (SI) 文字で印付けられます。データベース表の列の長さを定義するときには、これらの文字のための十分な余地を見込んでください。
- 列内のデータが一定のパターンになっているのでない限り、可変長文字列の使用をお勧めします。一定のパターンになっているのであれば、固定長でもかまいません。

## SQLQMF ユーティリティの置き換え

DDCS (OS/2 版) の SQLQMF ユーティリティ機能は、DB2 コネクトのインポート / エクスポート機能に置き換えられました。利点は、以下のとおりです。

- ホスト上では、QMF を必要としません。
- ホストにログオンする必要がありません (DB2 (MVS/ESA 版) または DB2 ユニバーサル・データベース (OS/390 版) では、引き続き TSO ID が必要)。
- DB2 (MVS 版)、DB2 (OS/390 版)、DB2 (AS/400 版)、および DB2 (VM および VSE 版) がサポートされています。
- 複合 SQL を使用することによってパフォーマンスが改善されます。
- ASCII に加えていくつかのファイル形式をサポートします。
- SNA 接続なしでクライアントのマシンから実行できます。

これらのコマンドの使用について詳しくは、コマンド解説書を参照してください。

---

## 第10章 機密保護

この章では、認証タイプを含む DB2 コネクトの機密保護の考慮事項について説明します。また、DB2 ユニバーサル・データベース (OS/390 版) ユーザーの機密保護に関する追加のヒントも提供しています。

DCE での機密保護のセットアップ方法について詳しくは、*管理の手引き* を参照してください。また、ご使用のホストまたは AS/400 データベース・サーバー用のデータベースおよび DCE のマニュアルも参照してください。

**注:** DCE 機密保護とともに DB2 コネクトを使用している場合、DB2 クライアント・ワークステーションとホストまたは AS/400 データベース・サーバー上で DCE ソフトウェアをインストールする必要がありますが、DB2 コネクト・ワークステーション上でそれをインストールする必要はありません。DCE のソフトウェア前提条件に関する詳細については、*DB2 コネクト 概説* および *インストール* を参照してください。

---

### 認証

DB2 コネクト管理者として、ホストまたは AS/400 データベースの管理者と協力して、ユーザー名とパスワードを検証する場所を決めることができます。5 つの可能性があります。

- クライアントでの検証
- DB2 コネクト・ワークステーションでの検証
- DB2 コネクト・ワークステーションとホストまたは AS/400 サーバーの両方での検証
- ホストまたは AS/400 サーバーでの検証
- DCE 機密保護サーバーでの検証

検証を行う場所は、システム・データベース・ディレクトリー内で認証タイプ・パラメーターを、APPC ノードか APPN ノードのノード・ディレクトリー内で機密保護タイプ・パラメーターを、それぞれ設定することによって決定します。これらのディレクトリーの更新の詳細については、85ページの『第6章 データベース・ディレクトリーの更新』を参照してください。

**注:**

1. DB2 コネクト自体はユーザー検証を行いません。DB2 コネクト・ワークステーションに検証を実行させたい場合、各 CONNECT 要求で提供されているユーザー ID およびパスワードを確認するために、ローカル・セキュリティ・サブシステムが使用されます。したがって、DB2 コネクト エンタープライズ・エディション・サーバーのセットアップ時に AUTHENTICATION=SERVER を使用する場合、サーバー・システム上の必要なすべてのユーザー ID およびパスワードをセットアップしなければなりません。
2. DCE ディレクトリー・サービスを使用している場合は、認証の作業は異なったものになります。詳細については、231ページの『DCE ディレクトリー・サービスと機密保護』を参照してください。

DB2 コネクトでは、以下の認証タイプが許可されています。

**CLIENT**

クライアントでユーザー名とパスワードの妥当性を検査します。

**SERVER**

DB2 コネクト・ワークステーションでユーザー名とパスワードの妥当性を検査します。認証が指定されない場合には、**SERVER** が想定されます。

**SERVER\_ENCRYPT**

SERVER 認証の場合に、DB2 コネクト・ワークステーションでユーザー名とパスワードの妥当性を検査しますが、その際に転送されるパスワードはクライアントで暗号化され、DB2 コネクト・ワークステーションで暗号化解除されます。

**DCS** AS/400 データベース・サーバーでユーザー名とパスワードの妥当性を検査します。

**DCS\_ENCRYPT**

DCS 認証の場合に、ホストまたは AS/400 データベース・サーバーでユーザー名とパスワードの妥当性を検査します。ただし、転送されるパスワードはクライアントで暗号化され、DB2 コネクト・ワークステーションで指定された認証タイプに従い、DB2 コネクト・ワークステーションか、ホストまたは AS/400 データベース・サーバーで暗号化解除されます。

**DCE** DCE 機密保護サーバーでユーザー名とパスワードの妥当性を検査します。



SERVER\_ENCRYPT 認証の認証場所は SERVER 認証と同じで、DCS\_ENCRYPT 認証の認証場所は DCS 認証と同じです。異なる点は、転送されるパスワードが、ソースでカタログされている認証タイプの指定に応じて、ソース (クライアントまたは DB2 コネクト・サーバー) で暗号化され、ターゲット (DB2 コネクト・サーバーまたはホストか AS/400 データベース・サーバー) で暗号化解除されることです。

続いて、対応する認証場所を持つ暗号化値と非暗号化値を使用して、クライアントと DB2 コネクト・サーバーとの間で、あるいは DB2 コネクト・サーバーとホストまたは AS/400 データベース・サーバーとの間でさまざまな暗号化の組み合わせを選択できますが、この処理は認証の行われる場所には影響を与えません。以下の例には、ゲートウェイ (DB2 コネクト・サーバー) のシナリオでこの処理が行われる様子が示されています。

クライアントでの認証	ゲートウェイでの認証	認証場所	クライアント - ゲートウェイ間の暗号化	ゲートウェイ - サーバー間の暗号化
SERVER_ENCRYPT	SERVER	ゲートウェイ	あり	なし
DCS_ENCRYPT	DCS	サーバー	あり	なし
DCS	DCS_ENCRYPT	サーバー	なし	あり
DCS_ENCRYPT	DCS_ENCRYPT	サーバー	あり	あり

SERVER\_ENCRYPT または DCS\_ENCRYPT を使用した場合にサポートされる APPC 機密保護パラメーターは SECURITY=NONE だけです。

**注:**

1. DB2 コネクトが接続を確立するために使用するシステム・データベース・ディレクトリー項目について、認証パラメーターを指定しないと、DB2 コネクトは認証 **SERVER** を使用します。
2. DB2 ユニバーサル・データベースのクライアント - サーバー通信の場合のように、DB2 コネクト エンタープライズ・エディション・ゲートウェイに接続されたりリモート・クライアントでは認証タイプは必要ではありません。ただし、パフォーマンスを最適化するためにそこで認証タイプを指定することがあります。ゲートウェイから認証タイプを獲得する必要がないので、トランザクションの所要時間は短縮されます。
3. クライアントの値とゲートウェイの値との間に矛盾がある場合には、DB2 コネクト・ゲートウェイで指定されている値が優先されます。

## 機密保護タイプ

このセクションでは、APPC 接続と TCP/IP 接続の両方で DB2 コネクトによってサポートされる認証と機密保護設定の様々な組み合わせをリストします。

以下の説明は、両方のタイプの接続に適用されます。

### APPC 接続用の機密保護タイプ

どの機密保護情報が通信層で流れるかを指定するために、以下の機密保護タイプが APPC 接続で許可されています。

**SAME** ユーザー名だけがホストまたは AS/400 データベース・サーバーに渡されます。

#### PROGRAM

ユーザー名とパスワードがホストまたは AS/400 データベース・サーバーに渡されます。

**NONE** 機密保護情報は流れません。

表5 は、DB2 コネクト・ワークステーション上で指定されるこれらの値と認証タイプの可能な組み合わせ、およびその各組み合わせについて検証が行われる場所を示します。この表に示される組み合わせだけが、APPC 接続上の DB2 コネクトによってサポートされます。

表5. APPC 接続用の有効な機密保護シナリオ

ケース	DB2 コネクト・ワークステーションのデータベース・ディレクトリ項目における認証の設定	機密保護	検証
1	CLIENT	SAME	クライアント
2	SERVER	SAME	DB2 コネクト・サーバー
3	SERVER	PROGRAM	DB2 コネクト・サーバーとホストまたは AS/400 データベース・サーバー
4	SERVER_ENCRYPT または DCS_ENCRYPT	NONE	ホストまたは AS/400 データベース・サーバー

表 5. APPC 接続用の有効な機密保護シナリオ (続き)

ケース	DB2 コネクト・ワークステーションのデータベース・ディレクトリー項目における認証の設定	機密保護	検証
5	DCS	PROGRAM	ホストまたは AS/400 データベース・サーバー
6	DCE	NONE	DCE 機密保護サーバー

リモート・クライアントが DB2 コネクト・エンタープライズ・エディション・サーバーに接続されている場合は、次の認証と機密保護のタイプを指定します。

- リモート・クライアントが APPC を介して DB2 コネクト・サーバーに接続されている場合は、機密保護タイプ NONE をリモート・クライアントで指定します。
- DB2 コネクト・サーバーのデータベース・マネージャー構成での認証タイプが CLIENT の場合は、各リモート・クライアントで CLIENT を指定します。
- DB2 コネクト・サーバーでの認証タイプが SERVER、SERVER\_ENCRYPT、DCS、または DCS\_ENCRYPT のいずれかである場合は、各リモート・クライアントでこれらのタイプのいずれかを指定します。(これら 4 つのタイプのどれをリモート・クライアントで指定しても相違はありません。)

注:

1. AIX ベースのシステムについては、APPC 機密保護タイプ SAME を使用するログイン・ユーザーはすべて、AIX システム・グループに属している必要があります。
2. リモート・クライアントを有する AIX ベースのシステムについては、DB2 コネクト・ワークステーション上で稼働する DB2 コネクト製品のインスタンスは、AIX システム・グループに属している必要があります。
3. ホストまたは AS/400 データベース・サーバーへのアクセスは、それ自体の機密保護機構またはサブシステムによって制御されています。たとえば、仮想記憶通信アクセス方式 (VTAM) および資源アクセス管理機能 (RACF) がそれです。保護されたデータベース・オブジェクトへのアクセスは、SQL GRANT および REVOKE ステートメントによって制御されます。

## TCP/IP 接続用の機密保護タイプ

TCP/IP 通信プロトコルは、ネットワーク・プロトコル層での機密保護オプションをサポートしません。したがって、認証タイプのみが、認証が行われる場所を制御します。この表に示される組み合わせだけが、TCP/IP 接続上の DB2 コネクトによってサポートされます。

表 6. TCP/IP 接続用の有効な機密保護シナリオ

ケース	DB2 コネクト・ワークステーションのデータベース・ディレクトリ項目における認証の設定	検証
1	CLIENT	クライアント
2	SERVER または SERVER_ENCRYPT	DB2 コネクト・ワークステーション
3	適用外	なし
4	DCS または DCS_ENCRYPT	ホストまたは AS/400 データベース・サーバー
5	DCE	DCE 機密保護サーバー

## 機密保護タイプについて

以下の記述は、上記で説明され、128ページの表5 と表6 にリストされているように、APPC 接続と TCP/IP 接続の両方に適用されます。各ケースについて、より詳細に説明します。

- ケース 1 は、ユーザー名とパスワードがリモート・クライアントでのみ検証される場合です。(ローカル・クライアントの場合、ユーザー名とパスワードが検証されるのは DB2 コネクト・サーバーだけです。)

ユーザーは、最初にサインオンした場所で認証されるよう期待されています。ユーザー ID はネットワークを通して送信されますが、パスワードは送信されません。このタイプの機密保護は、すべてのクライアント・ワークステーションが信頼の置ける適切な機密保護機構を持っている場合にのみ使われます。

- ケース 2 の場合、ユーザー名とパスワード妥当性が検査されるのは、DB2 コネクト・サーバーだけです。パスワードは、ネットワークを介してリモート・クライアントから DB2 コネクト・サーバーへ (ホストまたは AS/400 データベース・サーバーへではない) に送信されます。
- ケース 3 の場合、ユーザー名とパスワードの妥当性は、DB2 コネクト・サーバーとホストまたは AS/400 データベース・サーバーの両方で検査されま

す。パスワードは、ネットワークを介してリモート・クライアントから DB2 コネクト・ワークステーションへ、次いで DB2 コネクト・ワークステーションからホストまたは AS/400 データベース・サーバーへ送信されます。

検証が 2 つの場所で行われるため、同じセットのユーザー名とパスワードが DB2 コネクト・サーバーとホストまたは AS/400 データベース・サーバーの両方で維持されなければなりません。

- ケース 4 は、ユーザー名とパスワードがホストまたは AS/400 データベース・サーバーでのみ検証される場合です。ユーザー ID とパスワードは、ネットワークを介してリモート・クライアントから DB2 コネクト・サーバーへ、次いで DB2 コネクト・サーバーからホストまたは AS/400 データベース・サーバーへ送信されます。
- ケース 5 では、DCE の暗号化されたチケットがクライアントによって DCE 機密保護サーバーから獲得されます。チケットは、DB2 コネクトを通してサーバーに変更なしで渡され、ここで、DCE 機密保護サービスを使用しているサーバーが妥当性検査を行います。

---

## MVS パスワードの変更

現在 DB2 コネクトには、ユーザー・パスワードを変更する機能が備えられています。この機能は、資源アクセス管理機能 (RACF) などのホスト機密保護サービスを使用してユーザーを認証する場合に特に有効です。以前は、ホストのパスワードを変更するのに、ユーザーが TSO セッションにログインして自分のパスワードを変更する必要がありました。DB2 コネクト製品に備えられている新しいパスワード保守サポートを使用すると、DB2 コマンド行プロセッサ (CLP) から SQL CONNECT ステートメントを発行するか、DB2 クライアント構成支援機能 (CCA) の「パスワード (PASSWORD)」ボタンを使用するか、または ODBC ログイン・ダイアログの「変更 (CHANGE)」ボタンを押して、自分のホスト・パスワードを変更できます。

TCP/IP を介して DB2 (OS/390 版) V5.1 に接続している DB2 コネクト・ワークステーションで MVS パスワードを変更するには、「DB2 (OS/390 版) 拡張セキュリティ (DB2 OS/390 Extended Security)」フィールドが「はい (Yes)」に設定されている必要があります。このフィールドは、DB2 OS/390 DSNTIPR パネルに表示されます。

SNA を介して接続しているホスト・システム上で MVS パスワードを変更するには、特別なパスワード有効期限管理プログラムをホスト上でセットアップし、このホスト・システムと通信するように DB2 コネクト・ワークステーションを構成する必要があります。

ホスト・パスワード有効期限管理プログラムは、以下の MVS プログラム製品の一部として備えられています。

- MVS/ESA SP バージョン 4.2 以降 (パスワード有効期限管理が APPC/MVS 構成要素の一部になっている)
- CICS/ESA バージョン 3.3 以降

また、IBM 資源アクセス管理機能 (RACF) 1.9.2 がインストールされています。

以下の作業を行う必要があります。

1. パスワード有効期限の保守に関する要求を受信するようホストのトランザクション・プログラムを構成する。
2. ホストのトランザクション・プログラムと通信するよう DB2 コネクト・ワークステーションを構成する。

## DB2 コネクト・ワークステーションでのパスワード有効期限管理の構成

ホスト・パスワード有効期限トランザクション・プログラムを構成し終えたら、ホスト・プログラムと通信するよう DB2 コネクト・ワークステーションを構成する必要があります。この構成には、以下の 2 つのステップが含まれます。

1. DB2 コネクト・ワークステーション上の SNA サブシステムで、ホスト・パスワード保守プログラムの記号宛先名を定義する。
2. この記号宛先名を、このホスト・システム上のデータベースの DCS ディレクトリに記録する。

### ステップ 1. 記号宛先名を定義する

ホスト・パスワード有効期限管理プログラムの記号宛先名を定義する方法は、使用している SNA サブシステムに応じて変わります。

- 使用中の SNA サブシステムを DB2 クライアント構成支援機能 (CCA) で構成できる場合は、CCA を使用してこの記号宛先名を構成する。パスワード有効期限管理プログラムの LU 名を、MVS 管理者から入手する必要があります。
- 使用中の SNA サブシステムを DB2 CCA で構成できない場合は、SNA サブシステムに付属の資料に従って記号宛先名を構成する。以下の情報を MVS 管理者から入手する必要があります。
  - 接続しているホストのネットワーク名
  - ホスト・パスワード有効期限管理プログラムの LU 名

記号宛先名を構成するには、トランザクション・プログラム (TP) 名として x'06F3F0F1' (16 進数) を指定し、機密保護を「NONE」に設定する必要があります。#INTER や、MVS で推奨されている他のモードも指定できます。

## ステップ 2. 記号宛先名を DCS ディレクトリーに記録する

DB2 CCA が備わっているプラットフォームで DB2 コネクトを実行している場合は、DB2 CCA を使用し、ホスト・パスワード有効期限管理プログラムの記号宛先名で DCS ディレクトリーを更新する必要があります。この処理は、DB2 コネクト・ワークステーション上の SNA サブシステムの種類に関係なく行う必要があります。

また、**catalog dcs database** コマンドを (DB2 CLP から) 使用しても、記号宛先名を DCS ディレクトリーに記録することができます。以下に例を示します。

```
catalog dcs database db1 as dsn_db_1 parms ",,,,,,CHGPWD_SDN=pempgm"
```

この場合は、記号宛先名として「pempgm」が記録されます。この名前は、ユーザーがデータベース「db1」のパスワード変更を要求する際に使用されます。

## ホストでのパスワード有効期限管理の構成

MVS パスワードについて詳しくは、以下のいずれかのオンライン・マニュアルを参照してください。

### APPC/MVS:

<http://www.s390.ibm.com/products/appc/library>

### SecureWay Communications:

<http://www.ibm.com/software/network/commserver/library>

### TxSeries:

<http://www.ibm.com/software/ts/txseries/library>

---

## 機密保護に関する追加のヒント

このセクションでは、DB2 コネクトのユーザーの機密保護に関する追加のヒントを提供しています。

## 拡張機密保護コード

DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 よりも前のバージョンでは、ユーザー ID またはパスワードを指定した接続要求が失敗した場合、SQL30082 理由コード 0 が返されるものの、問題となっている点に関して他の指示は出されませんでした。

DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 では機能が拡張され、拡張機密保護コードをサポートするようになりました。拡張機密保護を指定すると、理由コードのほかに (PASSWORD EXPIRED) などの診断が追加されます。

これを活用するには、DB2 ユニバーサル・データベース (OS/390 版) の拡張機密保護用に ZPARM インストール・パラメーターの値を YES に設定する必要があります。EXTSEC=YES に設定するには、DB2 ユニバーサル・データベース (OS/390 版) のインストール・パネル DSN6SYSP を使用します。これを設定するために、DDF パネル 1 (DSNTIPR) を使用することもできます。省略時値は EXTSEC=NO です。パスワードの有効期限が切れた場合、PC、UNIX、Apple Macintosh、および DB2 コネクトを使用している Web アプリケーションはエラー・メッセージ SQL01404 を受け取ります。

## 検査済みの TCP/IP 機密保護

DB2 ユニバーサル・データベースの機密保護オプション AUTHENTICATION=CLIENT をサポートさせたい場合、DB2 ユニバーサル・データベース (OS/390 版) のインストール・パネル DSNTIP4 (DDF パネル 2) を使用して、検査済みの TCP/IP 機密保護を YES に設定します。

## デスクトップ ODBC および Java アプリケーションの機密保護

ワークステーション ODBC および Java アプリケーションは動的 SQL を使用します。この動的 SQL はインストールの際に機密保護上の問題の原因となることがあります。DB2 ユニバーサル・データベース (OS/390 版) は、所有者とバインダーの両方の権限で動的 SQL を実行できる、新しいバインド・オプション DYNAMICRULES(BIND) を導入しています。DYNAMICRULES を DB2 コネクトを介して指定する方法については、コマンド解説書を参照してください。

DB2 ユニバーサル・データベースおよび DB2 コネクトには、DB2CLI.INI 構成ファイル内に新しい CLI/ODBC 構成パラメーター CURRENTPACKAGESET があります。これは適切な特権を持つスキーマ名に設定することが必要です。SQL SET CURRENT PACKAGESET schema ステートメントは、アプリケーションの各接続ごとに自動的に出されます。



DB2CLI.INI を更新するためには、ODBC マネージャーを使用します。詳しくは、インストールおよび構成 補足 を参照してください。

## パスワード変更サポート

SQL CONNECT ステートメントがユーザー ID のパスワードの有効期限が切れていることを示すメッセージを戻す場合、DB2 コネクトのバージョン 5.2 以上では TSO にサインオンしなくてもパスワードを変更できるようになっています。DB2 ユニバーサル・データベース (OS/390 版) では、DRDA を介してパスワードを変更できます。

旧パスワードと新規パスワード、および確認用パスワードを入力する必要があります。DB2 コネクト エンタープライズ・エディションのサーバーで指定された機密保護が DCS である場合、パスワード変更の要求は DB2 ユニバーサル・データベース (OS/390 版) データベース・サーバーに送られます。指定した機密保護が SERVER である場合、DB2 コネクト・サーバーのパスワードは変更されます。

さらに別の利点として、別個の LU 定義が不要であることがあります。追加情報は、DB2 コネクト エンタープライズ・エディションの概説およびインストール を参照してください。



---

## 第11章 SQLCODE マッピング

異なる IBM 関係データベース製品は、類似のエラーについて必ずしも同じ SQLCODE を発行するものではありません。SQLCODE が同じであるとしても、指定の異なるトークンを伴うことがあります。トークンのリストは、SQLCA の SQLERRMC フィールド内で渡されます。省略時には、DB2 コネクトは SQLCODE およびトークンを各 IBM ホストまたは AS/400 サーバーから該当する DB2 ユニバーサル・データベース SQLCODE のシステムへマッピングします。

---

### SQLCODE マッピングをオフにする

SQLCODE マッピングをオフにしたいときは、DCS ディレクトリーまたは DCE 経路指定情報のパラメーター・ストリングで NOMAP を指定します。DCS ディレクトリーの更新の情報については、85ページの『第6章 データベース・ディレクトリーの更新』を参照してください。DCE の使用に関する情報については、223ページの『付録D. DCE ディレクトリー・サービスの使用』を参照してください。

アプリケーションをホストまたは AS/400 データベース・サーバー (DB2 UDB (OS/390 版) など) から直接移植する場合は、SQLCODE マッピングをオフにするとよいでしょう。このようにすれば、アプリケーションが参照する SQLCODE を変更することなく、アプリケーションを使用することができます。

---

### SQLCODE マッピングの調整

省略時には、DB2 コネクトは SQLCODE およびトークンを各 IBM ホストまたは AS/400 サーバーから該当する DB2 UDB SQLCODE のシステムへマッピングします。以下のファイルは、省略時の SQLCODE マッピングのコピーです。

#### **dcs1dsn.map**

DB2 (MVS/ESA 版) および DB2 UDB (OS/390 版) の SQLCODE をマッピングします。

#### **dcs1ari.map**

DB2 (VSE および VM 版) の SQLCODE をマッピングします。

## dcslqsq.map

DB2 UDB (AS/400 版) の SQLCODE をマップします。

OS/2 および UNIX ベースの DB2 システムについては、マッピングは必要ありません。

省略時の SQLCODE マッピングを上書きしたい場合、または SQLCODE マッピングを持たないホストまたは AS/400 データベース・サーバー (非 IBM データベース・サーバー) を使用している場合は、これらのファイルの 1 つをコピーして、それを新規の SQLCODE マッピング・ファイルの基礎として使用することができます。そのファイルを直接編集するよりは、むしろそれをコピーすることによって、必要な場合いつでも元の SQLCODE を確実に参照できます。

DCS ディレクトリーまたは DCE 経路指定情報のパラメーター・ストリングに、新規の SQLCODE マッピング・ファイルのファイル名を指定します。DCS ディレクトリーの更新の情報については、85ページの『第6章 データベース・ディレクトリーの更新』を参照してください。DCEの使用に関する情報については、223ページの『付録D. DCE ディレクトリー・サービスの使用』を参照してください。

各マッピング・ファイルは、ASCII エディターを使用して作成、編集できる ASCIIファイルです。初期インストールでは、ファイルはインストール・パス内の map ディレクトリーに保管されます。

ファイルには、以下の特殊タイプの行を含めることができます。

- &&** 理論的なファイルの開始。&& が最初に現れるよりも前の行はすべて、自由形式のコメントとみなされて無視されます。ファイルが && の後に何も含んでいない場合は、SQLCODE マッピングは実行されません。また、前に説明したように、NOMAP パラメーターを使用することにより SQLCODE マッピングをオフにすることもできます。
- \*** 行の最初の文字として、コメントを示します。
- W** 行にある唯一の文字として、警告フラグが再マップされるべきことを示します。省略時には、オリジナルの警告フラグが渡されます。W は大文字でなければなりません。

&& 後のその他のすべての行は、ブランクか、または以下の書式のマッピング・ステートメントのいずれかです。

```
input_code [, output_code [, token_list]]
```

*input\_code* は、以下のいずれかを表します。

*sqlcode*

ホストまたは AS/400 データベース・サーバーからの SQLCODE

**U** すべての未定義の負の数の SQLCODE (このファイルにはリストされていないもの) は、指定された *output\_code* にマップされます。

*output\_code* がこの行で指定されなかった場合は、元の SQLCODE が使用されます。この文字は必ず大文字です。

**P** すべての未定義の正の数の SQLCODE (このファイルにはリストされていないもの) は、指定された *output\_code* にマップされます。

*output\_code* がこの行で指定されなかった場合は、元の SQLCODE が使用されます。この文字は必ず大文字です。

**ccnn** ホストまたは AS/400 データベース・サーバーから得られる SQLSTATE クラス・コード。*nn* は以下のいずれかです。

**00** 非修飾の正常終了

**01** 警告

**02** データなし

**21** カーディナリティー違反

**22** データ例外

**23** 制約違反

**24** 無効なカーソル状態

**26** 無効な SQL ステートメント識別子

**40** トランザクション・ロールバック

**42** アクセス違反

**51** 無効なアプリケーション状態

**55** 前提条件の状態にないオブジェクト

**56** その他の SQL または製品エラー

**57** 利用不能なりソースまたは操作員介入

**58** システム・エラー

マッピング・ファイル内で明示的に指定されていないこのクラス・コードを伴うすべての SQLCODE に対して、指定の *output\_code* が使用されます。この行で *output\_code* が指定されない場合は、元の SQLCODE が、コピーされたトークンなしでそれ自体へマップされます。

**cc** の文字は必ず小文字です。

同じ *input\_code* が 2 回以上マッピング・ファイルに現れた場合は、最初に現れたものが使用されます。

*output\_code* は、出力 SQLCODE を表します。値が指定されていない場合は、元の SQLCODE が使用されます。

また、出力コードを指定する場合は、以下のいずれかを指定することもできます。

- (s) 入力 SQLCODE プラス 製品 ID (ARI, DSN または QSQ) が、SQLCA メッセージ・トークン・フィールドに入れられます。

元の SQLCODE は、唯一のトークンとして戻されます。このオプションは、+965と -969 の例外を除き、未定義の SQLCODE を扱うように設計されています。+965 または -969 が *output\_code* である場合、SQLCA の SQLERRMC フィールドに戻されるトークン・リストには、元の SQLCODE、次に製品識別子、続いて元のトークン・リストが含まれます。

**s** の文字は必ず小文字です。

#### (トークン・リスト)

トークンのリストで、コンマで分離します。特定のトークンをスキップするには、コンマだけで指定します。たとえば、書式 (*t2*..*t4*) は、1 番目と 3 番目の出力トークンは空白値であることを意味します。

各トークンは、番号の書式 (*n*) を持っており、オプションとしてその前に **c**、その後には **c** または **i** を付けます。それは、次のように解釈されます。

**c** この位置にあるトークンのデータ・タイプは CHAR (省略時値) です。**c** が *n* の前にある場合は、入力トークンを示します。**c** が *n* の後にある場合は、出力トークンを示します。**c** の文字は必ず小文字です。

**i** この位置にあるトークンのデータ・タイプは INTEGER です。**i** が *n* の後にある場合は、出力トークンを示します。**i** は *n* の前には来ません。というのは、IBM ホストまたは AS/400 データベース・サーバー製品は、CHAR トークンだけをサポートするからです。**i** の文字は必ず小文字です。

*n* どのホストまたは AS/400 データベース・サーバーのトークンが使用されているかを示す数字です。それらの数字は、出力

SQLCA 内で配置要求される順序で配置されています。その数字は、ホストまたは AS/400 データベース・サーバーのトークンを示しています。その配置は、トークンが SQLCA 内で並べられる順序を示しています。

たとえば、ホストまたは AS/400 データベース・サーバーが 2 つのトークン、1 および 2 を戻すとします。トークン 2 を出力 SQLCA の中でトークン 1 の前に表示したい場合は、(2,1) と指定します。

複数のトークン数字は、それらをピリオドで接続することによって、1 つの CHAR 出力トークンを形成するよう結合させることができます。

コンマは、出力トークンを分離するために使用します。コンマの前にトークンが指定されていない場合、その位置では SQLCA 内に出力トークンは組み込まれません。最後に指定されたトークンに続いて出力 SQLCA に出現するトークンはすべて、ヌル・トークンへマップされます。

図6 は、SQLCODE マッピング・ファイルのサンプルを示します。

```
&&
-007      ,   -007      ,   (1)
-010
-060      ,   -171      ,   (2)
...
-204      ,   -204      ,   (c1.2c)
...
-633      ,   -206      ,   (,c1i)
-30021    ,   -30021    ,   (c1c,c2c)

cc00      ,   +000
...
U          ,   -969      ,   (s)
P          ,   +965      ,   (s)
```

図6. SQLCODE マッピング・ファイル

ファイル内の各マッピング・ステートメントの説明は以下のとおりです。

1. SQLCODE は -007 から -007 へマップされます。ホストまたは AS/400 データベース・サーバーから受信する 1 番目のトークンは、1 番目の出力トークンとして使用され、省略時の設定である CHAR になります。他のトークンは、転送されません。

2. SQLCODE は、-010 から -010 へとマップされます (出力 SQLCODE が指定されていないので)。トークンは、出力 SQLCA に配置されません。
3. SQLCODE は -060 から -171 へマップされます。ホストまたは AS/400 データベース・サーバーから受け取った 1 番目の入力トークンは、破棄されます。2 番目のトークンは、出力 SQLCA で 1 番目のトークンとして使用されます。それは CHAR です。出力 SQLCA に 2 番目のトークンはありません。
4. SQLCODE は -204 から -204 へマップされます。ホストまたは AS/400 データベース・サーバーから受け取る 1 番目と 2 番目のトークンは CHAR です。この 2つの入力トークンは結合され、SQLCA で 1 番目の出力トークンになる 1 つの CHAR 出力トークンを形成します。
5. SQLCODE は -633 から -206 へマップされます。ホストまたは AS/400 データベース・サーバーから受け取られた 1 番目の入力トークンは CHAR です。それは INTEGER へ変換され、出力 SQLCA で 2 番目のトークンとして使用されます。出力 SQLCA の 1 番目のトークンは、コマンドで示されているように、ヌルです。
6. SQLCODE は -30021 から -30021 へマップされます。ホストまたは AS/400 データベース・サーバーから受け取る 1 番目および 2 番目の入力トークンは CHAR です。それらは、出力 SQLCA 内で 1 番目および 2 番目のトークンとして使用されます。
7. 00 クラスの SQLSTATE を指定した SQLCA 内の SQLCODE はすべて、SQLCODE +000 へマップされます。
8. 未定義の SQLCODE はすべて -969 へマップされます。まったく同一なものやマッピングが必要でないものも含め、マップ可能なコードのすべてがリストされている場合に限り、このオプションを使用してください。(s) オプションは、SQLCA の SQLERRMC フィールドに戻されるトークン・リストに、元の SQLCODE、次に、エラーが発生した製品、続いて元のトークン・リストが含まれることを示します。U 項目が組み込まれていない場合、リストされていないコードはすべて、マッピングされずに渡されます。
9. 未定義で正の SQLCODE はすべて、+965 へマップされます。まったく同一なものやマッピングが必要でないものも含め、マップ可能なコードのすべてがリストされている場合に限り、このオプションを使用してください。(s) オプションは、SQLCA の SQLERRMC フィールドに戻されるトークン・リストに、元の SQLCODE、次に、警告が発生した製品、続いて元のトークン・リストが含まれることを示します。P 項目が組み込まれていない場合、リストされていない正のコードはすべて、マッピングされずに渡されます。



---

## 第12章 パフォーマンス

DB2 コネクト製品は、DRDA アプリケーション・サーバー製品、クライアント製品、および通信製品を含む各種の製品と相互に作用し合います。そのパフォーマンスは、これらの構成部品と一緒に効率よく働くことに依存します。

---

### パフォーマンスの概念とツール

パフォーマンスは、コンピューター・システムが与えられた特定の作業負荷に対して作用する仕方です。それは、利用可能なリソースと、それらがどのように使用され分配されるかによって影響を受けます。パフォーマンスを改善しようとする場合は、まず最初にパフォーマンスの具体的な意味を決定しなければなりません。各種のパフォーマンス・メトリック を選ぶことができますが、その一部には次のものが含まれます。

#### 応答時間

アプリケーションがデータベース要求を送信した時からアプリケーションが応答を受信する時までの時間間隔。

#### トランザクション・スループット

時間単位ごとに完了できる作業単位の数。作業単位は、行の取り出しや更新のように単純なものから、数百の SQL ステートメントが関係するような複雑なものまであります。

#### データ転送速度

時間単位ごとに DB2 コネクト・アプリケーションとホストまたは AS/400 データベースとの間で転送されるデータのバイト数。

パフォーマンスは、利用可能なハードウェアおよびソフトウェア・リソースによって限定されます。CPU、メモリー、およびネットワーク・アダプターは、ハードウェア・リソースの例です。通信サブシステム、ページング・サブシステム、mbuf (AIX 用)、および link (SNA 用) は、ソフトウェア・リソースの例です。

#### データ・フロー

144ページの図7 は、ホストまたは AS/400 データベース・サーバーとワークステーションの間を DB2 コネクトを介して流れるデータの経路を示しています。

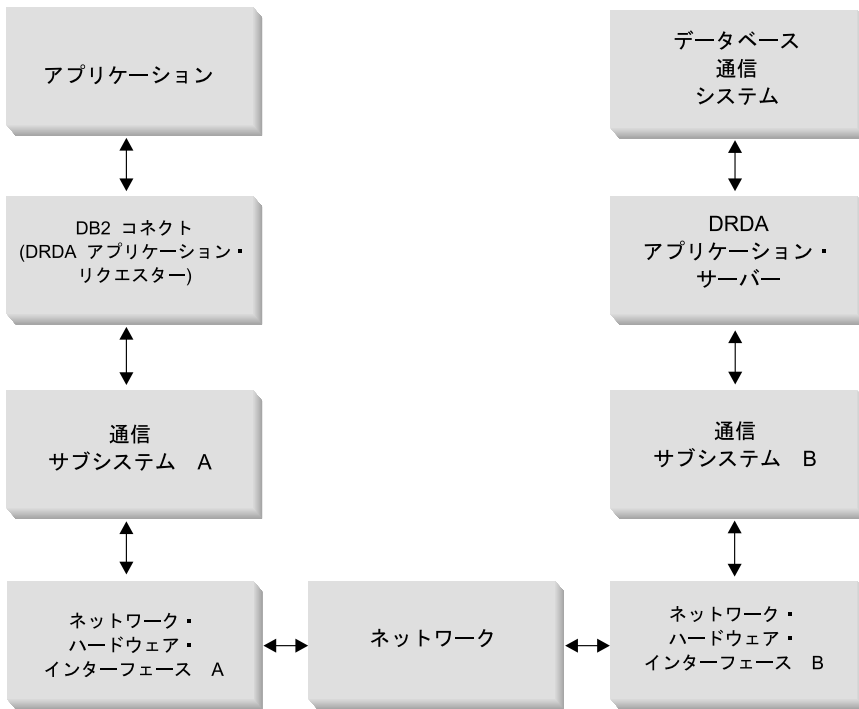


図7. DB2 コネクト内のデータ・フロー

- ホストまたは AS/400 データベースと、通信サブシステム B の部分は、通常同じシステム上で稼働します。このシステムは、1 つ以上の CPU、主記憶域、入出力サブシステム、DASD、およびオペレーティング・システムで成り立っています。他のプログラムはこれらの構成要素を共用しているので、リソースの競合によってパフォーマンス問題が引き起こされる可能性があります。
- ネットワークは、ケーブル、ハブ、通信回線、スイッチ、その他の通信制御装置の組み合わせで構成されています。たとえば、ネットワーク・ハードウェア・インターフェース B は、3745 または 3172 あるいは AS/400 版のトークンリング・アダプターのような通信制御装置になり得ます。ネットワーク・ハードウェア・インターフェース A と Bの間には複数の伝送媒体が関係することもあります。
- ネットワーク・ハードウェア・インターフェース A は、トークンリング、イーサネット (Ethernet\*\*), その他の LAN アダプター、または SDLC もしくは X.25 プロトコルをサポートするアダプターとすることができます。通

信サブシステム A は、IBM Communications Server for OS/2、Microsoft SNA Server、IBM SNA サーバー (AIX 版)、または SNAplus2 (HP-UX 用) とすることができます。

- DB2 コネクト製品と通信サブシステム A は、通常同じシステム上に位置付けられます。この章では、アプリケーションもまた、同じシステム上にあることを想定しています。

## ボトルネック

トランザクション・スループットは、システム内の最も遅い構成要素に依存します。パフォーマンスのボトルネックを識別できた場合は、構成パラメーターを変更したり、問題の構成要素にもっとリソースを割り振ったり、構成要素を更新したり、仕事の一部の負荷を軽減するため新規の構成要素を追加したりして、多くの場合問題を軽減することができます。

各種のツールを利用して、各構成要素の照会時間がどれだけかかったかを判別することができます。このようにして、パフォーマンスを改善するにはどの構成要素を調整またはアップグレードしたらよいかについて構想が得られます。たとえば、ある照会がその時間の 60% を DB2 コネクト・マシン内で使ったと判別した場合、おそらく DB2 コネクトを調整するか、または (リモート・クライアントがある場合は) ネットワークにもう 1 つ別の DB2 コネクト・マシンを追加することができます。

パフォーマンス・ツールの詳細については、146ページの『パフォーマンス・ツール』を参照してください。

## ベンチマーク

ベンチマークとは、1つの環境におけるパフォーマンスを別の環境におけるパフォーマンスと比較する仕方をいいます。

ベンチマークは、通常環境でテスト・アプリケーションを実行することから始めることができます。パフォーマンスの問題点が絞り込まれるにつれて、特殊化されたテスト・ケースが開発され、テストして観測する機能の範囲を限定していくことができます。

ベンチマークは、複雑にする必要はありません。特殊化されたテスト・ケースは、有効な情報を入手するために、必ずしもアプリケーション全体をエミュレートする必要はありません。単純な計測から始めて、保証が得られる場合だけ複雑性を増すようにします。

良いベンチマークの特性

- 各テストが再現できること。
- テストの各反復は同じシステム状態で開始されること。
- ベンチマーク用に使用されるハードウェアとソフトウェアは、ユーザーの実稼働環境と一致していること。
- 計測される機能またはアプリケーション以外には、活動状態にある他の機能またはアプリケーションがシステム内にないこと。ある量の他の活動がシステム内で実行中であることがシナリオに含まれている場合を除く。

**注:** 開始されたアプリケーションは、最小化またはアイドル時でもメモリーを使用します。このことは、ベンチマーク結果のページングまたはスキューの原因になり得ます。

## パフォーマンス・ツール

次の表は、システム・パフォーマンスを計測するのに役立つツールのいくつかをリストしています。これらのツールは、それ自体がシステムのリソースを使用しますので、それらを常時活動状態にしておくことは望ましいことではありません。

表7. パフォーマンス・ツール

システム	ツール	説明
<b>CPU およびメモリーの使用状況</b>		
AIX	vmstat、time、ps、tprof	DB2 コネクト ワークステーションおよびリモート・クライアントでの CPU またはメモリーの競合問題についての情報を提供します。
HP-UX	vmstat、time、ps、monitor および glance (利用可能な場合)	
OS/2	SPM/2、THESEUS/2、pstat	
Win NT および Windows 2000	MS パフォーマンス・モニター	
<b>データベースの活動</b>		
すべて	データベース・モニター	問題がデータベースから発しているかどうかを判別します。

表7. パフォーマンス・ツール (続き)

システム	ツール	説明
MVS または OS/390	DB2PM (IBM)、 OMEGAMON/DB2 (Candle)、TMON (Landmark)、INSIGHT (Goal Systems)、および DB2AM (BMC)	
Win NT および Windows 2000	MS パフォーマンス・ モニター	
<b>ネットワークの活動</b>		
AIX	netpmon	低水準のネットワーク統計 を報告します。それには秒 当りに受信するパケットま たはフレームの数のような TCP/IP および SNA 統計が 含まれます。
DOS または OS/2	トークンリング・ネットワ ーク 16/4 トレースおよびパ フォーマンス・プログラム	大部分のネットワーク・モ ニターはプラットフォーム に従属します。このツール はトークンリング専用で す。
ネットワーク制御装置、た とえば 3745	NetView パフォーマンス・ モニター	通信制御および VTAM の使 用効率を報告します。
OS/2	DatagLANce	パフォーマンスに関する データをユーザーへ図形的 に示すトレース・ツールで す。
UNIX 系	netstat	TCP/IP の通信量を取り扱 います。

## ODBC アクセスの最適化

DB2 ユニバーサル・データベースでは、ODBC を介した通信のパフォーマンスを改善するために設計された特別な最適化機能が準備されています。これらの拡張機能は、Microsoft Access、Lotus Approach、または Visual Basic で使用することができます。DB2 のクライアント構成アシスタント (CCA) を使用する際の ODBC スループットが高速になります。

最適化したODBC を活動化するには、以下の事柄を実行します。

新しい通信を定義する場合:

1. DB2 CCA を開始します。
2. 最適化するデータベース別名を選択します。
3. 「プロパティ (Properties)」押しボタンをクリックします。
4. 「このデータベースを ODBC 用に登録 (Register this Database for ODBC)」を選択します。
5. このデータベースを登録する方法を説明したラジオ・ボタンを選択します。
6. 「設定 (Settings)」押しボタンをクリックします。
7. 「最適化 (Optimize)」押しボタンをクリックして、ODBC 設定を最適化するアプリケーションを選択します。
8. 「OK」をクリックして、CCA を終了します。

既存の通信を更新する場合:

1. DB2 CCA を開始します。
2. 最適化するデータベース別名を選択します。
3. 「プロパティ (Properties)」押しボタンをクリックします。
4. 「設定 (Settings)」押しボタンをクリックします。
5. 「CLI/ODBC 設定 (CLI/ODBC Settings)」ウィンドウで「最適化 (Optimize)」押しボタンをクリックし、最適化するアプリケーションを選択します。
6. 「OK」をクリックして、CCA を終了します。

CCA について詳しくは、*DB2 コネクト概説およびインストール* を参照してください。ブラウザで以下の URL を表示すると、DB2 (OS/390 版) への ODBC アクセスの最適化に関する最新情報を見ることができます。

<http://www.ibm.com/software/data/db2/os390/odbcatl.html>

---

## アプリケーションの設計

アプリケーションを作成するとき、いくつかの方法でパフォーマンスを改善することができます。たとえば以下のものが含まれます。

- 複合 SQL およびストアド・プロシージャの使用
- グループ要求
- 必要なデータだけを要求するための述部論理の使用
- データ・ブロックの使用

- 可能な場合、静的 SQL の常時使用

## 複合 SQL およびストアード・プロシージャ

多くのコマンドと応答を受け渡しするアプリケーションでは、ネットワークのオーバーヘッドが重要になります。複合 SQL とストアード・プロシージャは、このオーバーヘッドを軽減する 2 つの方策です。

1 つのアプリケーションがいくつかの SQL ステートメントをプログラミング論理の介入なしに送信する場合、複合 SQL を使用することができます。プログラミング論理が SQL ステートメントのグループ内で必要な場合は、ストアード・プロシージャを使用することができます。

以下のものを除き実行可能なステートメントはすべて、複合 SQL ステートメント内に含めることができます。

```
CALL  
FETCH  
CLOSE  
OPEN  
Compound SQL  
Connect  
Prepare  
Release  
Describe  
Rollback  
Disconnect  
Set connection  
execute immediate
```

詳細については、*SQL 解説書* を参照してください。

複合 SQL をアプリケーション内で使用するときの情報については、61ページの『NOT ATOMIC 複合 SQL』を参照してください。複合 SQL をインポート・ユーティリティとともに使用するときの情報については、122ページの『インポートおよびエクスポート・ユーティリティの使用』を参照してください。

ストアード・プロシージャを使用すると、プログラム論理がサーバーに入れられるので、ネットワーク通信量を削減するのに役立ちます。バージョン 5.0 より前の DB2 では、ストアード・プロシージャで出力パラメーターを戻すことしかできなかったため、アプリケーションでコミット・コマンドを別に発行する必要がありました。したがって、ネットワーク通信が 2 回行われました。DB2 バージョン 5.0 以降では、プロシージャの終了時に自動的にコミットできます。また、結果セットを戻すこともできます。こうするとクライアントのアプリケーション論理を最小化できます。

ストアド・プロシージャの使用に関する情報については、57ページの『ストアド・プロシージャ』を参照してください。

## 要求のグループ化

関連する複数のデータベース要求 (SQL ステートメント) を 1 つのデータベース要求にグループ化すれば、ネットワークを通して伝送する要求と応答の数を減らすことができます。たとえば、以下のステートメントをグループ化して、

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1  
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=2
```

次のようにすると、

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1 OR ROW_ID=2
```

ネットワークを通して送られる要求の数が減ります。

また、IN および BETWEEN のようなキーワードを使用することにより、戻される行数を減らすことができます。さらに、UPDATE および DELETE ステートメントについて、WHERE、IN、および BETWEEN キーワードを使用することができます。

## 述部論理

必要な行および列だけを要求する場合に、述部論理を使用することができます。これは、ネットワーク通信量およびデータ伝送の CPU オーバーヘッドを最小化します。

たとえば、次の照会を使用しないようにします。

```
SELECT * FROM TABLEA
```

ROW\_ID=1 を持つ TABLEA の 1 番目の行だけが実際に必要な場合や、1 番目と 2 番目の列だけがが必要な場合には、上の照会を使用しません。

## データのブロック化

サーバーからの大量のデータが予想される場合は、データ・ブロックを使用します。このブロック化によって、ネットワーク帯域幅の使用は改善され、ホストまたは AS/400 データベース・サーバーと DB2 コネクト・ワークステーションの両方の CPU オーバーヘッドが減少します。

サイズに関係なく、送受信される各メッセージについて一定量の CPU とネットワークのオーバーヘッドがかかります。データ・ブロックは、同じ量のデータ転送に必要とされるメッセージの数を減らします。



ブロック化を使用すると、照会からのデータの 1 番目の行は、1 番目のブロックが受け取られるまではアプリケーションに送達されません。ブロック化は、1 番目の行を探す検索時間を増加させますが、その後続く行については検索時間を短縮できます。

別の考慮事項は、使用される記憶容量です。メモリー作業セットは、ブロック化がオンになると通常は増加します。SNA 使用時のブロック化に関する十分な説明については、分散関係データベース体系 接続の手引き を参照してください。

DB2 コネクト内では、153ページの『RQRIOBLK』で説明するとおり、各ブロック内で転送されるデータの量を制御することができます。

ブロック化を呼び出すには、`prep` または `bind` コマンドの `BLOCKING` オプションを使用します。(詳細については、107ページの『`BIND` コマンド』をご覧ください。) ブロック化は、次の場合にオンになります。

- カーソルが読み取り専用になっているとき、または
- カーソルが未確定で、ブロック化が `prep` または `bind` コマンドの間指定されているとき

読み取り専用、更新可能、未確定カーソルの定義については、アプリケーション開発の手引き を参照してください。

**注:** 動的 SQL を使用している場合は、カーソルは常に未確定です。

### **BLOCKING を伴う SQL ステートメント**

更新可能な `SELECT` ステートメント (`UPDATE/DELETE WHERE CURRENT OF` ステートメントを使用する) は、非ブロック化の照会です。したがって、絶対に必要なときだけそれを使ってください。

更新可能な `SELECT` は、`SELECT` が完了した時と `UPDATE/DELETE` が発行される時との間にその行が決して変更されないようにします。このレベルの並行性がアプリケーションにとって重要でない場合は、別の方法として、非更新可能な `SELECT` から戻される値に基づく探索基準を用いて、`DELETE` または `UPDATE` を使用します。

読み取り専用の `SELECT` については、`FOR FETCH ONLY` を指定します (VM および VSE の場合を除きます。この場合、サポートされていません)。

## 静的 SQL と動的 SQL

静的 SQL をできるだけ使用してください。それは、実行時 SQL セクション 準備および未確定カーソルを回避します。動的 SQL の使用が避けられない場合は、ネットワーク通信量を最小にしてパフォーマンスを改善するために、以下のことを行うことができます。

- ステートメントが SELECT であり、準備の必要がある場合は、`PREPARE ... INTO SQLDA` を実行します。SQLDA は、ユーザーの設定に必要なフル・サイズに割り振ってください。列の最大数が  $x$  であり、それ以内にとどまると予想される場合は、SQLDA に  $x$  個の SQLVAR を割り振ります。潜在的な列数が不確実な場合（そしてメモリーが問題にならない場合）は、SQLVAR の最大数 (256) を使用します。

SQLDA の割り振りが、戻り SQLDA を保管するのに十分な大きさでない場合は、プログラムは、その結果を再度保管するのに十分な大きさの SQLDA を持つもう一つの DESCRIBE を発行する必要があります。このことは、ネットワーク通信量を増加させます。

PREPARE そして DESCRIBE という順序を使用しないでください。

PREPARE.....INTO ステートメントを使用すると、パフォーマンスは向上します。

- 静的にバインドされた SQL COMMIT または ROLLBACK ステートメントを、動的 COMMIT または ROLLBACK ステートメントの代わりに実行します。
- それが SELECT、COMMIT、または ROLLBACK ステートメントでない場合は、PREPARE および EXECUTE の順序の代わりに、EXECUTE IMMEDIATE を発行してステートメントを実行します。
- ODBC アプリケーションは動的 SQL を使用します。CLI/ODBC 静的プロファイル作成機能を使用すれば、パフォーマンスを改善することができます。この機能を使用すると、ODBC 呼び出しを取り込んで、データベース・パッケージに保管された静的ステートメントに変換することができます。実際に得られるパフォーマンスは、アプリケーションがどの程度複雑であるかによって決まります。詳細については、コール・レベル・インターフェースの手引きおよび解説書 を参照してください。

## その他の SQL 考慮事項

コマンド行プロセッサを使用すると、一般に、プログラム内に動的 SQL を有する場合より動作が遅くなります。なぜならコマンド行プロセッサは、SQL をデータベース・エンジンへ発信する前に入力を構文解析する必要があるからです。また、コマンド行プロセッサは、データを受け取った時にそれを形式化しますが、アプリケーションにとっては不必要なことです。

インタープリター言語 (たとえば REXX) による SQL ステートメントは、コンパイル言語 (たとえば C 言語) による同じ SQL ステートメントよりかなり処理が遅くなります。

CONNECT ステートメントについては、タイプ 1 およびタイプ 2 と呼ばれる 2 つのタイプがあります。タイプ 2 の接続を使用してデータベースへ接続した場合は、以前の接続を休止状態にしますが、ドロップはしません。その後で休止状態の接続に切り換えれば、ライブラリーのロードおよび内部データ構造のセットアップのオーバーヘッドを避けることができます。この理由から、タイプ 2 の接続を使用すれば、複数のデータベースにアクセスするアプリケーションについてはパフォーマンスを改善することができます。タイプ 2 接続の詳細については、[管理の手引き](#) および [SQL 解説書](#) を参照してください。

---

## DB2 コネクットのチューニング

データベース・マネージャー構成ファイルの各種パラメーターは、DB2 コネクトをチューニングするために使用することができます。これらパラメーターの変更の詳細については、[管理の手引き](#) を参照してください。

### RQRIOBLK

RQRIOBLK パラメーターは、ネットワーク入出力ブロックの最大サイズを設定します。ブロック・サイズを大きくすれば、大きな要求のパフォーマンスを改善することができます。ブロック・サイズは、通常、単一行のデータを求める要求のような、小さな要求の応答時間には影響を与えません。

通常はブロック・サイズを大きくすると、DB2 コネクト・ワークステーション上で、より大きなメモリーが必要になります。このことは、作業セットのサイズを大きくし、小さなワークステーションで大量のページングが生じる原因となります。

アプリケーションの実行に際してそれほど大きなページングを起こさない場合は、省略時の DRDA ブロック・サイズ (32767) を使用します。その他の場合は、ページングがなくなるまで入出力ブロック・サイズを引き下げます。いったんページングが始まると、パフォーマンスの著しい低下が起こります。パフォーマンス・モニター・ツール (たとえば、UNIX ベースのシステム用の vmstat ツールや OS/2 用の SPM/2) を使用して、システム上でページングが起きているかどうかを判別してください。その他のツールについては、146ページの『パフォーマンス・ツール』を参照してください。

## DIR\_CACHE

DIR\_CACHE パラメーターにより、ディレクトリー情報がキャッシュされるかどうかが決まります。キャッシュを使用する場合 (DIR\_CACHE=YES)、ディレクトリー・ファイルが読み込まれてメモリーにキャッシュされます。これによって、内部ディレクトリー構造が作成され、接続が確立されるたびにディレクトリー・ファイルを読み取るというオーバーヘッドが最小化されます。

キャッシュを使用しない場合 (DIR\_CACHE=NO)、データベースへの接続が行われるたびに、該当するディレクトリーがディスクから読み取られ、次に検索が行われます。要求項目が見つかった後、ディレクトリー検索に関係したすべてのメモリーが解放されます。

キャッシュを使用すると、共有ディレクトリー・キャッシュが **db2start** プロセスの間に作成され、DB2 が停止する時に解放されます。このキャッシュは、すべての DB2 サーバー・プロセス(db2agent) によって使用されます。また、アプリケーションがデータベースへのその最初の接続命令を発行した時、私用アプリケーション・ディレクトリー・キャッシュが作成され、アプリケーションが終了した時に解放されます。

各キャッシュは、システム・データベース・ディレクトリー、データベース接続サービス・ディレクトリー、およびノード・ディレクトリーのイメージを提供します。キャッシュは、ディレクトリー・ファイル入出力を削減し、ディレクトリー検索を最小化することにより、接続コストを引き下げます。

キャッシュ・ディレクトリーを更新した場合、その変更は即時にはキャッシュへ伝搬されません。ディレクトリー項目がキャッシュ内に見つからない場合は、元のディレクトリーが検索されます。

キャッシュによって、アプリケーションが存続する間必要とされる私用メモリーが大きくなります。キャッシュを使用しなければ、このメモリーはディレクトリー参照処理の間だけ必要になります。DB2 による共有メモリーの全般的な使用は少し増加します。データベース・エージェントの間で共有されるディレクトリー情報は共有メモリーへ移動するからです。キャッシュに必要なメモリーのサイズは、各ディレクトリー内で定義される項目の数によります。

## 他の DB2 コネクト・パラメーター

MAXDARI および NUMDB は、DB2 コネクト・ワークステーションにローカル・データベースがない場合は、最小値に設定してください。このように設定すれば、リソースの消費は最小化されます。

AGENTPRI は、リモート・クライアントを使用する場合だけ適用されます。AGENTPRI は、オペレーティング・システム・スケジューラーによって DB2 コネクト・インスタンスのエージェントに与えられる優先順位を制御します。DB2 コネクト・インスタンスは、それが高い優先順位 (小さい番号) を持っている場合は、より大きな CPU サイクルが与えられます。これにより、DB2 コネクト・ワークステーションで実行しているその他のプロセスに対しては CPU サイクル数が減ります。たとえば、同じワークステーション上で異なる AGENTPRI を持つ高優先順位の DB2 コネクト・インスタンスと低優先順位の DB2 コネクト・インスタンスを持つことができます。

DB2 コネクトを介してクライアント・マシンからホストまたは AS/400 データベース・サーバーへつながる各接続はすべて、DB2 コネクト・ワークステーション上で稼働するエージェントを必要とします。MAXAGENTS については、DB2 コネクト・ワークステーションを介してホストまたは AS/400 データベース・サーバーにアクセスするリモート・クライアント接続のピーク数以上の値に設定してください。

会計ストリングの使用を決定した場合、sqlsact() API を使用すると、DB2ACCOUNT 環境変数方式よりもパフォーマンスについて有利になります。詳細については、64ページの『DB2 ユニバーサル・データベース (OS/390 版)でのチャージバック会計の実装』を参照してください。

調整済みの SQLCODE マッピング・ファイルを必要としない場合は、省略時の SQLCODE マッピングを使用するか、または SQLCODE マッピングをオフにすることによって、パフォーマンスを改善することができます。(省略時のマッピング・ファイルは、DB2 コネクト・ライブラリーに組み込まれています。調整済みのマッピング・ファイルはディスクから読み取る必要がありますが、これがパフォーマンスに影響を与えます。) SQLCODE マッピングの詳細については、137ページの『第11章 SQLCODE マッピング』を参照してください。

---

## 接続のプール

DB2 コネクト エンタープライズ・エディション・サーバーは、しばしば、同時に行われる何千ものクライアント要求に対するデータベース接続を提供します。データベース・サーバーへの接続の確立と切断は、リソースを集中的に使用するプロセスのため、データベース・サーバーと DB2 コネクト・サーバーの両方のパフォーマンスに悪影響を及ぼす場合があります。このことは、Web ページにアクセスするたびにデータベース・サーバーへの新規接続を行い、照会を実行してから接続を終了する必要のある Web 環境で顕著に見られます。

このオーバーヘッドを減らすため、DB2 コネクト エンタープライズ・エディションは、接続プールを使用して、即座にアクセス可能なプールでデータベースへのオープン接続を維持します。

## 接続プールの働き

接続プールの存在は、DB2 コネクトを介してホストに接続するアプリケーションからは認識されません。アプリケーションでホストからの切断が要求されると、DB2 コネクトではアプリケーションとのインバウンド接続は切断されますが、ホストとのアウトバウンド接続はプール内に保持されます。新しいアプリケーションが接続を要求すると、DB2 コネクトは既存のプールからの接続を使用します。すでに存在している接続を使用すると、全体の接続時間が短縮されるだけでなく、ホストでの高い CPU 接続コストも削減されます。

接続プールを使用するには、次の APAR が DB2 (OS/390 版) バージョン 6.1 に適用されている必要があります。

APAR PQ33473

DB2 コネクト・エージェントは、アイドルまたはアクティブの 2 つの状態のいずれかになっています。エージェントがアプリケーションの作業を実行している場合、そのエージェントはアクティブです。この作業が完了すると、エージェントはアイドル状態になり、同じアプリケーションまたは別のアプリケーションからの次の作業を待ちます。すべてのアイドル・エージェントは、アイドル・エージェント・プールとして知られている場所に一緒に保持されます。このプールのサイズは、NUM\_POOLAGENTS 構成パラメーターを使用して構成できます。このパラメーターは、システムが保守するアイドル・エージェントの最大数と同じです。このパラメーターをゼロに設定すると、接続プール機能はオフになります。

DB2 コネクトは、最初のクライアント要求を受け取る前に、データベースへの接続を確立することはありません。しかし、望むなら、クライアントが要求を出す前に、アイドル・エージェントのプールを満たすことができます。

NUM\_INITAGENTS 構成パラメーターを使用すると、開始時にプールを満たすことができます。このパラメーターは、始動時に作成されるアイドル・エージェントの数を決定します。これらのアイドル・エージェントが、ホスト・データベース・サーバーに最初に接続することはありません。

クライアントがホストへの接続を要求すると、DB2 コネクトはホスト・データベース・サーバーに接続しているプールの中からエージェントを取得しよう

とします。それが失敗すると、アイドル・プールで使用可能なエージェントを検索します。プールが空の場合、DB2 コネクトは新しいエージェントを作成します。

MAX\_COORDAGENTS 構成パラメーターを使用して、同時にアクティブにできるエージェントの最大数を制御することができます。この数字を超えると、新しい接続はエラー SQL コード SQL1226 を発行して失敗します。(このコードは、同時に行うアウトバウンド接続の最大数を超過したことを意味します。)

DB2 登録変数 DB2CONNECT\_IN\_APP\_PROCESS を使用すると、DB2 コネクト EE と同じマシンで稼働しているアプリケーションが、アプリケーション・プロセス内で DB2 コネクトを実行するか(省略時の動作)、アプリケーションが DB2 コネクト EE サーバーに接続してから、エージェント内でホスト接続を実行するかのいずれかを行うことができます。アプリケーションが接続プールを使用するには、DB2 コネクト EE サーバーのエージェント内からホストに接続し、DB2CONNECT\_IN\_APP\_PROCESS を NO に設定する必要があります。

## DB2 コネクトの接続コンセントレーター

DB2 コネクトの接続コンセントレーター 技術を使用すると、DB2 コネクト エンタープライズ・エディション・サーバーは、商取引を行う何千人ものユーザーをサポートすると共に、S/390 ホストまたは AS/400 データベース・サーバーで必要とされるリソースを大幅に削減することができます。この技術は、すべてのアプリケーションからの作業負荷を、より少ない数の S/390 ホスト接続または AS/400 データベース・サーバー接続に集中することにより、このことを成し遂げます。これは前述の接続プール機能とよく似ているように思われるかもしれませんが、実際には非常にボリュームの大きい OLTP (オンライン・トランザクション処理) アプリケーションのリソース使用量を減らすためのさらに洗練された方法です。

接続プールは、アプリケーションが終了して接続が必要なくなるときに、接続を確立するのに必要なコストを節約します。言い換えると、プールした接続を別のアプリケーションが再使用するには、その前にアプリケーションが接続を切断する必要があります。

一方、接続コンセントレーターを使用すると、DB2 コネクトはアプリケーションがトランザクションを終了するとすぐ、別のアプリケーションで利用可能にすることができます。このとき、そのアプリケーションは接続を切断する必要はありません。本来、データベース・サーバー接続とそれに関連付けられた

ホストおよび DB2コネクトのリソースがアプリケーションで使用されるのは、アクティブなトランザクションがある場合だけです。トランザクションが完了するとすぐ、接続とそれに関連付けられているリソースは、トランザクションを実行する必要のある他のアプリケーションで使用できるようになります。

### 接続コンセントレーターの実装方法

DB2 コネクトの以前のバージョンでは、すべてのアクティブ・アプリケーションに、データベース接続に加えてアプリケーション要求を管理するエンジン・ディスパッチ可能単位 (EDU) がありました。この EDU は通常、調整エージェントと呼ばれていました。それぞれの調整エージェントは、アプリケーションと EDU の状態またはコンテキストを追跡しました。各 EDU は、接続数の増加時に相当量のメモリーを必要とするため、エージェント間でのコンテキスト切り替えではさらにオーバーヘッドが増えてしまいます。

上記のアーキテクチャーでは、接続と EDU は 1 対 1 の関係にあります。しかし、接続コンセントレーターを使用すると、接続と EDU の関係を複数対 1 にすることができます。つまり、接続 (X) と EDU (Y) の関係は  $X \geq Y$  になります。

接続コンセントレーターは、エージェントを 2 つのエンティティー (論理エージェントと作業エージェント) に分割します。論理エージェントはアプリケーションを表しますが、特定の EDU を参照することはありません。論理エージェントには、アプリケーションが必要とするすべての情報と制御ブロックが含まれています。n 個のアプリケーションがサーバーに接続している場合、そのサーバーには n 個の論理エージェントがあります。作業エージェントは、アプリケーションの要求を実行する物理 EDU ですが、指定したアプリケーションへの永久接続は持ちません。作業エージェントは論理エージェントと連携して、トランザクションを実行します。それから、その連携をトランザクション境界で終了し、使用可能なプールに戻ります。

論理エージェント・スケジューラーとして知られるエンティティーが、作業エージェントを論理エージェントに割り当てます。特定のコンピューティング・プラットフォームで開くことができるファイル・ハンドルの数が制限されている場合、論理エージェントの数がファイル・ハンドルの限度を超えたときにスケジューラー・インスタンスが複数になる場合があります。

### コンセントレーターの活動化

接続コンセントレーターを使用するには、次の APAR が DB2 (OS/390 版) バージョン 6.1 に適用されている必要があります。

APAR PQ33473



データベース・マネージャー構成パラメーター `MAX_LOGICAGENTS` は、論理エージェントの最大数を設定します。 `MAX_LOGICAGENTS` の値を省略時値よりも大きい任意の値に設定することにより、コンセントレーター機能をアクティブにすることができます。 `MAX_LOGICAGENTS` の省略時値は、 `MAX_COORDAGENTS` の省略時値と同じです。アプリケーションごとに 1 つの論理エージェントがあるため、 `MAX_LOGICAGENTS` は実際にはデータベース・インスタンスに接続できるアプリケーションの数を制御し、 `MAX_COORDAGENTS` は同時にアクティブになれるインバウンド接続の数を制御します。 `MAX_LOGICAGENTS` は、 `MAX_COORDAGENTS` ~ 64,000 までの範囲の数値を取ります。省略時の論理エージェントの数は、 `MAX_COORDAGENTS` と同じです。

既存の構成パラメーターの中にも、エージェントを構成するために使われるものがあります。それには、以下のパラメーターが含まれます。

### **MAXAGENTS**

作業エージェントの最大数。

### **MAX\_COORDAGENTS**

アクティブな調整エージェントの最大数。

### **NUM\_POOLAGENTS**

エージェント・プールのサイズ。エージェント・プールには、アクティブでないエージェントやアイドル状態のエージェントが含まれています。

### **NUM\_INITAGENTS**

プール内の作業エージェントの初期数。これらはアイドル状態のエージェントです。

### **XA トランザクション・サポート**

接続コンセントレーターのアーキテクチャーを使用すると、DB2 コネクトは DB2 (OS/390 版) および DB2 (AS/400 版) と密接な関係にある XA トランザクション・サポートを提供することができます。コンセントレーターは、他のすべてのトランザクションの場合と同じように、作業エージェントを特定の XA トランザクション (単一の XID) に関連付けます。しかし、XA トランザクションが `xa_end()` (分岐境界) によって終了する場合、作業エージェントが汎用プールに解放されることはありません。作業エージェントはその XA トランザクションに関連付けられたままです。別のアプリケーションが同じ XA トランザクションと結合すると、作業エージェントはそのアプリケーションに関連付けられます。

トランザクション境界を呼び出すと、エージェントはプールに戻されます。たとえば、`xa_prepare()` (読み取り専用)、`xa_rollback()`、`xa_recover()`、`xa_forget()`、`xa_commit()`、またはロールバックを引き起こすすべての XA エラーは、エージェントを通常のプールに戻します。`xa_end()` が終わらせるのはトランザクションの分岐だけです。これは `XID` との関連付けを終わらせるには不十分です。

## 例

1. 4,000 以上の同時接続を必要とする環境について考えてみます。CGI アプリケーションを使用する Web サーバー、または多くのデスクトップ・ユーザーが存在するオフィス・システムでは、両方ともこの要件を超えてしまう可能性があります。こういう場合、効率的な処理には、通常は DB2 コネクトがスタンドアロン・ゲートウェイとして動作することが求められます。すなわち、データベースと DB2 コネクトを別々のマシンに置く必要があります。

DB2 コネクト・サーバー・システムは、データベース・マシンに対する 4,000 もの同時に行われるオープン接続を維持できない場合があります。たいていの場合、特定の瞬間に生じるトランザクション数は、同時接続の数よりもかなり小さくなります。そのため、システム管理者は、データベース構成パラメーターを以下のように設定することにより、システムの効率を最大にすることができます。

```
MAX_LOGICAGENTS = 4,000
MAX_AGENTS       = 1,000
MAX_COORDAGENTS  = 1,000
NUM_POOLAGENTS   = 1,000
```

ゲートウェイが同時に処理しているトランザクション数が 1,000 しかない場合でも、コンセントレーターは最大 4,000 の並行セッションをオープンし続けます。

2. 上記の例では、作業エージェントと論理エージェントの関連付けは、常に形成されたり解除されたりしています。アイドル状態でないそれらのエージェントは、データベースへの接続は維持していますが、特定のトランザクションに関与してはいません。そのため、接続を要求する任意の論理エージェント (アプリケーション) が使用することができます。

XA トランザクションの場合は、いくらか異なっています。この例では、DB2 コネクト・ゲートウェイと OS/390 または AS/400 データベースで TP モニターを使用していることを想定しています。アプリケーションが接続を要求すると、コンセントレーターは、その要求に応じるためアクティブでないエージェントをアクティブにするか、新しい作業エージェントを作成

します。アプリケーションが XA トランザクションを要求するものとします。このトランザクションに応じて XID が作成され、作業エージェントがそれに関連付けられます。

アプリケーションの要求が処理されると、`xa_end()` が発行され、作業エージェントから切り離されます。作業エージェントと、トランザクションの XID との関連付けは残ります。このとき、このエージェントは、関連付けられている XID を持つトランザクション要求にのみ応じることができます。

この時点で、別のアプリケーションが非 XA トランザクションを要求する場合があります。他に使用可能な作業エージェントがない場合でも、XID に関連付けられたエージェントは 2 番目のアプリケーションで使用可能にされることはありません。それは、アクティブであるとみなされます。2 番目のアプリケーション用には、新しい作業エージェントが作成されます。2 番目のアプリケーションがトランザクションを終了すると、そのアプリケーションの作業エージェントは使用可能なプールに解放されます。

一方、最初のエージェントの XID に関連付けられたトランザクションを要求している他のアプリケーションが、そのエージェントに接続したりエージェントから切断されたりする場合は、そのエージェント専用の XA トランザクションが実行されます。そのトランザクションを要求するアプリケーションはすべて、この作業エージェント (空き状態であれば) に送信されません。

作業エージェントは、アプリケーションがトランザクション境界呼び出し (`xa_end()` ではない) を発行するまでは、汎用プールに解放されることはありません。たとえば、アプリケーションが `xa_commit()` でトランザクションを終了する場合、その時点で作業エージェントは XID との関連付けを解除し、使用可能なプールに戻ります。この時点で、要求元のアプリケーションはすべて、別の XA トランザクションか非 XA トランザクションのいずれかの作業エージェントを使用することができます。

### 制約事項

ゲートウェイ・コンセントレーターの使用については、重要な制約事項がいくつかあります。システムで接続コンセントレーターの使用を試みる前に、以下の情報をすべて検討してください。

- 接続コンセントレーターは、DB2 バージョン 7 以降のクライアントでのみ使用できます。
- コンセントレーターがサポートするのは、DB2 (OS/390 版) または DB2 (AS/400 版) ホストだけです。

- コンセントレーターは、ローカル・クライアントからリモート・クライアントへのインバウンド接続を確立するに際し、TCP/IP プロトコルに依存します。TCP/IP またはローカル (IPC) を使用するインバウンド接続だけが、プールのアウトバウンド接続を利用することができます。コンセントレーターは、SNA などの他の通信プロトコルを経由した接続を受け入れますが、その接続で XA 集中機能を使用することはできません。
- コンセントレーターがゲートウェイで使用可能にされている場合には、クライアント・アプリケーションで静的 SET ステートメントは使用しないでください。静的 SET を使用しても DB2 がエラーを戻すことはありませんが、ご使用のアプリケーションや同じアウトバウンド接続を共用する他のアプリケーションに悪影響を及ぼす場合があります。
- SET ステートメントでは、即時実行のみがサポートされています。
- グローバル一時表を宣言する場合、その表はトランザクションまたは分岐境界で明示的に閉じる必要があります。表を閉じ忘れると、後でトランザクション中にエラーが生じる場合があります。
- XA と密接に結びついたトランザクション・サポートの場合、同じ XA トランザクションに関与するアプリケーションはすべて、ホストに接続するために同じゲートウェイを使用する必要があります。
- 保留カーソルのトランザクション境界を閉じるアプリケーションだけが、コンセントレーターの恩恵にあずかることができます。保留カーソルを閉じないトランザクションは、依然として処理はされますが、専用の作業エージェントが割り当てられるため、コンセントレーターの全機能セットを使用することはできません。
- 同じ XA トランザクションに関与するすべてのアプリケーションに、同じ CCSID が必要です。また、それは同じユーザー ID を使用して接続する必要があります。
- 2 フェーズ接続をサポートするためにアウトバウンド接続が確立された場合、その接続のエージェントは 2 フェーズ接続をサポートするためのみ使用することができます。同様に、1 フェーズ接続をサポートするために確立されたエージェントは、1 フェーズ接続だけをサポートします。
- 認証タイプ DCS\_ENCRYPT は、バージョン 7 のコンセントレーターでは動作しません。
- コンセントレーターがサポートするのは、コマンド行インターフェースからの動的 SQL だけです。組み込み動的 SQL アプリケーションからの動的準備要求は拒否されます。静的 SQL を使用するか、動的 SQL ステートメント用の CLI を使用するよう、アプリケーションを変更する必要があります。

## データベースのチューニング

システム・パフォーマンスは、ホストまたは AS/400 データベース・サーバーのデータベースのパフォーマンスによって影響を受けます。

それぞれのデータベース管理システムによって、異なるパフォーマンス機能が備わっています。各種システムの SQL 最適化プログラムは、たとえば、同じアプリケーションを使用しても異なる挙動をとることがあり得ます。詳細については、ホストまたは AS/400 データベース・サーバーのシステム・パフォーマンス資料を参照してください。

DB2 ユニバーサル・データベース (AS/400 版) 用については、非コミット読み取り (UR) またはジャーナルを避けるための「コミットなし」(NC) バインド・オプションを使用することによって、パフォーマンスを改善することができます。

**注:** UR を使用する場合、ジャーナルされていないデータは、読み取りはされますが更新されません (これは、ブロッキングを ALL に設定している場合です)。

アプリケーション・サーバーおよびそれが提供するロックの細分性により、照会またはアプリケーションに使用される分離レベルは、パフォーマンスに有効な影響を与えることがあります。

データベースは、適切なレベルの正規化、索引の効果的な使用、およびデータベース・スペースの適切な割り振りを行う必要があります。また、パフォーマンスは、以下のセクションで説明するとおり、使用するデータ・タイプによって影響を受けます。

### DB2 (OS/390 版) のチューニング

OS/390 V1R3 は、TCP/IP サポートの最小要件です。OS/390 V2R5 以降の使用を強くお勧めします。

分散アプリケーションを DB2 (OS/390 版) に接続する処理は、分散データ機能 (DDF) によって行われます。DDF はアプリケーション・サーバーとしてセットアップする必要があります。これを行うには、リモート・システムの LU 名を SYSIBM.LUNAMES 表に挿入するか、または LUNAME、SYSMODENAME、USERSECURITY、ENCRYPTPSWDS、MODESELECT、および USERNAMES 値を SYSIBM.SYSLUNAME 表に挿入します。続いてブートストラップ・データ・セット (BSDS) に対して DDF 更新を実行します。たとえば、次のようにします。

```
DDF LOCATION=LOC1,LUNAME=LU1,PORT=8000,RESPORT=8001
```

最高のパフォーマンスを得るには、推奨されている DDF アドレス空間の優先順位を使用する必要があります (COMPAT モードの場合は DBM1 と同じか多少小さい値)。VLF 内では許可の RACF キャッシングを使用してください。また、使用できる場合は V5 パッケージ許可のキャッシングを使用してください。ほとんどの操作では CACHEPAC=32768 の値で十分です。

DDF では VTAM に対する接続が試行されるので、DDF を開始する際には VTAM をアクティブにしていなければなりません。VTAM APPL 定義の例を以下に示します。

SYD51TC* APPL AUTH=(ACQ),	X
PARSESS=YES,	X
HAVAIL=YES,	X
EAS=1600,	X
APPC=YES,	X
DSESLIM=1024,	X
DMINWNL=512,	X
DMINWNR=512,	X
AUTOSES=1,	X
SECACPT=ALREADYV,	X
SRBEXIT=YES,	X
SYNCLVL=SYNCPT,	X
MODETAB=DB2MODET,	X
VPACING=63	X

OS/390 では非アクティブなスレッド処理を最適化できます。V3 では最大 10,000 までクライアントを並行接続でき、V4 と V5 では最大 25,000 まで並行接続できます。ただし、いずれの場合も並行してアクティブにできる最大数は 1999 です。個々のワークステーション・クライアントは非アクティブになっても接続したままにしておくことができます。そのスレッドは、コミットのたびに非アクティブ・チェーンに組み込まれます。

DSNZPARM パラメーターの CMTSTAT、CONDBAT、および MAXDBAT はスレッド処理に影響を与えません。最高のパフォーマンスを得るには、CMTSTAT を INACTIVE に設定し、良好なパフォーマンスが得られる DBAT の最大接続数に CONDBAT を調整し、受け入れられるアクティブな DBAT の最大数に MAXDBAT を調整してください。

VTAM 構成を含めて、DRDA ネットワーク内で DB2 (OS/390 版) を接続することに関する完全な説明については、オンラインのコネクティビティー 補足を参照してください。

## データ変換

データを 1 つの環境から別の環境へ転送する時、それを変換する必要があり得ます。この変換はパフォーマンスに影響を与えることがあります。

次のプラットフォームについて考慮してみます。

- Intel (OS/2、Windows NT、または Windows 2000)
- IEEE (UNIX ベースのシステム)
- System/370 および System/390 (MVS、OS/390、VM、および VSE)
- OS/400

そして、以下の数値データ・タイプについて考慮してみます。

- パック 10 進数
- ゾーン 10 進数
- 整数
- 浮動小数点数

表8 は、いつ変換が行われるかを示します。

表 8. データ変換

	Intel	IEEE	S/370 & S/390	OS/400
<b>パック 10 進数データ</b>				
Intel	No	No	No	No
IEEE	No	No	No	No
S/370/390	No	No	No	No
OS/400	No	No	No	No
<b>ゾーン 10 進数データ</b>				
Intel	No	No	Yes	Yes
IEEE	No	No	Yes	Yes
S/370/390	Yes	Yes	No	No
OS/400	Yes	Yes	No	No
<b>整数データ</b>				
Intel	No	Yes	Yes	Yes
IEEE	Yes	No	No	No
S/370/390	Yes	No	No	No
OS/400	Yes	No	No	No
<b>浮動小数点データ</b>				
Intel	No	Yes	Yes	Yes
IEEE	Yes	No	Yes	No
S/370/390	Yes	Yes	No	Yes
OS/400	Yes	No	Yes	No

1 バイト文字のデータ変換の CPU コストは、数値データの変換コストよりも一般に小さいといえます (データ変換が必要な場合)。

DATE/TIME/TIMESTAMP のデータ変換コストは、1 バイト CHAR の場合とほぼ同じです。浮動小数点データの変換コストが最大です。アプリケーション設計者は、DB2 コネクト・ベースのアプリケーションを設計するときは、これらの事実の利点を取り入れることができます。

データベース表が 'FOR BIT DATA' と定義される列を持っている場合、アプリケーションとデータベースとの間で転送される文字データはデータ変換をなら必要としません。このことは、ホストまたは AS/400 データベース・サーバー上でデータを保存するときに利用することができます。

### 文字データのデータ・タイプ

文字データは、CHAR または VARCHAR のどちらかのデータ・タイプを持つことができます。どのデータ・タイプがより効率的かは、フィールド内のデータの代表的な長さによります。

- 実際のデータ・サイズがかなり変化する場合は、VARCHAR の方が、より効率的といえます。というのは、CHAR の場合はフィールドを埋めるために余分のブランク文字を追加する必要があるからです。これらのブランク文字は、他の文字と同様、ネットワークを通して伝送される必要があります。
- 実際のデータのサイズがそれほど変化しない場合は、CHAR の方がより効率的です。なぜなら各 VARCHAR フィールドには、長さ情報の数バイトがあり、それも伝送しなければならないからです。

## ネットワークのチューニング

分散データベース環境で全体のパフォーマンスを向上させるには、ネットワークからの遅延をなくすことが最善の方法です。一般に、ネットワーク管理者にとって効率的なネットワークとは、伝送と伝送の間にできる限り多くのデータが収集されるネットワークです。この考え方は、分散データベースなどのアプリケーションには当てはまりません。この種のアプリケーションはネットワーク内で遅延が発生するからです。エンド・ユーザーにはネットワークの効率がよいとは思えず、遅延していることだけが分かります。

ほとんどのネットワーク装置には遅延パラメーターがありますが、その大部分の省略時値は分散データベースにとっては非常に不適切なものです。パフォーマンスを改善するには、この種のパラメーターを見付けて、可能であればゼロに設定する必要があります。また、装置のバッファ・サイズが十分で、データが脱落して再送が行われたりしないことを確認する必要があります。たとえば、UNIX システムでは送信または受信待ち行列の深さの省略時値は通常 32で



す。パフォーマンスを向上させるには、待ち行列の深さを 150 に設定してください。DLC の設定でこれに対応するパラメーターは Receive Depth で、この値も 150 に設定する必要があります。

ほとんどの場合、IOBUF パラメーターは低過ぎる値に設定されています。この値は普通は 500 に設定されていますが、経験が示すところによると、特に ESCON や 3172 などのチャンネル接続の場合に、大量のデータを移動するには 3992 の値が最適です。

SNA 接続の場合、ワークステーション・ソフトウェアのモード・プロファイルを 63 に設定する必要があります。一般的には、ネットワーク全体の受信ページングの値を最大値に設定する必要があるため、DB2 APPL ステートメントの VPACING パラメーターと PACING パラメーター、および交換回線大ノードのワークステーションの PU/LU も 63 に設定する必要があります。このように設定すると、送信側が応答を待機する状態になるまでに流れるメッセージの量を、継続的に増やすことができます。

LAN システムでは、DLC や LLC の送信ウィンドウや受信ウィンドウのサイズによりパフォーマンスはかなり左右されます。送信値は 7 以上に設定する必要があります。また、ほとんどの構成では受信値を 4 以下に設定するのが最適です。

イーサネットを実行している場合は、TCP セグメントのサイズを 1500 バイトに設定する必要があります。トークンリングや FDDI ネットワークの場合はこの値を 4400 バイトにする必要があります。TCP/IP で ESCON アダプターを使用している場合はセグメントのサイズを常に 4096 にする必要があります。

最後に TCP/IP ネットワークの場合は、TCP 送信および受信バッファ・サイズを 32768 より大きな値に設定する必要があります。通常は 65536 の値が最適です。

**注:** ゲートウェイからサーバーへの接続 (アウトバウンド接続) を確立するには、クライアントからゲートウェイへの接続 (インバウンド接続) を確立する場合よりコストがかかります。数千ものクライアントがゲートウェイを介してサーバーに対し接続と切断を頻繁に繰り返す環境では、アウトバウンド接続を確立するのに相当の処理時間を要します。DB2 コネクトでは TCP/IP 上で接続プーリングが行われます。クライアントでサーバーからの切断が要求されると、ゲートウェイではクライアントとのインバウンド接続は除去されますが、サーバーとのアウトバウンド接続はプール内に保持されます。新しいクライアントでゲートウェイに対して接続要求がなされ

ると、ゲートウェイによりプールから既存の接続が提供されるので、接続時間が全体として短縮され、高コストのサーバーへの CPU 接続が節約されます。

DB2 での接続プールについては、[管理の手引き](#) を参照してください。

次の表に、ネットワーク・パフォーマンスのチューニング方式を要約してあります。

参照するもの	例	設定	注意
意図的な遅延	ネットワーク装置の遅延パラメーター	0 に設定します。	通常、省略時値は大き過ぎます。
バッファ	IOBUF パラメーター	3992 にセットアップします。	ESCON や他のチャネル・アダプターの場合に特に有効です。
	RUSIZE	最適なサイズは 4096 です。	RUSIZE と RQRIOBLK を同サイズに設定すると最善のパフォーマンスが得られます。
	ペーシング	VPACING、PACING、および Mode Profiles を 63 に設定します。	適切な場合は適応ペーシングを使用します。
アダプターの設定	送信受信待ち行列の長さ	推奨値は 150。	通常、省略時値は 32 です。
	SNA での DLC ウィンドウ	送信ウィンドウ・サイズは大きい値 (8 以上) に設定します。受信ウィンドウ・サイズは小さい値 (1 など) に設定して、繰り返しテストしながら値を大きくし、理想的な値を見つけます。	それぞれの論理装置により遅延が追加されます。ネットワーク・トポロジーをできるだけ単純なものにしてください。

参照するもの	例	設定	注意
TCP の設定	セグメント・サイズ	イーサネットでは 1500、トークンリングおよび FDDI では 4400 です。	TCP/IP 接続に ESCON アダプターを使用する場合は必ず 4096 に設定する必要があります。
	送信 / 受信スペースのサイズ	両方とも 64K にする必要があります。	Windows の場合、省略時値はほんの 8192 です。 Windows レジストリーで設定できます。

## ネットワーク・ハードウェア

以下の考慮事項は、ハードウェアに関するものです。

- ネットワークまたは伝送媒体の速度

パフォーマンスは、より速い伝送媒体の使用により改善されます。たとえば、以下に挙げるような、いくつかの代表的な生データ転送速度がありません。

### チャンネル間 (光ファイバー)

4.0 MB/s

### 16 Mbps LAN

2.0 MB/s

### チャンネル間 (正規)

1.0 MB/s

### 4 Mbps LAN

0.5 MB/s

### 高速 T1 搬送波 (1.544 Mbps)

0.193 MB/s

### 高速リモート 56 Kbps 電話回線

0.007 MB/s

### 19.6 Kbps モデム

0.002 MB/s

### 9600 bps モデム

0.001 MB/s

データ転送速度は、ホストまたは AS/400 データベース・サーバーに向かうパスの中のもっとも遅い伝送媒体によって制約されます。

- ネットワーク・アダプターまたは通信制御装置  
ネットワーク・アダプターおよび通信制御装置のメモリー使用については、注意深く計画を立てる必要があります。さらに、ネットワーク担当技術者と一緒に、制御装置が DB2 コネクトで発生した余分の通信量を取り扱う能力をもっているか確認してください。
- ネットワーク・トポロジー  
データが LAN から LAN へと、また 1 つの SNA ネットワークから別の SNA ネットワークへとわたって移動する場合は、搬送時間を考慮してください。ブリッジ、ルーター、およびゲートウェイは経過時間を上乗せすることになります。たとえば、横断するブリッジの数を減らせば、各要求に必要なホップの数を減らすことになります。  
ノード間の物理的距離も、考慮に入れなければなりません。メッセージが通信衛星によって転送される場合でも、転送時間は、光速 ( $3 * 10^{**8}$  m/s) および送信者と受信者との間の往復距離によって制約を受けます。
- ネットワーク通信量  
ネットワークの帯域幅がフルに活用されている場合、単一アプリケーションについての応答時間とデータ転送速度が減少します。  
データがネットワークの特定部分に蓄積されたとき、ネットワーク内に混雑が起きます。たとえば、バッファ・サイズが非常に小さい旧 NCP の場合などです。
- ネットワークの信頼性  
ネットワークのエラー率が高い場合、ネットワークのスループットが減少して、データの再伝送のためにパフォーマンスの低下を招きます。

## システム・リソースの競合

システム内の多くのタスクがシステム・リソースを求めて競合する場合は、パフォーマンスの劣化があり得ます。以下の質問を考慮してください。

- CPU は飽和状態ですか。システムのアップグレード、システム・ワークロードの低減、およびプロセス・オーバーヘッド低減のためのシステムのチューニングを考慮してください。
- メモリーは、オーバー・コミットになっていますか。メモリーのアップグレード、システム・ワークロードの低減、およびプロセス・オーバーヘッド低減のためのシステムのチューニングを考慮してください。

- 通信アダプター / 通信制御プログラムは混雑していますか。ネットワークのアップグレードまたはトークンリング・カードのペアアップを考慮してください。
- サブシステムの 1 つが特に混雑していますか。そしてこのサブシステムはデータ・パス上にありますか。
- 不必要なプロセスまたはタスクがシステム上で稼働していませんか。一般規則として、常時使用するサービスでなければ、システム・リソースの無駄使いになりますので、構成したり起動したりしないことです。
- 少数のプロセスまたはタスクがリソースの大部分を使用していますか。それらは止められますか。それらの優先順位は下げられますか。それらを洗練して多くのリソースを使わないようにできますか。

## パフォーマンスのトラブルシューティング

DB2 コネクト・ユーザーが、ホストまたは AS/400 サーバーから大きな照会をしていて長い時間待っても応答がない場合、以下の領域を考慮して、考えられるパフォーマンス上の問題の原因を調べてください。

1. ホストまたは AS/400 サーバーから多数のデータ・ブロックが戻ってしまうような照会に関しては (通常は 32K かそれ以上のデータ)、データベース・マネージャー構成パラメーター RQRIOBLK を 32767 に設定しているか確認してください。これは、コマンド行プロセッサ (CLP) を以下のように使用して行います。

```
db2 update database manager configuration using RQRIOBLK 32767
```

2. ホストまたは AS/400 サーバーへの接続に VTAM を使用している場合、「交換回線大ノード (switched major node)」構成の下の PACING パラメーターの値を見てください。DB2 コネクト・ワークステーション上で、IBMRDB モード定義用の「LU 6.2 モード・プロファイル (LU 6.2 Mode Profile)」の通信設定を検査してください。この定義において、「受信パシングウィンドウ (Receive pacing window)」パラメーターの値が VTAM で定義した PACING 値以下になっているかを確認してください。DB2 コネクト・ワークステーションの「歩調合わせウィンドウ (Receive pacing window)」と VTAM の "PACING" の共通の値は、8 です。
3. IBMRDB モード定義で定義した最大 RU サイズが適切な値に設定されているか確認してください。トークンリング・ハードウェアを使用して接続する場合、4K を下回らない値に設定することをお勧めします。イーサネット・ハードウェアを使用して接続する場合、イーサネット・フレーム・サイズの最大値は 1536 バイトであることに注意してください。その値が限界要因になります。

4. 使用している環境の VTAM 管理者と相談して、VTAM が DB2 コネクト・ワークステーションとの LU-LU セッションで「適応歩調合わせ」を使用しているか確認してください。

---

## SNA パフォーマンス調整の追加のヒント

このセクションには、DB2 コネクトで使える SNA パフォーマンス調整の追加のヒントを記載します。

### DB2 コネクトに関する一般パフォーマンス情報

DB2 コネクトのパフォーマンス特性は、ほとんどプロセッサを使用して、入出力はわずかしかな行わないところにあります。プロセッサ速度が速ければ速いほど、DB2 コネクトの実行速度も速くなります。DB2 コネクトは、SMP のプロセッサ構成を十分に活用しています。

DB2 コネクト エンタープライズ・エディション・サーバーの速度が速いと、クライアント処理時間、ネットワーク処理時間、ホストまたは AS/400 サーバーでの処理時間を計算しない場合、SQL 要求 / 応答の対を 5 ミリ秒未満で処理することができます。わずかなデータ行しかない単純な SQL ステートメントまたは照会なら、終端間 (クライアントからホストまたは AS/400 サーバーへの往復) で 0.1 秒未満で完了できます。

1 つの照会に 5 個以上の SQL ステートメントがある場合は、ストアード・プロシージャを使えば、OLTP の高いパフォーマンスが保証され、SQL ステートメント間のネットワーク処理の遅延が原因で発生するロック競合の増加を回避することもできます。

パフォーマンス上の問題は、通常は使用中のホスト処理装置接続機構のタイプによって、また、ネットワーク経路指定、チューニング特性、およびアプリケーション設計が原因で発生します。一般的な DB2 コネクトのパフォーマンス情報は、174ページの『DB2 コネクトのパフォーマンスに関するその他の情報源』にあります。

### ネットワーク接続機構の選択と調整

DB2 コネクトの使用時に最高のパフォーマンスが得られるネットワーク接続機構には、以下のものがあります、

1. チャンネル接続カード
2. IBM 3172 モデル 3 以降のモデル、または同等の機能をもつ製品
3. IBM 2216

4. 開放型システム・アダプター・カード (OSA-1 ではなく OSA-2)
5. IBM 3745 (ネットワーク制御プログラム (NCP) 付き)
6. IBM 3174 端末制御機構、または同等の機能をもつ製品

最後の選択肢は、お勧めしません。以下をご覧ください。

ホストに接続する方法で推奨される最善の方法は、AIX、Windows NT、または Windows 2000 の ESCON チャンネル接続カードを使うことです。IBM 3172 モデル 3 および 2216 のパフォーマンスも良好ですが、送達のスループットは ESCON 使用時よりも劣る傾向が見られます。

ESCON カードで AIX を使う場合は、MPC (マルチ・パス・チャンネル) に関連する PTF を適用してください。これらの PTF がないと、AIX SNA ESCON ドライバーのパフォーマンスが低下します。詳細については、174ページの『ESCON を介した SNA のマルチ・パス・チャンネル・サポート』を参照してください。次のアドレスでも、詳細情報が提供されています。

<http://www.networking.ibm.com.cms/cmsnew01.html>

DB2 コネクトのパフォーマンスを最適化するために調整する必要のある通信サーバー、NCP、および VTAM パラメーターのチェックリストについては、174ページの『NCP を介する DB2 コネクト接続の調整方法』を参照してください。NCP 固有以外の推奨事項は、すべてのタイプの DB2 コネクトおよびクライアント / サーバー接続機構に適用できます。

System/390 で OSA-2 カードを使うと、小さなトランザクションが大量にある場合などは、1 秒当たりの送信フレーム機能が劣るため、3272 モデル 3 使用時ほど高いスループットは得られません。最新の機能強化の詳細については、178ページの『OSA-2 機能強化に関する情報』を参照してください。

NCP 付きの 3145 は、通常は特に既存のネットワーク通信量に合わせて調整されます。このため、データベース・クライアント / サーバー・アプリケーションではあまりパフォーマンスが良くない場合があります。ほとんどの DB2 コネクトのパフォーマンス上の問題は、NCP と VTAM との間や NCP 間の時間遅延が原因で発生します。調整チェックリストを提供している 174ページの『NCP を介する DB2 コネクト接続の調整方法』を参照してください。

一般に、3174 端末制御機構は、パケット・サイズ (RU サイズ) が 256 バイトでは小さすぎるので、使用しないようお勧めします。APPC データベース接続で、独立 LU サポートを提供するには、3174 マイクロコード・レベル C が必要です。OEM 3174 と同等の機能を持つ製品の中には、類似の従属性を持つものがあるかもしれません。

## DB2 コネクトのパフォーマンスに関するその他の情報源

- <http://www.ibm.com/software/data/db2/library> の DB2 テクニカル・ライブラリーの Web サイトを検索してください。キーワード「DB2CONNECT」と「Performance」を指定して DB2 ユニバーサル・データベース・ライブラリーで「Technotes」を検索します。World Wide Web で DB2 コネクトの考慮事項のセクションの最新情報を入手できます。
- DB2 コネクトに関する DB2 クライアント / サーバーのパフォーマンス報告も多数あります。 <http://www.ibm.com/software/data/performance> をご覧ください。

## ESCON を介した SNA のマルチ・パス・チャネル・サポート

ESCON を介して SNA のマルチ・パス・チャネル (MPC) サポートを使うと、IBM eNetwork 通信サーバーを実行するシステムで、ESCON アダプターを使用してホストに MPC リンク・ステーションを作成できます。MPC は、普通 CDLC より高速です。理由は以下のとおりです。

1. MPC は、読み取りと書き込みで別々のサブチャネルを使用している。
2. MPC は、IOBUF サイズによる制限がない。フレームは 4K で、ブロック化できます。

テストによると、IOBUF サイズ 1K 未満の ESCON チャネル・データ・リンク制御 (CDLC) リンクと比較して、MPC リンクではパフォーマンスが 3 倍も向上しました。AIX SNA MPC では、ESCON と MVS VTAM V4R4 以降、および機能コード 4024 の Communications Server for AIX (5765-652) が必要です。Windows NT システムでは、IBM eNetwork Communications Server for Windows NT バージョン 6 を使用する必要があります。

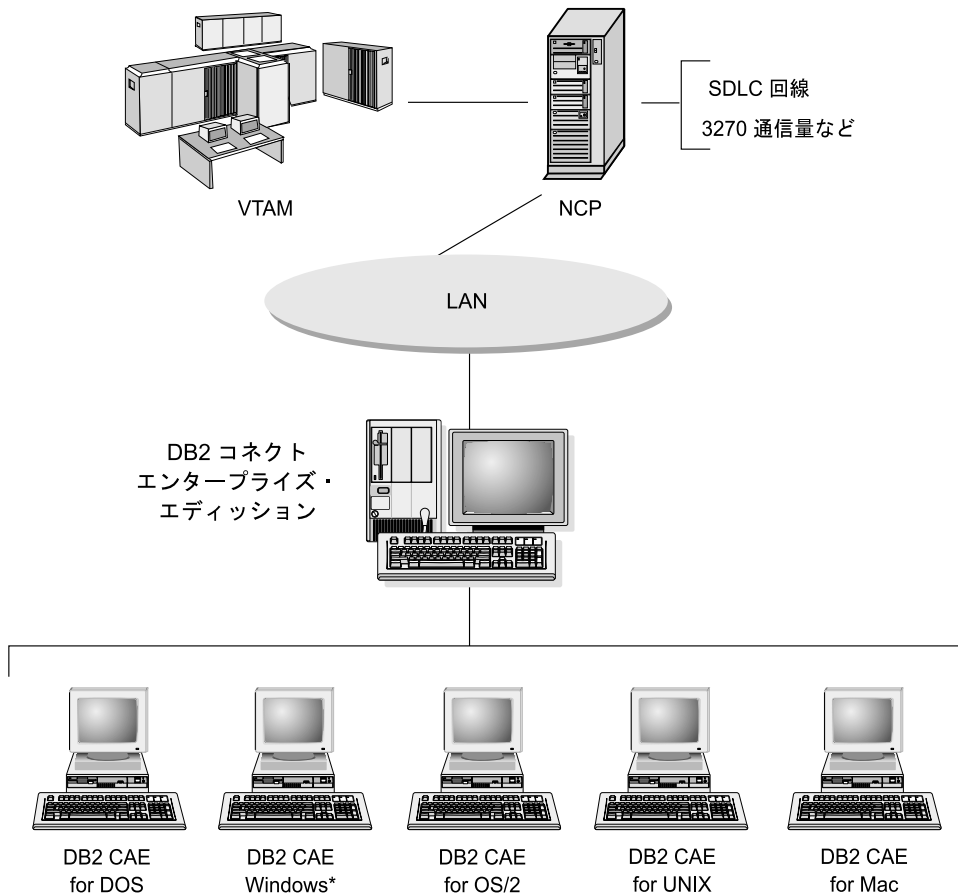
MPC で必要な Communications Server for AIX の PTF を、以下に記載します。

APAR #	PTF #	LPP name
IX67032	U449693	sna.books.chdoc
IX67032	U449693	sna.books.escdoc
IX67032	U449300	sna.rte
IX67032	U450027	sna.msg.en_US.rte
IX65820	U447759	sna.dlcchannel
IX67618	U449691	mpc.rte
IX65813	U447758	devices.mca.8fc3.rte

## NCP を介する DB2 コネクト接続の調整方法

典型的なネットワーク構成は、以下のようになっています。





\*16 および 32 ビット Windows オペレーティング・システム

図8. DB2 コネクト エンタープライズ・エディッション・ゲートウェイの SNA ネットワークのシナリオ

このシナリオは、ホストまたは AS/400 データベース・サーバーから DB2 コネクト エンタープライズ・エディッション・ゲートウェイまでのスループットと応答時間、およびこれらに影響を与える可能性のあるいろいろなパラメーターに照準を合わせています。

### 調整の基準

提案されている変更順序は、以下のとおりです。

- 1 - DELAY on PCCU macro\*
- 2 - DLC/LLC Tuning\*
- 3 - PIU size\*
- 4 - Pacing window changes\*
- 5 - DELAY on LINE macro\*

- 6 - MAXBFRU changes
- 7 - LAN Frame sizes
- \* Major improvement in throughput is possible

**PIU サイズ (RU + 29 バイト):** ホストと DB2 コネクト・サーバーの RU サイズは、最大にしてください。このことは、VTAM プログラム・スタックが走査する回数を最小に抑えるため、API 交差 (可能な場合トランザクションの SEND と RECEIVE の両方のデータ) を収容できるような RU サイズになっている必要があることを示唆しています。また、RU のセグメント化が望ましくない場合は、ネットワーク・フレーム・サイズによっても RU の最大サイズが制限されます。

DB2 コネクトのブロック・サイズ (RQRIOBLK)、RU、およびページング値を、「RU \* 歩調合わせ >= RQRIOBLK」となるように設定するのは良案です。たとえば、省略時の RQRIOBLK サイズ 32K はほとんどの場合ちょうど良い値なので、この値の利点を活用するため RU = 4K に設定し、受信ウィンドウ・ページングを 8 にします。

- RU サイズとページングは、DB2 コネクト・ワークステーションと VTAM の両方で定義されるモード・テーブルで設定します。モード・テーブル定義は、どちらにおいても同じになっている必要があります。
- RQRIOBLK は、DB2 UPDATE DBM CFG コマンドを使用して設定します。
- ネットワーク・フレーム・サイズ I フレームは、DB2 コネクト・ワークステーションの DLC 構成と NCP で設定します。

**ページング・ウィンドウ・サイズ:** セッションおよび仮想経路ページングのウィンドウは最大化しておいてください。ネットワーク輻輳や仮想経路占有条件などが発生する原因とならない最大値を使う必要があります。テスト環境の場合は、歩調合わせを 0 (歩調合わせなし) か、最大値 X'3F' に設定します。

**コートテーリング値 (DELAY):** コートテーリングは、DELAY パラメーターによって制御されます。PCCU マクロにある DELAY パラメーターは、アウトバウンド・コートテーリング (ホストに関してアウトバウンド) を制御します。NCP の LINE 定義ステートメントにある DELAY 値は、インバウンド・コートテーリング (ホストに関してインバウンド) を制御します。

DELAY 値は、PIU が伝送されるまで待ち行列 (NCP または VTAM) に保留される時間の長さを決定します。この待ち時間の目的は、他の PIU が一時的に着信し、そのすべてを単一のチャネル・プログラムで伝送できるようにする可能性を高めることにあります。待ち時間を最短にするには、DELAY 値を 0 に設定してください。アウトバウンド・コートテーリング遅延値を 0 に変更し

でも、アウトバウンド通信量のパフォーマンスが高くなるだけで、ホストには特に何の影響もありません。インバウンド通信量のパフォーマンスもいくらか向上します。

NCP の DELAY を 0 に変更するときは、もう少し注意する必要があります。この値を 0 に設定できるのは、NCP が過負荷になっておらず、インバウンド通信量のかなりの部分が小さなフレームで構成されていない場合です。特に負荷が軽い場合やテスト / ベンチマーク環境で DELAY の値を 0 に設定すると、応答時間がかなり短くなります。

```
VTAMB7  PCCU  CUADDR=CAF,
AUTODMP=NO,
AUTOIPL=NO,
AUTOSYN=YES,
BACKUP=YES,
DELAY=0,
VFYLM=YES,
CHANCON=UNCOND,
MAXDATA=32768,
DUMPDS=NCPDUMP,
OWNER=HOSTB7,
SUBAREA=17
```

```
LNCTLS      GROUP  LNCTL=CA,CA=TYPE6,DELAY=0.0,TIMEOUT=500.0
CA0         LINE   ADDRESS=00
PUCHAN0    PU     PUTYPE=5,TGN=1
CA1         LINE   ADDRESS=01
PUCHAN1    PU     PUTYPE=5,TGN=1
```

DELAY の考慮事項については、*VTAM Network Implementation Guide* に文書化されています。

**MAXBFRU:** MAXBFRU 値は、最大 PIU サイズの 2、3 倍の値に設定する必要があります。

**DLC/LLC レイヤーの調整:** NCP と DB2 コネクト エンタープライズ・エディション・ゲートウェイの LLC2 ウィンドウ・サイズ (DLC の送受信ウィンドウ数) は、同じになるようにしてください。こうしておく、特にサーバーが DB2 コネクト (AIX版) の場合にかかなりの効果があります。送信ウィンドウ数は受信ウィンドウ数より高くするようお勧めします。

一般に、トークンリングを介する SNA 接続の場合は、LLC2 タイマー / ウィンドウを最適化する必要があります。この変更によって、スループットと応答時間が 6 倍も向上する場合があります。

**LAN フレーム・サイズ:** トークンリングの最大フレーム・サイズは、できるかぎり大きくしてください。

## OSA-2 機能強化に関する情報

以下の資料は、IBM WSC Flash 文書番号 9718 からの複製です。

TITLE: WSC FLASH 9718: OSA-2 ENHANCEMENTS AVAILABLE  
DOCUMENT ID G023691 UNCLASSIFIED

Open Systems Adapter 2 (OSA-2) Systems Network Architecture (SNA) enhancements are being made available earlier than previously announced. The enhancements are:

- o SNA/APPN enhancements for OS/390, MVS/ESA, VM/ESA, and VSE/ESA
  - Enhanced availability: load balancing, redundancy, and overflow
  - Enhanced connectivity: increased Physical Unit (PU) support (from 255 PUs per port to 2047 PUs per port).
- o Support for ACF/VTAM for VSE/ESA networks

NOTE: These enhancements do not pertain to OSA-1.

### LOAD BALANCING, REDUNDANCY, AND OVERFLOW

**LOAD BALANCING:** A single Medium Access Control (MAC) address can now be defined for attached OSA-2 SNA/APPN Physical Units (PUs), even though connections may be via multiple physical ports. This support is offered for source-route bridged environments only (Token-Ring and FDDI). The number of sessions established through a port is monitored, and user session loads are evenly distributed across the equally configured ports.

**REDUNDANCY:** A secondary path between the LAN workstation and the host system can now be configured. If the primary path becomes unavailable, the secondary path will receive the LAN traffic. This increases system availability and simplifies network management.

**OVERFLOW:** User sessions flow through the primary OSA-2 port until the session capacity has been reached. Additional user sessions will automatically flow to the next OSA-2 port. Since all user workstations are identically configured, network administration is simplified and the network becomes more scalable. New users can be added non-disruptively.

Load balancing, redundancy, and overflow support is provided by PTFs for OSA/SF as follows:

- o OS/390 and MVS - OW20205/UW34618 03/31/97
- o VM/ESA - OW23952/UW37028 03/31/97
- o VSE/ESA - Provided with VSE/ESA V2.2.1 04/29/97

### INCREASED PHYSICAL UNIT (PU) SUPPORT (VIA OSA/SF):

The architecture has been changed to allow up to a maximum of 2047 PUs per physical port to be defined for OSA-2 Ethernet, Token-Ring and FDDI features instead of the current 255 PUs per port. This enhancement is available for currently installed features, as well as new installations. Actual connectivity may vary based upon user workloads.

Increased Physical Unit (PU) Support is provided by PTFs for OSA/SF as follows:

- o OS/390 and MVS - OW23429/UW37210 03/31/97
- o VM/ESA - OW24952/UW37028 03/31/97
- o VSE/ESA - PQ03091/UQ04224 04/29/97

Increased Physical Unit (PU) Support is provided by PTFs for ACT/VTAM as follows:

- o ACF/VTAM for OS/390 and MVS
  - VTAM 4.1 OW14043/UW24904
  - VTAM 4.2 OW14043/UW24905
  - VTAM 4.3 OW14043/UW24906
- o ACF/VTAM VM/ESA
  - VM60877/UV59834
- o ACF/VTAM VSE/ESA
  - DY44347/UD50254

VSE/ESA - SNA SUPPORT

---

OSA-2 and OSA/SF support is delivered via VSE/ESA Version 2 Release 2.1. This announcement of VSE/ESA support satisfies the Statement of General Direction contained in Hardware Announcement 196-194, and Hardware Announcement 196-193, dated September 10, 1996.

The OSA-2 feature provides ACF/VTAM for VSE/ESA host applications with direct access to Ethernet, Token-Ring, and FDDI LANs and Asynchronous Transfer Mode (ATM) Forum-compliant LAN emulation networks.

OSA/SF is available:

- o As a non-exclusive element of OS/390 Release 1 or above (5645-001)
- o As a separate program product, S/390 Open Systems Adapter Support Facility Version 1 Release 2 for MVS/ESA 4.3 or above (5655-104)
- o As a facility of VM/ESA Version 2 Release 2.0 (5654-030)
- o As a component of VSE Central Functions 6.1.1 in VSE/ESA Version 2 Release 2.1 (5690-VSE).

MORE INFORMATION

---

Announcements 297-043, 297-040

---

## その他の情報源

このセクションでは、本書以外の情報源を列挙します。

### その他の関連資料

パフォーマンスの詳細については、以下を参照してください。

- *DB2 Connect for OS/2 to DB2 Performance Benchmark*

- *SNA Server for AIX and SNA Server Gateway for AIX Performance Guide*

## **WWW の使用**

DB2 コネクトのパフォーマンスのチューニングに関する広範な情報や、事例研究や例などを World Wide Web に見いだすことができます。 Web ブラウザーを次の URL に設定します。

<http://www.ibm.com/software/data/db2/performance/>

## **SNA ユーザーに関する追加のヒント**

172ページの『SNA パフォーマンス調整の追加のヒント』を参照してください。

---

## 第13章 問題判別

DB2 コネクト環境には、複数のソフトウェア、ハードウェア、および通信製品が含まれます。問題判別の最も良い方法は、利用可能なデータを排除および限定していき、結論 (エラーの発生箇所) に到達することです。

以下のトピックは、問題判別プロセスの助けとして提供されています。

- 『その他の情報源』
- 182ページの『関係する情報の収集』
- 182ページの『初期接続が正常に行われなかった場合』
- 184ページの『初期接続後に発生する問題』
- 185ページの『診断ツール』
- 186ページの『トレース・ユーティリティー (ddcstrc)』
- 196ページの『よくある DB2 接続問題』

関係のある情報を収集し、あてはまるトピックの選択に基づいて、その後、参照されたセクションへ進んでください。

---

### その他の情報源

このセクションでは、本書以外の情報源を列挙します。

#### 問題判別の手引きの使用

DB2 コネクトおよび DB2 ユニバーサル・データベースの問題判別トピックに関する詳細については、*問題判別の手引き* を参照してください。

#### WWW の使用

DB2 コネクトの問題判別のヒントに関する最新の情報を、World Wide Web の DB2 Product and Service Technical Library に見い出すことができます。

1. Web ページ <http://www.ibm.com/software/data/db2/library/> にアクセスします。
2. 「DB2 ユニバーサル・データベース」リンクを選択します。
3. 「DDCS」あるいは「Connect」というキーワードを使用して、「Technotes」を検索します。

## APPC、CPI-C、および SNA センス・コードの資料

APPC、CPI-C、および SNA センス・コードに関する資料は再パッケージされ、PostScript ファイル、および HTML ブック (英語版のみ) として提供されます。

このブックの PDF 版は、DB2 資料の CD-ROM に見いだすことができます。DB2 コネクトがインストールされている場合は常にこのブックの HTML コピーがインストールされており、このブックは DB2 インフォメーション・センターに統合されています。

---

### 関係する情報の収集

問題判別には、問題の範囲を絞り込み、考えられる原因を調査することが含まれます。はじめに、関係する情報を収集し、分っている事柄、まだ収集していないデータや、省略できるパスなどの判別を行ってください。少なくとも以下の質問に答えるようにしてください。

- 初期の接続は成功しましたか。
- ハードウェアは正常に機能していますか。
- 通信パスは機能していますか。
- 以前のディレクトリー項目を無効にするような通信ネットワークの変更が行われましたか。
- データベースは始動していますか。
- 通信中断は、クライアントと DB2 コネクト・ワークステーションとの間ですか。DB2 コネクト・ワークステーションとホストまたは AS/400 データベース・サーバーとの間ですか。すべてのクライアントですか、それとも 1 つのクライアントですか。
- メッセージ内容およびメッセージに戻されているトークンから、何を判別することができますか。
- 現時点での診断ツールの使用は効果がありそうですか。
- 他の機械で同じような作業を行っている場合、正しく作動していますか。
- これがりモート・タスクの場合、ローカルに行っても正常に作動しますか。

---

### 初期接続が正常に行われなかった場合

以下の質問を検討して、インストール・ステップが正しく行われたか確認してください。

1. インストール・プロセスは正常に完了しましたか。
  - 前提条件のソフトウェア製品はすべて使用可能でしたか。



- メモリーおよびディスク・スペースは十分ありましたか。
  - リモート・クライアント・サポートはインストールされましたか。
  - 通信ソフトウェアのインストールは、何のエラー状態もなく完了しましたか。
2. UNIX 系のシステムの場合、製品インスタンスは作成されましたか。
    - ルートとして、インスタンス所有者になるユーザーおよび `sysadm` グループになるグループを作成しましたか。
  3. ライセンス情報は正常に処理されましたか (該当する場合)。
    - UNIX 系のシステムの場合、ノード・ロック・ファイルを編集し、IBM が指定したパスワードを入力しましたか。
  4. ホストまたは AS/400 データベース・サーバーとワークステーションとの通信の構成は正常に行われましたか。
    - 考慮すべき 3 つの構成があります。
      - a. ホストまたは AS/400 データベース・サーバーの構成は、サーバーに対するアプリケーション・リクエスターを識別します。ホストまたは AS/400 サーバー・データベース管理システムは、そのリクエスターをロケーション、ネットワーク・プロトコル、および機密保護の点で定義するシステム・カタログ項目を持つことになります。
      - b. DB2 コネクト・ワークステーション構成は、サーバーへのクライアント数を定義し、クライアントへのホストまたは AS/400 サーバーを定義します。
      - c. クライアント・ワークステーション構成では、ワークステーションの名前と通信プロトコルが定義されている必要があります。
    - 初期接続の不実行についての問題分析には、SNA 接続を検査し、すべての LU (論理装置) 名および PU (物理装置) 名が完全であって正しいかどうか確認することが含まれます。また、TCP/IP 接続を検査し、正しいポート番号とホスト名を指定しているかどうか確認することが含まれます。
    - ホストまたは AS/400 サーバー・データベース管理者およびネットワーク管理者の両方は、問題の診断に利用可能なユーティリティを持っています。
  5. ホストまたは AS/400 サーバー・データベースを使用するために、ホストまたは AS/400 サーバー・データベース管理システムにとって必要な権限レベルを所有していますか。
    - ユーザーのアクセス権限、表修飾子の規則、および予測される結果を考慮してください。

6. コマンド行プロセッサを使用してホストまたは AS/400 データベース・サーバーに対して SQL ステートメントを発行しようとする、正常に実行できませんか。
  - コマンド行プロセッサをホストまたは AS/400 データベース・サーバーへバインドする手順を守り行いましたか。

---

## 初期接続後に発生する問題

問題の範囲を絞り込むのに役立つ始点として、次の質問が挙げられます。

1. 何か特別なもしくは異常な操作状況が見られますか。
  - それは新しいアプリケーションですか。
  - 新しいプロシージャが使用されていますか。
  - システムに影響を与える可能性のある変更が最近行われましたか。たとえば、アプリケーションまたはシナリオが正常に実行された後、ソフトウェア製品またはアプリケーションのどれかが変更されましたか。
  - アプリケーション・プログラムについて、そのプログラムを作成するためにどんなアプリケーション・プログラミング・インターフェース (API) が使用されましたか。
  - そのソフトウェアまたは通信 API を使用した他のアプリケーションが、ユーザーのシステムで実行されたことがありますか。
  - PTF を最近インストールしましたか。オペレーティング・システムにおいて、インストールされて以来使用されていない (もしくはロードされていない) 機能を用いようとして問題が発生した場合は、IBM の最新 PTF レベルを判別し、その機能をインストールした後、そのレベルをロードしてください。
2. このエラーは以前にも起こりましたか。
  - 以前のエラー状態には文書化された解決法がありましたか。
  - システムへの参加者はだれでしたか。その人たちが、考えられる処置の方向性にヒントを与えることができませんか。
3. ネットワークに関する情報を戻す通信ソフトウェア・コマンドを利用して、よく調べてみましたか。
  - SNA ソフトウェアで利用できる検査ツールがありますか。
  - TCP/IP を使用している場合、TCP/IP コマンドおよびデーモンを使用して取り出される貴重な情報のある場合があります。
4. *SQLCA* (*SQL* 連絡域) に戻される有用な情報がありますか。

- 問題処理手順は、SQLCODE および SQLSTATE の各フィールドの内容を調べるステップを含んでいる必要があります。
  - SQLSTATE によって、アプリケーション・プログラマーはデータベース製品の DB2 ファミリーに共通するエラーのクラスについてテストすることができます。分散リレーショナル・データベース・ネットワーク内でのこのフィールドは、共通の基礎を提供する場合があります。詳細については、メッセージ解説書 を参照してください。
5. *DB2START* を、サーバーで実行しましたか。 加えて、*DB2COMM* 環境変数が、サーバーにリモート・アクセスするクライアント用に正しく設定されているかを確認してください。
  6. 同じ作業を行っている他の機械は、サーバーに正常に接続することができましたか。サーバーに接続しようとしているクライアントが、最大数に達している場合があります。別のクライアントがサーバーから切断された場合、その直前のクライアントも接続できませんでしたか。今接続することはできますか。
  7. 機械は適正なアドレス指定を行っていますか。その機械がネットワーク上で固有のものかどうか検査してください。
  8. リモート接続をしている場合、適正な権限がクライアントに認可されていますか。インスタンスへの接続が正常に行われても、データベースや表レベルでは権限が認可されていない場合があります。
  9. これはリモート・データベースに接続する最初の機械ですか。分散環境内では、ネットワーク間のルーターやブリッジが、クライアントとサーバーとの通信をブロックする場合があります。たとえば、APPC を使用する場合には、セッションが設定可能かどうか確認してください。TCP/IP を使用する場合は、リモート・ホストを PING できるかどうか確認してください。

---

## 診断ツール

問題が発生したときは、以下のツールを使用することができます。

- 基本障害保守ログ。ここでは診断情報が統合され、読み取れる形式で保管されます。詳細については、問題判別の手引き を参照してください。ログ内のメッセージに関する情報については、メッセージ解説書 を参照してください。
- *db2diag.log*  
このファイルは、UNIX システムでは `/u/db2/sql1lib/db2dump/db2diag.log` 内にあります。db2 はインスタンス名です。  
このファイルは、Intel システムでは `x:%sql1lib%db2%db2diag.log` 内にあります。x: は論理ドライブで、db2 はインスタンス名です。

- db2alert.log (db2diag.log と同じファイル位置にあります。)
- トレース・ユーティリティ。『トレース・ユーティリティ (ddcstrc)』に記述されているとおり。
- UNIX ベースのシステムについては、**ps** コマンド。これは活動状態のプロセスについてのプロセス状況情報を標準出力に戻すものです。
- UNIX ベースのシステムについては、コア・ファイル。これはサーバー・エラーが起きたとき、現行ディレクトリー内で作成されます。コア・ファイルには、終了したプロセスのメモリー・イメージが入っていて、どの機能がエラーの原因となっているかを判別するのに使用できます。
- Windows NT および Windows 2000 システムについては、イベント・ビューアを使用してください。

TCP/IP 接続のトラブルシューティング (または別のトピック) に関する詳細は、[問題判別の手引き](#) を参照してください。または、DB2 Product and Service Technical Library 上の「Technotes」を検索してください (181ページの『WWW の使用』を参照)。

---

## トレース・ユーティリティ (ddcstrc)

**ddcstrc** ユーティリティを利用して、DB2 コネクト・ワークステーション (データベース・クライアントの代理) とホストまたは AS/400 データベース・サーバー管理システムとの間で交換されたデータの記録を入手することができます。

データベース管理者として (またはアプリケーション開発者として)、このデータ・フローがどのように働くかを理解することは有用です。この知識は、特定の問題の源を判別するのに役立つからです。たとえば、CONNECT TO データベース・ステートメントをホストまたは AS/400 データベース・サーバーに対して発行したが、コマンドが失敗して、失敗の戻りコードを受け取ったとします。そのとき、どのような情報がホストまたは AS/400 データベース・サーバー管理システムに送られたかを正確に理解していれば、たとえ戻りコードの情報が一般的なものであったとしても、失敗の原因を判別することができます。ユーザー自身による単純なエラーが、多くの失敗の原因となっています。

ddcstrc からの出力は、DB2 コネクト・ワークステーションとホストまたは AS/400 データベース・サーバー管理システムとの間で交換されたデータ・ストリームをリストします。ホストまたは AS/400 データベース・サーバーへ送られたデータには SEND BUFFER とラベル付けされ、ホストまたは AS/400 データベース・サーバーから受け取られたデータは RECEIVE BUFFER とラベル付けされます。

受信バッファが SQLCA 情報を含んでいる場合、その後、このデータの書式化された解釈が続き、SQLCA とラベル付けされます。SQLCA の SQLCODE フィールドは、ホストまたは AS/400 データベース・サーバーにより戻されるとおりのマップされていない値です。(マッピングに関する詳細については、137ページの『第11章 SQLCODE マッピング』を参照してください。)送信バッファと受信バッファは、ファイル内で最も古いものから順に、最新のものへと配置されます。それぞれのバッファには、以下のものが入ります。

- プロセス ID
- SEND BUFFER、RECEIVE BUFFER、または SQLCA ラベル。バッファ内の 1 番目の DDM コマンドまたはオブジェクトは、DSS TYPE とラベル付けされています。

送信バッファと受信バッファ内の残りのデータは、以下のものを構成する 5 つの列に分けられます。

- バイト・カウント。
- 第 2 列および第 3 列は、2 つのシステム間で交換される DRDA データ・ストリームを ASCII または EBCDIC で表します。
- 第 2 列および第 3 列の ASCII 表示。
- 第 2 列および第 3 列の EBCDIC 表示。

DDM の詳細については、以下を参照してください。

- *DB2 (OS/390 版) リモート DRDA リクエスター / サーバー解説書*
- *Distributed Relational Database Reference*
- *Distributed Data Management Architecture Level 3: Reference*

## トレース構文

このコマンドは、オペレーティング・システム・コマンド・プロンプトから、以下の構文で呼び出されます。

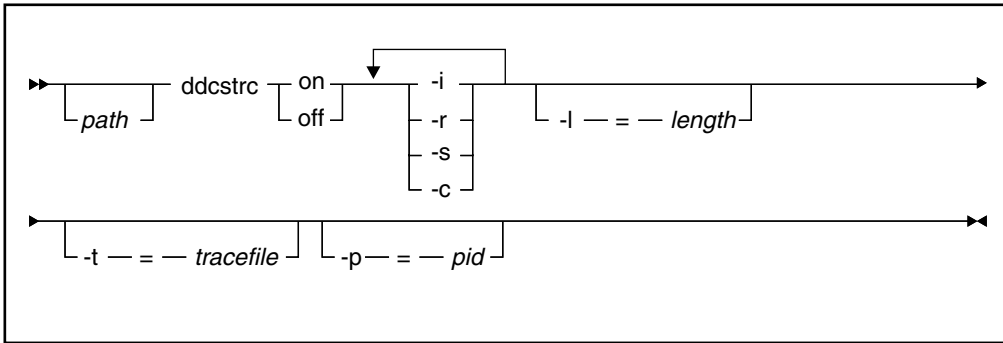


図9. *ddcstrc* コマンドの構文

**注:** このコマンドの構文は、使用しているオペレーティング・システムにより微妙に変わることがあります。たとえば、OS/2 オペレーティング・システムでは `.` の代わりに `/` が使用されます。

## トレース・パラメーター

- on** DB2 コネクトによる、ホストまたは AS/400 データベース・サーバーでの DRDA フローのトレースをオンにします。
- off** DB2 コネクトによる、ホストまたは AS/400 データベース・サーバーでの DRDA フローのトレースをオフにします。
- i** トレース情報にタイム・スタンプを含めます。
- r** ホストまたは AS/400 サーバー・システムから受信された DRDA データ・ストリームをトレースします。
- s** ホストまたは AS/400 データベース・サーバーに送信された DRDA データ・ストリームをトレースします。
- c** ホストまたは AS/400 データベース・サーバーから受信された SQLCA をトレースします。

省略時の値は、`-r`、`-s`、および `-c` です。

### **-l=length**

トレース情報の保管に使用するバッファのサイズを指定します。省略時値は 1M で、最小値は 64K です。

### **-t=tracefile**

トレースの宛先を指定します。 *tracefile* には、ファイルまたは標準装

置の名前を指定できます。ファイル名がフルパスで指定されていない場合、省略されている部分には現行パスが使用されます。省略時のファイル名は、`ddcstrc.dmp` です。

**-p=*pid*** この処理に関する事象のみトレースします。-p が指定されていない場合、ユーザーのインスタンスに関するすべての処理が出力ファイルに書き込まれます。

**注:** リモート・クライアントについては、*pid* は、データベース・システム・モニターによって戻されるエージェント ID フィールド内で検出できます。

詳細については、109ページの『第8章 データベース・システム・モニター』を参照してください。

## トレース出力

`ddcstrc` ユーティリティーは、*tracefile* に以下の情報を書き込みます。

- -r
  - DRDA 応答 / オブジェクトのタイプ
  - 受信バッファ
- -s
  - DRDA 要求のタイプ
  - 送信バッファ
- -c
  - SQLCA
- CPI-C エラー情報
  - 受信関数の戻りコード
  - 重大度
  - 使用したプロトコル
  - 使用した API
  - 機能
  - CPI-C 戻りコード
  - エラー番号
  - 内部戻りコード
- SNA エラー情報
  - 受信関数の戻りコード
  - 重大度

- 使用したプロトコル
- 機能
- パートナー LU 名
- エラー番号
- TCP/IP エラー情報
  - 受信関数の戻りコード
  - 重大度
  - 使用したプロトコル
  - 使用した API
  - 機能
  - エラー番号

**注:**

1. 終了コードのゼロ値は、そのコマンドが正常に完了したことを示し、ゼロ以外の値は、そのコマンドが正常に完了しなかったことを示します。
2. 戻されるフィールドは、使用した API によって変わります。SNA API は、2PC SPM 接続にのみ使用されます。
3. 戻されるフィールドは、同じ API の場合でさえ、DB2 コネクトが実行しているプラットフォームによって変わります。
4. ddstrc がすでに存在しているファイルへ出力を送信した場合、ファイル上の許可により消去を禁止しているのではない限り、以前のファイルは消去されてしまいます。

## トレース出力ファイルの分析

以下のページは、出力例を示しており、DB2 コネクト・ワークステーションとホストまたは AS/400 データベース・サーバーとの間で交換されるいくつかの DRDA データ・ストリームを例示しています。ユーザーの観点からは、コマンド行プロセッサを使用して CONNECT TO データベース・コマンドを発行しています。

191ページの図10 は、APPC 接続上で DB2 コネクト エンタープライズ・エディション バージョン 7 および DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 を使用します。

192ページの図11 は、TCP/IP 接続上で DB2 コネクト エンタープライズ・エディション バージョン 7 および DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 を使用します。



```

1      DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
      pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 177
      SEND BUFFER:  EXCSAT RQSDSS      (ASCII)      (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 006AD04100010064 10410020115E8482 .j.A...d.A. . . . .}.....;db
0010 F282974040404040 4040404040404040 ...@@@@@@@@@@@@ 2bp
0020 4040F0F0F0F1F7F3 C5C3000C116DA685 @@.....m.. 000173EC..._we
0030 81A2859340400013 115AC4C2F240C396 ....@...Z...@.. ase1 ...]DB2 Co
0040 95958583A340F54B F200141404140300 .....@.K..... nnect 5.2.....
0050 0414440003240700 05240F0003000D11 ..D..$...$..... .....
0060 47D8C4C2F261F6F0 F0F00085D0010002 G....a..... .QDB2/6000.e}...
0070 007F200100162110 E2C1D56DC6D9C1D5 .. ...!...m.... .".....SAN_FRAN
0080 C3C9E2C3D6404040 40400006210F2407 .....@@@@...!$. CISCO .....
0090 000D002FD8E3C4E2 D8D3C1E2C3000C11 .../..... ....QTDSQLASC...
00A0 2EE2D8D3F0F5F0F2 F0003C210437E2D8 .....

```

図 10. トレース出力の例 (APPC 接続) (1/2)

```

3      DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
      pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 177
      SEND BUFFER:  RDBCMM RQSDSS      (ASCII)      (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 000AD00100010004 200E ..... . .}.....
4      DB2 fnc_data      gateway_drda_ar      sqljcrecv (1.35.10.81)
      pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 178
      RECEIVE BUFFER:  ENDUOWRM RPYDSS (ASCII)      (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 002BD05200010025 220C000611490004 .+.R...%"....I.. ..}.....
0010 00162110E2C1D56D C6D9C1D5C3C9E2C3 ..!...m..... ....SAN_FRANCISC
0020 D640404040400005 211501000BD00300 .@@@@...!..... 0 .....}..
0030 0100052408FF .....$.. .....
5      DB2 fnc_data      gateway_drda_ar      sqljmsca (1.35.10.108)
      pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 179
      SQLCA
      SQLCAID:  SQLCA
      SQLCABC:  136
      SQLCODE:  0
      SQLERRML: 0
      SQLERRMC:
      SQLERRP:  DSN
      SQLERRD[0->5]: 00000000, 00000000, 00000000, 00000000, 00000000, 00000000
      SQLWARN(0->A): , , , , , , , , ,
      SQLSTATE: 000000

```

図 10. トレース出力の例 (APPC 接続) (2/2)

```

1      DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 177
SEND BUFFER: EXCSAT RQSDSS      (ASCII)      (EBCDIC)
  0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 006ED04100010068 10410020115E8482 .n.A...h.A. .^.. .>}.....;db
0010 F282974040404040 4040404040404040 ...@@@@@@@@@@@@ 2bp
0020 4040F0F0F0F1F3F9 F9C5000C116DA685 @@.....m.. 0001399E..._we
0030 81A2859340400013 115AC4C2F240C396 ...@@...Z...@. ase1 ...]DB2 Co
0040 95958583A340F54B F200181404140300 .....@.K..... nnect 5.2.....
0050 0514740005240700 05240F0003144000 ..t..$...$...@. .........
0060 05000D1147D8C4C2 F261F6F0F0F00010 .....G.....a..... .....QDB2/6000..
0070 D0410002000A106D 000611A20003003C .A....m.....< }....._...s....
0080 D04100030036106E 000611A200030016 .A...6.n..... }.....>...s....
0090 2110E2C1D56DC6D9 C1D5C3C9E2C3D640 !...m.....@ ..SAN FRANCISCO
00A0 40404040000C11A1 9781A2A2A6969984 @@@@..... .....password
00B0 000A11A0A4A28599 8984009CD0010004 ..... .....userid.}...
00C0 0096200100162110 E2C1D56DC6D9C1D5 .. ...!...m.... .o.....SAN_FRAN
00D0 C3C9E2C3D6404040 40400006210F2407 .....@@@@@...!$. CISCO .....
00E0 000D002FD8E3C4E2 D8D3C1E2C3000C11 .../..... .....QTDSQLASC...
00F0 2EE2D8D3F0F5F0F2 F0003C210437E2D8 .....

```

図 11. トレース出力の例 (TCP/IP 接続) (1/2)

```

3      DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 177
SEND BUFFER: RDBCMM RQSDSS      (ASCII)      (EBCDIC)
  0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 000AD00100010004 200E ..... . ..}.....
4      DB2 fnc_data      gateway_drda_ar      sqljcrecv (1.35.10.81)
pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 178
RECEIVE BUFFER: ENDUOWRM RPYDSS (ASCII)      (EBCDIC)
  0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 002BD05200010025 220C000611490004 .+.R...%"....I.. ..}.....
0010 00162110E2C1D56D C6D9C1D5C3C9E2C3 ..!...m..... ..SAN FRANCISCO
0020 D640404040400005 211501000BD00300 .@@@@@...!..... 0 .....}...
0030 0100052408FF .....$.. .....
5      DB2 fnc_data      gateway_drda_ar      sqljmsca (1.35.10.108)
pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 179
SQLCA
SQLCAID: SQLCA
SQLCABC: 136
SQLCODE: 0
SQLERRML: 0
SQLERRMC: 0
SQLERRP: DSN
SQLERRD[0->5]: 00000000, 00000000, 00000000, 00000000, 00000000, 00000000
SQLWARN(0->A): , , , , , , , , , ,
SQLSTATE: 00000

```

図 11. トレース出力の例 (TCP/IP 接続) (2/2)

以下の情報がトレースに取り込まれます。

- クライアント・アプリケーションのプロセス ID (PID)
- データベース接続サービス (DCS) ディレクトリーでカタログされた RDB\_NAME
- DB2 コネクト CCSID (コード化文字セット識別子)
- ホストまたは AS/400 データベース・サーバーの CCSID。
- DB2 コネクト・システムの通信相手のホストまたは AS/400 データベース・サーバー管理システム。

1 番目のバッファには、ホストまたは AS/400 データベース・サーバー管理システムに送信される交換サーバー属性 (EXCSAT) およびアクセス RDB (ACCRDB) コマンドが入っています。そして、それらを CONNECT TO データベース・コマンドの結果として送信します。

2 番目のバッファには、DB2 コネクトがホストまたは AS/400 データベース・サーバー管理システムから受け取る応答が入ります。このバッファは、交換サーバー属性応答データ (EXCSATRD) およびアクセス RDB 応答メッセージ (ACCRDBRM) を含んでいます。

### EXCSAT および ACCRDB の分析

EXCSAT コマンドには、サーバー名 (SRVNAM) オブジェクトにより指定されたクライアントのワークステーション名が入っています。そのオブジェクトのコード点は X'116D' であり、DDM 仕様に従っています。EXCSAT コマンドは、1 番目のバッファにあります。EXCSAT コマンドでは、値 X'116DA68581A28593' (CCSID 500 によりコード化) は、X'116D' を除去すると、*weasel* に変換されます。

また EXCSAT コマンドには、EXTNAM (外部名) オブジェクトも含まれます。このオブジェクトは、しばしば、ホストまたは AS/400 データベース管理システムについての診断情報に入れられます。それは、20 バイトのアプリケーション ID、続いて 8 バイトのプロセス ID (または 4 バイトのプロセス ID と 4 バイトのスレッド ID) から成ります。それは、コード点 X'115E' で表され、この例ではその値は db2bp\_32 で、ブランクが埋め込まれ 0000BE5C へと続きます。UNIX ベースのデータベース・クライアントについては、この値は **ps** コマンドを使用して関連させることができ、このコマンドは活動状態のプロセスについてのプロセス状況情報を標準出力に戻します。

ACCRDB コマンドは、RDBNAM オブジェクトにある RDB\_NAME を含んでいます。そのコード点は X'2110' です。ACCRDB コマンドは、1 番目のバッファにおいて EXCSAT コマンドの後に続きます。ACCRDB コマンドでは、値 X'2110E2C1D56DC6D9C1D5C3C9E2C3D6' は X'2110' を除去すると、

SAN\_FRANCISCO に変換されます。これは、DCS ディレクトリーにあるターゲット・データベース名フィールドに対応しています。

会計ストリングのコード点は X'2104' です (64ページの『DB2 ユニバーサル・データベース (OS/390 版) でのチャージバック会計の実装』を参照してください)。

DB2 コネクト ワークステーション用に構成されたコード・セットは、ACCRDB コマンドにおいてコード点が X'119C' である CCSID オブジェクトの CCSIDSBC (1 バイト文字の CCSID) の位置により示されます。この例では、CCSIDSBC は X'0352' です。これは 850 になります。

追加のオブジェクト CCSIDDBC (2 バイト文字の CCSID) と CCSIDMBC (混合バイト文字の CCSID) がそれぞれコード点 X'119D' と X'119E' で示される場合、DB2 コネクト・ワークステーションは DBCS コード・ページ・サポート用に構成されます。出力ファイルの例では、2 つの追加コード点を組み込んでいないため、ワークステーションは DBCS 用に構成されてはいません。

**注:** TCP/IP フローには、2 つの新しいコマンドが含まれています。1 つは ACCSEC で、機密保護マネージャーにアクセスし、サポートされる機密保護メカニズムを交換するために使用されます。もう 1 つは SECCHK で、接続のエンド・ユーザーを認証するために使用される認証トークンが入っています。ACCSEC および SECCHK は TCP/IP 接続の場合だけに現れ、EXCSAT と ACCRDB の間で実行します。

### **EXCSATRD および ACCRDBRM の分析**

また CCSID 値は、ホストまたは AS/400 データベース・サーバーから、2 番目のバッファ内にあるアクセス RDB 応答メッセージ (ACCRDBRM) にも戻されます。このバッファには、EXCSATRD とそれに続く ACCRDBRM が入っています。出力ファイルの例では、ホストまたは AS/400 データベース・サーバー・システム 500 (X'01F4', SBCS CCSID) の値が含まれています。

ホストまたは AS/400 データベース・サーバーから戻ってきているコード・ページを DB2 コネクトが認識しない場合は、SQLCODE -332 がソースおよびターゲット・コード・ページと共にユーザーに戻されます。DB2 コネクトから送信されたコード・セットをホストまたは AS/400 データベース・サーバーが認識しない場合、ホストまたは AS/400 データベース・サーバーは VALNSPRM (サポートされていないパラメーター値、DDM コード点 X'1252') を戻し、それはユーザー用に SQLCODE -30073 に変換されます。

ACCRDBRM には、パラメーター PRDID (製品特定識別子、コード点は X'112E') も含まれています。値は、X'C4E2D5F0F5F0F1F0' です。この 16 進数ストリングは、EBCDIC で DSN05010 に対応します。標準によれば、DSN は DB2 (MVS/ESA 版) または DB2 ユニバーサル・データベース (OS/390 版) です。バージョンが 5.1 であることも示されています。ARI は DB2 (VSE および VM 版) で、SQL は DB2 共通サーバー、および QSQ は DB2 ユニバーサル・データベース (AS/400 版) です。

### 後続のバッファの分析

それ以降の送信バッファと受信バッファを分析して、追加情報を得ることができます。3 番目のバッファは、コミットを含んでいます。commit コマンドは、ホストまたは AS/400 データベース・サーバー管理システムが現行の作業単位をコミットするよう命令します。4 番目のバッファは、ホストまたは AS/400 データベース・サーバー管理システムから、コミットまたはロールバックの結果として受け取られます。そこには終了作業単位応答メッセージ (ENDUOWRM) が含まれ、それは現行の作業単位が終了したことを示します。この例では、4 番目のバッファは、ヌルの SQLCA を含み、DDM コード点 X'2408' とそれに続く X'FF' が示されています。ヌルの SQLCA (X'2408FF') は、成功 (SQLCODE 0) を示しています。受信バッファに SQLCA (ヌルの SQLCA もあり得ます) が入っているとき、この受信バッファの後には、SQLCA 情報の書式化された解釈を伴う ddscrc が続きます。

196ページの図12 は、受信バッファ (エラー SQLCA を含む) および書式化された SQLCA の表示の例を示しています。この SQLCA は、存在しない表から行削除を試みた結果の例です。

```

1      DB2 fnc_data      gateway_drda_ar      sqljcrecv (1.35.10.81)
      pid 48732; tid 1; node 0; cpid 0; sec 0; nsec 0; tpoint 178

      RECEIVE BUFFER:  SQLCARD OBJDSS      (ASCII)      (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 0065D0030001005F 240800FFFFFFFF34F4 .e.....$_.....4. ...}....^.....4
0010 F2F7F0F4C4E2D5E7 D6E3D34000E2C1D5 .....@.... 2704DSNXOTL .SAN
0020 6DC6D9C1D5C3C9E2 C3D64040404040FF m.....@... _FRANCISCO .
0030 FFFE0C0000000000 000000FFFFFFFFF00 .....
0040 0000000000000040 4040404040404040 .....@...
0050 4040000000FC4C4 C3E2E4E2F14BD4E8 @@.....K.. ....DDCSUS1.MY
0060 E3C1C2D3C5 ..... TABLE

```

```

2      DB2 fnc_data      gateway_drda_ar      sqljmsca (1.35.10.108)
      pid 48732; tid 1; node 0; cpid 0; sec 0; nsec 0; tpoint 179
      SQLCA

      SQLCAID:  SQLCA
      SQLCABC:  136
      SQLCODE:  -204
      SQLERRML: 15
      SQLERRMC: DDCSUS1.MYTABLE
      SQLERRP:  DSNXOTL
      SQLERRD[0->5]: FFFFFFFE0C, 00000000, 00000000, FFFFFFFF, 00000000, 00000000
      SQLWARN(0->A): , , , , , , , , , ,
      SQLSTATE: 42704

```

図 12. 受信バッファの例

---

## よくある DB2 接続問題

このセクションでは、DB2 コネクトの使用時に接続問題が生じたときの最も一般的な症状をリストします。どの場合でも、以下の形式で示されます。

- 表示されたメッセージに関連した、メッセージ番号と戻りコード (またはプロトコル固有の戻りコード) の組み合わせ。各メッセージと戻りコードの組み合わせには、個別のヘッダーがあり、このヘッダーはメッセージ番号順、その後に戻りコードの順で並べられます。
- 症状が示されます。通常は、サンプル・メッセージのリスト形式で示されます。
- 解決方法が提示されます。エラーの推定原因が示されます。場合によっては、複数の解決方法が提示されることがあります。

**注:**

1. ご使用の製品の概説およびインストールの資料や、推奨されるソフトウェア修正レベルについての最新の情報を収めた最新のリリース情報を参照してください。
2. APPC 通信に固有のメッセージと戻りコードの組み合わせの場合、SNA センス・コードも表示されます。今のところ、特定のメッセージと関連した SNA センス・コード情報は、SNA サブシステムから入手する必要があります。  
システム・ログを見ていくと、SNA センス・コードを見られます。これが当てはまるかどうかは、使用する SNA サブシステムによって異なります。状況によっては、SNA トレースをアクティブにして問題を再現し、センス・コード情報を入手しなければならないこともあります。
3. ゲートウェイという用語は、DB2 コネクト エンタープライズ・エディションを指しています。

## SQL0965 または SQL0969

### 症状

DB2 ユニバーサル・データベース (AS/400 版)、DB2 ユニバーサル・データベース (OS/390 版)、DB2 (MVS/ESA 版)、および DB2 (VM および VSE 版) からのさまざまな異なる戻りコードとともに、メッセージ SQL0965 および SQL0969 が発行されます。

いずれかのメッセージが出された場合、そのメッセージを発行したデータベース・サーバー製品の資料で、元の SQL コードを調べる必要があります。

### 解決方法

ホスト・データベースから受信された SQL コードを変換できません。そのエラー・コードに基づいて問題を訂正してから、失敗したコマンドを再発信してください。

## CONNECT 時の SQL1338

### 症状 / 原因

記号宛先名が定義されていないか、適切に定義されていません。

たとえば、これは APPC ノードを使用していて、DB2 ノード・ディレクトリに指定した記号宛先名がローカル APPC 通信サブシステム構成の CPI-C 項目と一致しない場合に、生じることがあります。

別の原因として、複数の SNA スタックをマシンにインストールしてあることもあります。PATH と LIBPATH を検査して、使用したいスタックがまず最初に参照されることを確認する必要があるかもしれません。

### 解決方法

1. DB2 ノード・ディレクトリー項目に指定した CPIC サイド情報のプロファイル名が、SNA 構成と一致するようにします (大文字小文字を区別します)。
2. PATH と LIBPATH を検査して、使用したい SNA スタックがまず最初に参照されることを確認する必要があるかもしれません。

## CONNECT 時の SQL1403N

### 症状

SQL1403N 指定されたユーザー名とパスワードのいずれか、またはその両方が正しくありません。

### 解決方法

1. DB2 コネクト・ワークステーションで認証に失敗します。ユーザーが DB2 コネクト・ワークステーションで認証されるようになっていないかを判別してください。

認証されるようになっていない場合、必要であれば CONNECT ステートメントに正しいパスワードを指定してあるか確認してください。

認証されるようになっていない場合、システム・データベース・ディレクトリー項目は、AUTHENTICATION SERVER (AUTHENTICATION を明示的に指定していない場合は、これが省略時値) を使用して不正にカタログ化されています。この場合、AUTHENTICATION DCS または CLIENT を使用して、項目を再カタログ化してください。

2. ターゲット・サーバー・データベースへ送信するときに使用できないパスワードです。システム・データベース・ディレクトリー項目が AUTHENTICATION DCS を使用してカタログされている場合は、パスワードを DB2 クライアントからターゲット・サーバー・データベースに送信する必要があります。特定のプラットフォーム、たとえば AIX などでは、パスワードは CONNECT ステートメントに指定してある場合に限り、入手することができます。



## SQL5043N

### 症状

1 つまたは複数の通信プロトコルに対するサポートが正常に開始できませんでした。ただし、コアとなるデータベース・マネージャの機能は正常に開始されました。

おそらく、TCP/IP プロトコルが DB2 コネクト・ゲートウェイで開始されていません。以前に成功したクライアント接続がまだ残っている可能性があります。

diaglevel = 4 の場合、db2diag.log に同様の項目が含まれている可能性があります。たとえば次のようになります。

```
1997-05-30-14.09.55.321092 Instance:svtldb5 Node:000
PID:10296(db2tcpcm) Appid:none
common_communication sqlcctcpconnmgr_child Probe:46
DIA3205E Socket address "30090" configured in the TCP/IP
services file and
required by the TCP/IP server support is being used by another
process.
```

### 解決方法

この警告は、DB2 コネクト (リモート・クライアントのゲートウェイとして動作している) が 1 つまたは複数のクライアント通信プロトコルを処理する際に問題が発生していることを示しています。これらのプロトコルは TCP/IP、APPC その他のものであり、このメッセージは通常、DB2 コネクトに定義されているこれらの通信プロトコルのいずれかが正しく構成されていないことを示しています。

DB2COMM プロファイル変数が定義されていないか、不正に定義されていることが原因かもしれません。一般に、問題は DB2COMM 変数と、データベース・マネージャ構成で定義した名前 (たとえば、svccname、nname、または tpname) との間のミスマッチの結果です。

可能性のあるシナリオとして 1 つあげられるのは、以前に成功した接続がそのままになっており、構成が変更されていないのに、SQL5043 エラー・メッセージを受け取ってしまったというものです。これは、TCP/IP プロトコルの使用時にリモート・システムが何らかの理由で接続を異常終了したときに発生する可能性があります。異常終了する場合、接続をクライアント上に残しておくことができるので、下記のコマンドを発行して介入しなくても、接続を復元することができます。

一番多いのは、ゲートウェイに接続しているクライアントの 1 つが TCP/IP ポート上でハンドルを持ったままになるというケースです。ゲートウェイに接続している各クライアント・マシン上で、以下を発行します。

1. db2 terminate
2. db2stop

## SQL30020

### 症状

SQL30020N 後続のコマンドまたは SQL ステートメントの正常な実行に影響を与える分散プロトコル・エラーのために、実行が失敗しました。

### 解決方法

このエラーが発生したら、サービス担当者に連絡してください。

db2dump ディレクトリーに ffdc ダンプ (pid.000) があるかどうか調べてください。ある場合は、このダンプ・ファイルを db2fdump によってフォーマットし、結果ファイルで "ERROR" を調べてください。そこに MVS ABEND がリストされている可能性があります。その場合、MVS コンソールで詳細を調べ、DB2 for MVS Messages and Codes の資料で異常終了コードを調べてください。

## SQL30060

### 症状

SQL30060N "<authorization-ID>" が、処理 "<operation>" を実行する権限を持っていません。

### 解決方法

DB2 (MVS 版) または DB2 (OS/390 版) への接続時に、通信データベース (CDB) 表が正しく更新されていません。次を参照してください。

- DB2 コネクト 概説およびインストール
- DB2 コネクティビティー補足

## SQL30061

### 症状

誤ったホストまたは AS/400 データベース・サーバー・ロケーションに接続しています。ターゲット・データベースが見つかりません。

## 解決方法

DCS ディレクトリー項目に誤ったサーバー・データベース名を指定した可能性があります。これが生じた場合、SQLCODE -30061 がアプリケーションに戻されます。

DB2 ノード、データベース、および DCS ディレクトリー項目を調べてください。DCS ディレクトリー項目の宛先データベース名のフィールドは、プラットフォームに基づいたデータベースの名前に対応していなければなりません。たとえば、DB2 ユニバーサル・データベース (OS/390 版) データベースの場合、使用する名前はブートストラップ・データ・セット (BSDS) の "LOCATION=locname" フィールドで使用した名前と同じでなければなりません。これは、分散データ機能 (DDF) を開始するときの DSNL004I メッセージにも示されています (LOCATION=location)。4ページの『データベースの概念』、および85ページの『第6章 データベース・ディレクトリーの更新』を参照してください。

ご使用の DB2 コネクト 概説およびインストールの資料には、DB2 カタログの更新方法を示す例が収められています。SNA 構成を説明する各章の『DB2 ディレクトリーの更新』を参照するか、『ホストおよび AS/400 データベースでの DB2 コネクトの構成』の章の『TCP/IP 接続の構成』を参照してください。

APPC または APPN ノードへの正しいコマンドは次のとおりです。

```
db2 catalog appc node <node_name> remote <sym_dest_name> security program
db2 catalog dcs database <local_name> as <real_db_name>
db2 catalog database <local_name> as <alias> at node <node_name>
authentication dcs
```

TCP/IP ノードへの正しいコマンドは次のとおりです。

```
db2 catalog tcpip node <node_name> remote <host_name_or_address>
server <port_no_or_service_name>
db2 catalog dcs database <local_name> as <real_db_name>
db2 catalog database <local_name> as <alias> at node <node_name>
authentication dcs
```

その後データベースへ接続するには、次のコマンドを発行します。

```
db2 connect to <alias> user <user_name> using <password>
```

## CONNECT 時の SQL30073 (戻りコード 119C)

### 症状

メッセージ SQL30073 は戻りコード 119C とともに発行されます。これが発行されるのは、ターゲット・サーバー・データベースが DB2 クライアント

(DB2 コネクトを使用する) のコード・ページをサポートしていない場合です。このコード・ページは、DB2 クライアントが実行されている操作環境の構成から派生します。

詳しくは、[管理の手引き](#) を参照してください。

### 解決方法

この問題は、ターゲット・サーバー・データベース・システムに修正ファイルをインストールすることにより、解決できることもあります。適切なサービス部門に連絡して、この症状に適した修正ファイルを入手し、それを適用してください。

一時的な作業環境として、ユーザーは DB2CODEPAGE 環境変数を設定することにより、省略時のコード・ページをオーバーライドすることができます。ロケールを調べるか、DB2CODEPAGE=850 を設定してください。

UNIX プラットフォームでは、LANG 環境変数を別の値に設定することにより、別のコード・ページに切り替えることができます。

## SQL30081N (戻りコード 1)

### 症状

症状は、次のメッセージと SNA センス・コードです。

```
db2 connect to <database name> user <userid>
Enter password for <userid>:
SQL30081N  A communication error has been detected.
Communication protocol
being used: "APPC".  Communication API being used: "CPI-C".
Location where
the error was detected: "".  Communication function detecting
the error:
"cmal1c".  Protocol specific error code(s): "1", "*",
"0x10030021".
SQLSTATE=08001
```

### 解決方法

この例では、センス・コードは 10030021 です。

このエラー・メッセージと関連する最も一般的なセンス・コード、およびそれぞれの場合の提案されている解決方法は、以下のとおりです。

1.

SQL30081N with return code 1 and sna sense code 0877002C

正しくないネットワーク名を指定しています。

2.

SQL30081N with return code 1 and SNA sense code ffff0003

正しくない MAC アドレスを指定しているか、SNA リンクがアクティブではありません。

3.

SQL30081N with return code 1 and SNA sense code 10030021

LU タイプのミスマッチがあります。

4.

SQL30081N with return code 1 and SNA sense code 084B6031

DSNZPARM の MAXDBAT (DB2 (MVS 版) または DB2 (OS/390 版) ホスト) は、0 に設定されます。

#### 他の提案

- ローカル LU プロファイルを作成しているときに、その LU を省略時の LU として定義します。たとえば、CM/2 の「SNA 機能 (SNA Feature)」リスト・パネルでは、以下のいずれかを行います。
  - 「このローカル LU を省略時 LU 別名 として使用する (Use this local LU as your default local LU alias)」チェック・ボックスにチェックマークを付ける
  - DB2 コネクト エンタープライズ・エディション・ゲートウェイ・システムのプロファイルまたは環境変数 APPCLLU をローカル LU 名に設定する。OS/2 システムでは CONFIG.SYS を編集することにより、Windows NT システムではコントロール・パネルによってこの作業を行うことができます。
- DB2 コネクト・ゲートウェイで SNA が開始していることをチェックします。
- DB2 (MVS 版) または DB2 (OS/390 版) を使用している場合、分散データ機能 (DDF) アドレス空間が開始されており、DB2 が実行中であることを確認します。

## SQL30081N (戻りコード 2)

### 症状

戻りコード 2 および SNA センス・コード 08120022 であるメッセージ SQL30081N を受け取ります。

## 解決方法

NCP (リンクのホスト端) の NUMILU パラメーターは、省略時値 (0) に設定されている可能性があります。このことを確認してください。再試行前、あるいは変更を有効にした後、必要であれば NCP 定義を変更します。

## SQL30081N (戻りコード 9)

### 症状

症状は次のメッセージです (この場合は、SNA センス・コードは必要ありません)。

```
db2 connect to <database> user <userid>
SQL30081N A communication error has been detected.
Communication protocol
being used: "APPC". Communication API being used: "CPI-C".
Location where
the error was detected: "". Communication function detecting
the error:
"cmsend". Protocol specific error code(s): "9", "*",
"0x10086021".
SQLSTATE=08001
```

### 解決方法

問題は、DB2 コネクト・システム上でトランザクション・プログラム名 (TPNAME) が正しく定義されていないことにあります。たとえば、ご使用の SNA 構成は更新してあっても、DB2 コネクト・ゲートウェイで検証されていない場合があります。詳細は、*DB2 コネクト エンタープライズ・エディション (OS/2 および Windows 版) 概説*および*インストール*、または *DB2 コネクト パーソナル・エディション 概説*および*インストール* の資料を参照してください。

## SQL30081N (戻りコード 10)

### 症状

症状は次のメッセージです (SNA センス・コードは必要ありません)。

```
SQL30081N A communication error has been detected.
Communication protocol
being used: "APPC". Communication API being used: "CPI-C".
Location where
the error was detected: "". Communication function detecting
the error:
"cmrcv". Protocol specific error code(s): "10", "*", "*".
SQLSTATE=08001
```

### 解決方法

DB2 を正しくインストールしたことを確認してください。

DB2 コネクト (OS/2 版) ゲートウェイを使用している場合は、TP 名が正しく定義されていないと次のように表示されます。

```
Protocol specific error code(s): "10", "*", "0x084C0000".  
SQLSTATE=08001
```

たとえば CM/2 では、この場合次のように定義する必要があります。

```
Transaction program name      = 'tpname'      (user defined)  
OS/2 program path and file name = notused
```

および (次の CM/2 構成画面)

```
Presentation type - background  
Operation type - Queued, operator preloaded
```

## SQL30081N (戻りコード 20)

### 症状

```
SQL30081N A communication error has been detected.  
Communication protocol  
being used: "APPC". Communication API being used: "CPI-C".  
Location where  
the error was detected: "". Communication function detecting  
the error:  
"xcstp". Protocol specific error code(s): "20", "*", "*".  
SQLSTATE=08001
```

### 解決方法

DB2 コネクト・システムで SNA サブシステムが開始していることを確認します。

## SQL30081N (戻りコード 27)

### 症状

戻りコード 27 および SNA センス・コード 800Axxxx であるメッセージ SQL30081N を受け取ります。

### 解決方法

VTAM パス情報単位 (PIU) が大きすぎます。

## SQL30081N (戻りコード 79)

### 症状

```
SQL30081N A communication error has been detected.  
Communication protocol  
being used: "TCP/IP". Communication API being used: "SOCKETS".
```

```
Location
where the error was detected: ". Communication function
detecting the error:
"connect". Protocol specific error code(s): "79", "*", "*".
SQLSTATE=08001
```

## 解決方法

このエラーは、リモート・クライアントが DB2 コネクト・ゲートウェイへの接続に失敗した場合に発生する可能性があります。さらに、DB2 コネクト・ゲートウェイからホストへの接続時にも発生することがあります。

1. DB2COMM プロファイル変数が、DB2 コネクト・ゲートウェイで正しく設定されていない可能性があります。このことを確認してください。たとえば、DB2 エンタープライズ拡張エディションを AIX 上で実行している場合は、コマンド `db2set db2comm=tcPIP` が `sqllib/db2profile` に存在していなければなりません。
2. DB2 クライアントと DB2 コネクト・ゲートウェイで、TCP/IP サービス名またはポート番号 (あるいはその両方) の仕様の間にミスマッチがある可能性があります。両方のマシンで、TCP/IP `services` ファイル内の項目を確認してください。
3. DB2 コネクト・ゲートウェイで DB2 が開始していることをチェックします。次のコマンドを使用して、データベース・マネージャー構成の `diaglevel` を 4 に設定してください。

```
db2 update dbm cfg using diaglevel 4
```

DB2 を停止して再始動したら、`db2diag.log` ファイルを参照して、DB2 TCP/IP 通信が開始していることをチェックします。次のような出力が含まれているはずです。

```
1998-02-03-12.41.04.861119 Instance:svtdbm2 Node:00
PID:86496(db2sysc) Appid:none
common_communication sqlcctcp_start_listen Probe:80
DIA3000I "TCPIP" protocol support was successfully started.
```

## SQL30081N (プロトコル固有のエラー・コード 10032)

### 症状

```
SQL30081N A communication error has been detected.
Communication protocol
being used: "TCP/IP". Communication API being used: "SOCKETS".
Location
where the error was detected: "9.21.85.159". Communication
function detecting
the error: "send". Protocol specific error code(s): "10032",
"*, "*".
SQLSTATE=08001
```



## 解決方法

このエラー・メッセージは、TCP/IP 通信に失敗したマシンから切断しようとするときに受け取ることがあります。TCP/IP サブシステムの問題を修正してください。

問題を修正する方法は、ほとんどのマシンでは、そのマシンの TCP/IP プロトコルを再始動することになります。マシン全体を再生しなければならないこともあります。



---

## 第3部 付録および後付け



---

## 付録A. 以前のリリースで追加された機能

このセクションでは、最新のバージョン / リリースから順に、各バージョンおよびリリースで追加された機能の要約を示します。

---

### DB2 コネクト バージョン 6 リリース 1

DB2 コネクト バージョン 6.1 では、次のように機能が拡張されました。

- TCP/IP 通信プロトコルの使用
- 2 フェーズ・コミット
- 複数行のストアード・プロシージャ
- DCE 機密保護
- DCE セル・ディレクトリー・サポートおよびホスト・システム
- 拡張された機密保護障害通知
- System/390 SYSPLEX の高度な利用
- ODBC および JDBC アプリケーションに対する最適化されたカタログ・アクセス
- 新しい BIND オプション
- Microsoft Transaction Server サポート
- 簡単になったパスワード管理
- クライアント情報の機能強化
- 両方向言語のサポート
- DB2 コネクト・アプリケーションのモニター
- 2 フェーズ・コミットの機能強化
- 簡単になった DB2 同期点管理プログラムの構成
- 追加のデータ・オブジェクトとタイプのサポート
- パーソナル・コミュニケーション用の DB2 コネクト

---

## DB2 コネクト バージョン 5 リリース 2

DB2 コネクト バージョン 5.2 では、ホストおよび AS/400 DRDA 機能が次のように拡張されました。

- DCE セル・ディレクトリー・サポート
- 拡張された機密保護障害通知
- System/390 SYSPLEX の高度な利用
- ODBC および JDBC アプリケーションに対する最適化されたカタログ・アクセス
- Microsoft Transaction Server サポート
- 新しい BIND オプション (DYNAMICRULES)
- クライアント情報設定 API 拡張機能
- DB2 コネクトに対する SQLDescribeParam サポート
- 両方向言語のサポート
- システム・モニターの機能強化
- 2 フェーズ・コミット・サポートの機能強化
- 簡単になった DB2 同期点管理プログラムの構成
- SCO\*\* オペレーティング・システムのサポート
- 大整数、ラージ・オブジェクト、行 ID、ユーザー定義特殊データ・タイプのサポート

---

## DB2 コネクト バージョン 5.0

- より簡単なパッケージの購入:
  - 単一の DB2 コネクト パーソナル・エディション・パッケージには、プロダクトの OS/2、Windows 3.1、Windows 95、および Windows NT バージョンが含まれています。このパッケージには、Lotus Approach の補足的なコピーを含め、開始するために必要なものすべてが入っています。
  - 単一の DB2 コネクト エンタープライズ・エディション・パッケージには、OS/2、Windows NT、およびすべての UNIX バージョンが含まれています。
- 機能:
  - 新しいレベル 3 ODBC ドライバーには、改良された点がたくさんあります。
  - JDBC ドライバーが更新され、Java のより良いサポートが可能となりました。

- 複数行の結果セットおよび複数の結果セットを戻すストアード・プロシージャをサポートしています (DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以上が必要)。
- 複製サポートが組み込まれています。
- 総称バインド・オプションがあります。ホスト・データベースがサポートするバインド・オプションを指定することができます。
- SYSPLEX を利用できます (DB2 コネクト パーソナル・エディションのみ。DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以上が必要)。
- 使用可能度:
  - インストールの方式が新しくなりました。
  - TCP/IP データベース接続は構成が簡単です (DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以上、または DB2 ユニバーサル・データベース (AS/400 版) バージョン 4.2 が必要)。
  - ポイント・アンド・クリック構成を使った統合 SNA サポートを利用することができます (DB2 コネクト パーソナル・エディションのみ)。
  - ホスト接続を構成するのに、新しいポイント・アンド・クリック構成ユーティリティを使用できます。
  - デスクトップ・クライアント・システムから DB2 コネクト エンタープライズ・エディション・サーバーに、より簡単に接続できます。クライアントは、DB2 コネクト・サーバーと、各サーバーで定義されているすべてのデータベースを検出することができます。
  - ODBC トレース機能が向上し、パフォーマンス分析に必要な情報が詳細になりました。
  - コントロール・センターおよびその他の GUI ツールにより、いくつかの DBA タスクが単純化されました。
- 機密保護:
  - DCE 機密保護 (DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以上が必要)
  - 基本テーブルに対して各ユーザーの許可を行わなくても、ODBC アプリケーションを実行できます。したがって、ODBC ドライバーをバインドした人の権限の下でアプリケーションを実行するように、各ユーザーは、ODBC ドライバーをバインドすることができます。
- パフォーマンス:
  - ODBC アプリケーションの DB2 カタログへのアクセスがより高速になりました。

- ネットワーク通信量が削減されました。
  - カーソルのための早期クローズ
  - 据え置き準備
  - 複合 SQL のバイト・カウントの削減
  - 複数の他のネットワーク・フローの拡張
  - ホストでの ASCII 格納がサポートされています (DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以上が必要)。
- 接続性:
  - 他の IBM DRDA アプリケーション・サーバーが TCP/IP のサポートを導入する際に、これらのアプリケーション・サーバーに TCP/IP を介して接続する DRDA をサポートします。
  - 統合 MPTN サポートを介する、TCP/IP 準拠の SNA があります (ホスト上に AnyNet が必要)。
  - 追加 SNA 接続性オプションのサポートがあります。
    - IBM Communications Server for Windows NT
    - IBM Personal Communications
- その他:
  - TCP/IP で 2 フェーズ・コミット・トランザクションを開始する機能 (DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以上が必要)
  - デスクトップ・アプリケーションがゲートウェイを必要とせずに 2 フェーズ・コミット・トランザクションに参加するための機能 (TCP/IP のみ。DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以上が必要)。
  - DB2 ユニバーサル・データベース (OS/390 版) でトランザクション調整の信頼性を高めるための機能 (TCP/IP および DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以上が必要)。
  - その他にシステム・パフォーマンス、信頼性、および使用可能度のあらゆる分野に関係する機能強化や修正が数多くあります。

---

## DDCS バージョン 2 リリース 4

Windows 単一ユーザー用分散データベース接続サービス (DDCS) バージョン 2.4 には、次のような機能が導入されました。

- データ・ソース・セットアップ・ツール。これにより、ホストおよび AS/400 サーバーへの接続を迅速かつ簡単に定義できます。



- Wall Data Rumba。これらの接続を確立するのに必要な通信サポートを提供します。
- DB2 パスワード有効期限保守 (DB2 Password Expiration Maintenance (DB2PEM)) ユーティリティー。これにより、TSO にログオンせずに DB2 (MVS/ESA 版) のパスワードを変更することができます。
- DB2 コネクトのパフォーマンスおよび柔軟性を向上させる機能強化として、次のようなものがあります。
  - 据え置き準備。ODBC および他の動的 SQL アプリケーションのパフォーマンスを向上させます。これは、PREPARE 要求と後続の要求を別々に送信するのではなく、PREPARE 要求と後続の要求をつなげて送信することによって可能となります。
  - 非同期 ODBC。ODBC アプリケーションの使用可能性を向上させます。以前は、あるネットワークでは、長い照会を処理していると、ODBC アプリケーションが遅れがちになることがありました。
  - AIX および OS/2 での、マルチスレッド・アプリケーションのサポート。これにより、非 ODBC アプリケーションが、自分のコンテキストで、複数のデータベース接続を維持できるようになりました。

---

## DDCS バージョン 2 リリース 3

DDCS バージョン 2 リリース 3.1 の新規の機能には次のようなものがあります。

- OS/2 および AIX で LU6.2 同期点管理プログラム (Syncpoint Manager、SPM) を使用して行う、DRDA 接続用 2 フェーズ・コミット。

DDCS バージョン 2 リリース 3.0 の新規の機能には次のようなものがあります。

- DB2 (MVS/ESA 版) バージョン 4.1 および DB2 ユニバーサル・データベース (AS/400 版) バージョン 3.1 サーバーでストアド・プロシージャを実行すると、クライアント・アプリケーションのパフォーマンスは向上します。57ページの『ストアド・プロシージャ』を参照してください。
- 単一トランザクションで複数のデータベースを処理することができます。
- SQL ステートメントを連結することによって、パフォーマンスを改善することができます。61ページの『NOT ATOMIC 複合 SQL』および 122ページの『インポートおよびエクスポート・ユーティリティーの使用』を参照してください。

- 会計文字列を使用してチャージバック会計を実施することができます。64ページの『DB2 ユニバーサル・データベース (OS/390 版) でのチャージバック会計の実装』を参照してください。
- アプリケーションを DRDA アプリケーション・サーバーにバインドするときに、多くの新規バインド・オプションを使用することができます。107ページの『BIND コマンド』を参照してください。
- DCE ディレクトリーを使用することにより、すべてのクライアントに必要とされるディレクトリー情報を、中央リポジトリー内に統合することができます。223ページの『付録D. DCE ディレクトリー・サービスの使用』を参照してください。
- SQLCODE 処理を一層柔軟に行うことができます。137ページの『第11章 SQLCODE マッピング』を参照してください。
- 診断情報は、読み取り可能なフォーマットで保管され、単一のロケーション(基本障害保守ログ)に統合されています。詳細については、*問題判別の手引き* を参照してください。
- DDCSSETP 環境変数は、SQLERROR CONTINUE などの BIND および PREPARE オプションによって置き換えられています。これにより、操作は単純になります。
- パフォーマンスを向上させるさまざまな機能が組み込まれています。

## 付録B. ディレクトリー・カスタマイズ・ワークシート

このワークシートを使用して、ディレクトリーをカスタマイズします。コマンド構文については、97ページの『ディレクトリーの更新』またはコマンド解説書を参照してください。

表9. ノード・ディレクトリー・パラメーター

パラメーター	例	ユーザー値
ノード名	DB2NODE または MVSIPNOD	
記号宛先名 (APPC ノード)	DB2CPIC	
リモート・ホスト名 (TCP/IP ノード)	MVSHOST	
サーバー (TCP/IP サービス名またはポート番号)	db2inst1c (または 446)	
機密保護タイプ	APPC ノードには PROGRAM、 TCP/IP ノードには NONE	
注:		
1. DRDA 用の省略時 TCP/IP ポート番号は 446 です。		
2. ホストまたは AS/400 データベース・サーバーが SECURITY SOCKS をサポートしていると分かっているのでない限り、TCP/IP ノードに SECURITY を指定しないでください。		

表10. DCS ディレクトリー・パラメーター

パラメーター	例	ユーザー値
データベース名	DB2DB	
宛先データベース名	NEW_YORK3	
アプリケーション・リクエスト		
パラメーター・ストリング	" ,,,,,LOCALDATE=¥/¥"YYMMDD¥/¥/¥"	

表11. システム・データベース・ディレクトリー・パラメーター

パラメーター	例	ユーザー値
データベース名	DB2DB	

表 11. システム・データベース・ディレクトリー・パラメーター (続き)

パラメーター	例	ユーザー値
データベース別名	NYC3	
ノード名	DB2NODE	
認証	DCS	

---

## 付録C. 各国語サポートの考慮事項

DB2 コネクトの各国語サポート (NLS) については以下の考慮事項があります。

- DB2 コネクト・メッセージは特定の言語に翻訳されます。翻訳されたメッセージへのアクセスに関する情報については、ご使用のプラットフォーム用の概説およびインストールブックを参照してください。
- DB2 コネクトは、多数の言語およびコード・ページをサポートします。これらのコード・ページのリストについては、管理の手引きを参照してください。
- データが DB2 コネクトとホストまたは AS/400 データベース・サーバーとの間で転送される場合は、通常ワークステーション・コード・ページからホスト CCSID へ (およびその逆に) 変換します。

以下の点を含む DB2 コネクトの使用に関する詳細は、DB2 コネクト 概説およびインストールに記載されています。

- 日付と時刻の形式。
- DB2 コネクト エンタープライズ・エディションおよび DB2 コネクト パーソナル・エディションでどの言語がサポートされているか。
- ユーザーが使用する特定の言語環境に応じて DB2 コネクトをカスタマイズする方法。
- ホスト・コード化文字セット ID (CCSID) の設定をカスタマイズする方法。

---

### 文字データの変換

文字データがマシン間で転送されるときは、受信する側のマシンが使用できる形に変換されなければなりません。

たとえば、データが DB2 コネクト・ワークステーションとホストまたは AS/400 データベース・サーバーとの間で転送される場合は、通常ワークステーション・コード・ページからホスト CCSID へ (およびその逆に) 変換します。2 つのマシンが異なるコード・ページまたは CCSID を使用している場合、コード・ポイントは 1 つのコード・ページまたは CCSID から別のものへとマップします。この変換は、常に受信側で実行されます。

データベースへ送られる文字データは、SQL ステートメントと入力データから成ります。データベースから送られる文字データは、出力データから成りません。ビット・データとして解釈される出力データ (たとえば、FOR BIT DATA 文節で宣言された列からのデータ) は、変換されません。その他の場合は、すべての入力および出力の文字データは、2 つのマシンが異なるコード・ページまたは CCSID を持つときは変換されます。

たとえば、DB2 ユニバーサル・データベース (OS/390 版) データにアクセスするために DB2 コネクトを使用する場合、次のことが起こります。

1. DB2 コネクトは、OS/390 へ SQL ステートメントと入力データを送信します。
2. DB2 ユニバーサル・データベース (OS/390 版) は、それらを EBCDIC CCSID に変換して処理します。
3. DB2 ユニバーサル・データベース (OS/390 版) は、結果を DB2 コネクト・ワークステーションに戻します。
4. DB2 コネクトは、結果を ASCII または ISO コード・ページに変換してそれをユーザーに戻します。

以下の表は、コード・ページ (ワークステーション側) と CCSID (ホスト側) との間でサポートされている変換を示します。サポートされているコード・ページ変換については、[管理の手引き](#) を参照してください。

表 12. ワークステーション・コード・ページからホスト CCSID への変換

ホスト CCSID	コード・ページ	国
037、 273、 277、 278、 280、 284、 285、 297、 500、 871、 1140-1149	437、 819、 850、 858、 860、 863、 1004、 1051、 1252、 1275	アルバニア、オーストラリア、オーストリア、ベルギー、ブラジル、カナダ、デンマーク、フィンランド、フランス、ドイツ、アイスランド、アイルランド、イタリア、ラテンアメリカ、オランダ、ニュージーランド、ノルウェー、ポルトガル、南アフリカ、スペイン、スウェーデン、スイス、英国、米国
423、 875	737、 813、 869、 1253、 1280	ギリシャ

表 12. ワークステーション・コード・ページからホスト CCSID への変換 (続き)

ホスト CCSID	コード・ページ	国
870	852、 912、 1250、 1282	クロアチア、チェコ共和国、ハンガリー、ポーランド、ルーマニア、セルビア / モンテネグロ (ローマ字)、スロバキア、スロベニア
1025	855、 866、 915、 1251、 1283	ブルガリア、FYR マケドニア、ロシア、セルビア / モンテネグロ (キリル文字)
1026	857、 920、 1254、 1281	トルコ
424	862、 916、 1255	イスラエル - 下記の注 3 を参照
420	864、 1046、 1089、 1256	アラブ諸国 - 下記の注 3 を参照
838	874	タイ
930、 939、 5026、 5035	932、 942、 943、 954、 5039	日本
937	938、 948、 950、 964	台湾
933、 1364	949、 970、 1363	韓国
935、 1388	1381、 1383、 1386	中華人民共和国
1112、 1122	921、 922	エストニア、ラトビア、リトアニア
1025	915、 1131、 1251、 1283	ベラルーシ
1123	1124、 1125、 1251	ウクライナ

**注:**

1. コード・ページ 1004 はコード・ページ 1252 としてサポートされます。
2. 一般に、データは、コード・ページから CCSID へ変換することができ、再度元へ戻してもなんら変化はありません。以下は、その規則に対する例外です。
  - 2 バイト文字セット (DBCS) コード・ページにおいて、ユーザー定義の文字を含むいくつかのデータは失われる可能性があります。
  - 混合バイトのコード・ページ内で定義された単一バイト・コード・ページと新規のいくつかの単一バイト・コード・ページの場合、存在していない

文字はどちらもソースで、ターゲットは置換文字へマップされてから、データが元のコード・ページへ再変換される時に失われる可能性があります。

3. 両方向言語の場合は、特殊な "BiDi CCSIDS" が IBM によっていくつか定義されており、DB2 コネクト でサポートされています。

データベース・サーバーの両方向属性がクライアントの属性と異なっている場合は、これらの特殊な CCSIDS を使ってこの相違を解決できます。

これらの特殊な CCSID の詳細については、[管理の手引き](#) を参照してください。これらの属性を DRDA ホスト接続用に設定する方法については、[DB2 コネクトのリリース・ノート](#)を参照してください。



---

## 付録D. DCE ディレクトリー・サービスの使用

DCE セル・ディレクトリー・サービス (CDS) を利用して、サーバー情報を各クライアント上に保管する代わりに、CDS に保管することができます。CDS は、すべてのプラットフォーム上のすべての DB2 ユニバーサル・データベース・クライアントと DB2 コネクトエンタープライズ・エディションでサポートされています。

**注:** DB2 コネクトの DCE セル・ディレクトリー・サービス・サポートを使用して、SNA 接続によって DB2 (MVS/ESA 版) に接続したい場合、DB2 (MVS/ESA 版) PTF UN73393 を適用する必要があります。それは、リモート・トランザクション・プログラム名 (RTPN) として DB2DRDA の使用をサポートします。

DCE ディレクトリーを使用したい場合は、以下のオブジェクトを作成します。

- データベース・オブジェクト、これはデータベースについての情報を含みます。
- データベース・ロケーター・オブジェクト、これはリモート・クライアントと DB2 コネクト・ワークステーションとの間の接続についての情報を含みます。
- 経路指定情報オブジェクト、これはデータベース・オブジェクトをデータベース・ロケーター・オブジェクトと一致させるものです。

アクセスする各ホストまたは AS/400 データベース・サーバーについて、これらのオブジェクトを作成する前に次のことを行ってください。

- 以下の DCE 属性が、オブジェクトの作成元のワークステーション上の cds 属性のファイルに追加されていることを確認してください。

### AIX システムの場合

ファイル名は `/etc/dce/cds_attributes` です。

### OS/2 システムの場合

ファイル名は `x:%opt%dcelocal%etc%cds_attr` です。ここで、`x:` はドライブ名です。

### Windows 32 ビット・システムの場合

ファイル名は `x:%root%dcelocal%etc%cds_attributes` です。ここで、`x:` はドライブ名、`root` は DCE をインストールするディレクトリーです。

1.3.18.0.2.4.30	DB_Comment	char
1.3.18.0.2.4.31	DB_Communication_Protocol	char
1.3.18.0.2.4.32	DB_Database_Protocol	char
1.3.18.0.2.4.33	DB_Database_Locator_Name	char
1.3.18.0.2.4.34	DB_Native_Database_Name	char
1.3.18.0.2.4.35	DB_Object_Type	char
1.3.18.0.2.4.36	DB_Product_Name	char
1.3.18.0.2.4.37	DB_Product_Release	char
1.3.18.0.2.4.38	DB_Target_Database_Info	char
1.3.18.0.2.4.39	DB_Authentication	char
1.3.18.0.2.4.63	DB_Principal	char

- オブジェクトを作成する十分な権限を持って DCE にログインしていることを確認してください。次の DCE コマンドを UNIX または Windows 2000 システム上でログインに使用できます。

```
dce_login principal-id password
```

次の DCE コマンドを OS/2 システム上でログインに使用できます。

```
dcelogin principal-id password
```

- 注:** これらのオブジェクトを使用してデータベースを接続する前に、ホストまたは AS/400 データベース・サーバー上とワークステーション上の通信の構成も行ってください。この情報については、適切な概説およびインストールの手引き で説明されています。

---

## データベース・オブジェクトの作成

データベース・オブジェクトは、DB2 コネクトに対してホストまたは AS/400 データベース・サーバーを定義します。アクセスする各ホストまたは AS/400 データベース・サーバーについて、DCE コマンド **cdscp create object** を使用して、データベース・オブジェクトを作成します。たとえば、次のようにします。

```
cdscp create object database_global_name
```

以下の属性をオブジェクトに加えます。

### **DB\_Object\_Type**

データベースを表す D

### **DB\_Product\_Name**

リレーショナル・データベース製品。たとえば、DB2\_for\_MVS、または DB2\_for\_OS390。

### **DB\_Native\_Database\_Name**

ホストまたは AS/400 データベース・サーバー・システム上のデータベース名は、以下のとおりです。

## **MVS または OS/390**

LOCATION 値

## **VSE または VM**

データベース名

## **OS/400**

関係データベース名

## **DB\_Database\_Protocol**

DRDA

## **DB\_Authentication**

SERVER、CLIENT、または DCE のどれか。(231ページの『DCE ディレクトリー・サービスと機密保護』で述べられているとおり。)

## **DB\_Principal**

認証方式が DCE の場合、この属性に DCE Principal を入力してください。

## **DB\_Communication\_Protocol**

DB2 コネクト・サーバーとホストまたは AS/400 データベース・サーバーとの間の通信プロトコルに関する以下の情報。

- 通信プロトコルが APPC の場合
  1. 通信プロトコル (APPC)
  2. ホストまたは AS/400 データベース・サーバーのネットワーク ID
  3. ホストまたは AS/400 データベース・サーバーの LU 名
  4. ホストまたは AS/400 データベース・サーバーへの接続用のトランザクション・プログラム名。DB2 (MVS/ESA 版) の場合、DB2DRDA を指定します。その他のオペレーティング・システムについては、有効な値 (16 進形式ではない) を指定します。
  5. モード名
  6. 機密保護のタイプ (231ページの『DCE ディレクトリー・サービスと機密保護』で述べられているとおり)。たとえば、次のようにします。

APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;PROGRAM

- 通信プロトコルが TCPIP の場合
  1. 通信プロトコル (TCPIP)
  2. 宛先 TCP/IP ホスト名 (ホストまたは AS/400 データベース・サーバー用)

### 3. TCP/IP ポート番号

4. 接続のタイプ (SOCKS または NONE を使用しているかどうか)。これは任意指定です。指定しない場合は NONE が使用されます。たとえば、以下に挙げるような通信プロトコル TCP/IP 用の属性値があります。

```
tcip;jaguar;19713;NONE
```

システム機密保護付きのデータベース・オブジェクトを作成するには、ファイル内に以下のような命令を加えることができます。

```
create object /.../cdscell1/subsys/database/DBMVS01
add object /.../cdscell1/subsys/database/DBMVS01 DB_Object_Type=D
add object /.../cdscell1/subsys/database/DBMVS01 DB_Product_Name=DB2_for_MVS
add object /.../cdscell1/subsys/database/DBMVS01 DB_Database_Protocol=DRDA
add object /.../cdscell1/subsys/database/DBMVS01 DB_Native_Database_Name=¥
NEW_YORK
add object /.../cdscell1/subsys/database/DBMVS01 DB_Authentication=SERVER
add object /.../cdscell1/subsys/database/DBMVS01 DB_Communication_Protocol=¥
APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;PROGRAM
```

その後、次のコマンドを入力します。

```
cdscp < filename
```

**注:** ファイルの中で、ステートメントを次の行に続けたいときはいつでも、円記号 (¥) を指定してください。

DCE 機密保護付きのデータベース・オブジェクトを作成するには、ファイル内に以下のような命令を加えることができます。

```
create object /.../cdscell1/subsys/database/DBMVS02
add object /.../cdscell1/subsys/database/DBMVS02 DB_Object_Type=D
add object /.../cdscell1/subsys/database/DBMVS02 DB_Product_Name=DB2_for_MVS
add object /.../cdscell1/subsys/database/DBMVS02 DB_Database_Protocol=DRDA
add object /.../cdscell1/subsys/database/DBMVS02 DB_Native_Database_Name=¥
NEW_YORK
add object /.../cdscell1/subsys/database/DBMVS02 DB_Authentication=DCE
add object /.../cdscell1/subsys/database/DBMVS02 DB_Principal=¥
/.../cdscell1/principal_name
add object /.../cdscell1/subsys/database/DBMVS02 DB_Communication_Protocol=¥
APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;NONE
```

その後、次のコマンドを入力します。

```
cdscp < filename
```

---

## データベース・ロケータ・オブジェクトの作成

データベース・ロケータ・オブジェクトは、DB2 コネクト エンタープライズ・エディションをそのクライアントに対して定義するために使用します。DB2 コネクト・ワークステーションについて、DCE コマンド **cdscp create object** を使用して、データベース・ロケータ・オブジェクトを作成します。たとえば、次のようにします。

```
cdscp create object object_global_name
```

以下の属性をオブジェクトに加えます。

### DB\_Object\_Type

ロケータ・オブジェクトを表す L

### DB\_Communication\_Protocol

以下に、DB2 コネクト・ワークステーションとリモート・クライアントとの間でそれぞれの通信プロトコルを設定するのに必要な情報を提供します。

プロトコル・サポートをプラットフォーム別にまとめると、以下のようになります。

- OS/2: APPC、IPX、NETBIOS、および TCP/IP
- Windows 32 ビット オペレーティング・システム:  
APPC、IPX、NETBIOS、NPIPE、および TCP/IP
- UNIX: APPC と TCP/IP

APPC の場合

1. 通信プロトコル (APPC)
2. DB2 コネクト・ワークステーションのネットワーク ID
3. DB2 コネクト・ワークステーションの LU 名
4. リモート・クライアントからの接続用のトランザクション・プログラム名
5. モード名
6. 機密保護のタイプ (231ページの『DCE ディレクトリー・サービスと機密保護』で述べられているとおり)。

TCP/IP の場合

1. 通信プロトコル (TCP/IP)
2. DB2 コネクト・ワークステーションのホスト名
3. リモート・クライアントからの接続を受け入れるために DB2 コネクト・ワークステーションで使用される接続ポート

4. 接続のタイプ (SOCKS の使用または NONE)。これは任意指定です。指定しない場合は NONE が使用されます。

#### IPX/SPX の場合

1. 通信プロトコル (IPXSPX)
2. ファイル・サーバー名。直接アドレス指定には \* を使用してください。
3. オブジェクト名。直接アドレス指定には、IP アドレスを使用してください。

#### 名前付きパイプの場合

1. 通信プロトコル (NPIPE)
2. DB2 コネクト・ワークステーションのコンピューター名
3. インスタンス名

#### NETBIOS の場合

1. 通信プロトコル (NETBIOS)
2. サーバーまたは DB2 コネクト エンタープライズ・エディション・ゲートウェイの NNAME

たとえば、ファイルに以下の行を組み込むことができます。

```
create object /.../cdscell11/subsys/database/DBAIX01
add object /.../cdscell11/subsys/database/DBAIX01 DB_Object_Type= L
add object /.../cdscell11/subsys/database/DBAIX01 DB_Communication_Protocol=¥
TCPIP;AIX001;3700
add object /.../cdscell11/subsys/database/DBAIX01 DB_Communication_Protocol=¥
APPC;SPIFNET;NYX1GW01;NYSERVER;IBMRDB;NONE
```

その後、次のコマンドを入力します。

```
cdscp < filename
```

Windows 32 ビット オペレーティング・システムの場合は、似たような方法で名前付きパイプを指定できます。たとえば、次のようにします。

```
add object /.../cdscell11/subsys/database/DBAIX01 DB_Communication_Protocol=¥
NPIPE;computer_name;instance_name
```

OS/2 では、DB\_Communication\_Protocol 属性でプロトコルを指定できます。たとえば、次のようにします。

- IPXSPX;fileserver;objectname
- NETBIOS;nname

---

## 経路指定情報オブジェクトの作成

経路指定情報オブジェクトは、DCEにおいて定義する必要があり、DB2クライアントにより検索されます。DCEコマンド **cdscp create object** を使用して、経路指定情報オブジェクトを作成します。たとえば、次のようにします。

```
cdscp create object object_global_name
```

**DB\_Object\_Type** 属性の R を追加します。

各データベース・オブジェクトごとに、1つの **DB\_Target\_Database\_Info** 属性を追加してください。各 **DB\_Target\_Database\_Info** 属性は、以下のパラメーターで構成されています。

### データベース

データベース・オブジェクト名。全パスを含みます。明白に指定されていない他のすべてのデータベースを示すには、\*OTHERDBS を指定します。

### アウトバウンド・プロトコル

ホストまたは AS/400 データベース・サーバー用のデータベース・プロトコル (DRDA)

### インバウンド・プロトコル

リモート・クライアント接続用のデータベース・プロトコル (DB2RA)

### ゲートウェイでの認証

0 (No) または 1 (Yes)。(231ページの『DCE ディレクトリー・サービスと機密保護』の記述どおり)

### ゲートウェイのパラメーター・ストリング

ゲートウェイ内で使用されるパラメーターを含むストリング。その内容はゲートウェイ固有のもので、DB2 コネクト・ゲートウェイの特定ストリングについては、87ページの『DCS ディレクトリー』を参照してください。

### データベース・ロケーター

DB2 コネクト・ワークステーションを表すデータベース・ロケーター・オブジェクトの名前。

たとえば、ファイルに以下の行を組み込むことができます。

```
create object /.../cdscell11/subsys/database/ROUTE1
add object /.../cdscell11/subsys/database/ROUTE1 DB_Object_Type=R
add object /.../cdscell11/subsys/database/ROUTE1 DB_Target_Database_Info=¥
/.../cdscell11/subsys/database/DBMVS01;DRDA;DB2RA;0;;¥
```

```
../cdscell11/subsys/database/DBAIX01  
add object ../cdscell11/subsys/database/ROUTE1 DB_Target_Database_Info=¥  
*OTHERDBS;DRDA;DB2RA;0;;¥  
../cdscell11/subsys/database/DBAIX02
```

その後、次のコマンドを入力します。

```
cdcsp < filename
```

---

## 構成パラメーターの設定

クライアントのデータベース・マネージャーの構成を、次のように更新します。

```
DB2 UPDATE DATABASE MANAGER CONFIGURATION USING  
[DIR_PATH_NAME path]  
DIR_OBJ_NAME loc_obj  
DIR_TYPE DCE  
[ROUTE_OBJ_NAME route_obj]  
[DFT_CLIENT_COMM protocol]  
[DFT_CLIENT_ADPT 0-15]
```

ここで、

- *path* は省略時のパスで、ターゲット・データベースの完全な名前を形成するために使用されます (省略時値は `./:/subsys/database/`)。
- *loc\_obj* は DCE 名前スペース内のクライアントを識別します。
- `DIR_TYPE DCE` は、DCE ディレクトリーがクライアント・アプリケーションによって使用されることを指定します。
- *route\_obj* は経路指定情報オブジェクトです (たとえば、ROUTE1)。
- プロトコル は、クライアントと DB2 コネクト・ワークステーションとの間の通信プロトコル (UNIX には APPC または TCPIP、OS/2 には APPC、IPXSPX、NETBIOS、または TCP/IP、Windows 32 ビット オペレーティング・システムには APPC、TCPIP、IPXSPX、NETBIOS、または NPIPE) です。
- 省略時クライアント・アダプター (NetBIOS に応じて 0 から 15 まで)。プロトコルが NetBIOS で、クライアント・アダプター番号が省略時値 0 でない場合、クライアント・アダプター番号を指定してください。

**注:** 以下の環境変数は、上にリストされたものをそれぞれ上書きすることができます。

- DB2DIRPATHNAME は、DIR\_PATH\_NAME を上書きできます。
- DB2ROUTE は、ROUTE\_OBJ\_NAME を上書きできます。
- DB2CLIENTCOMM は、DFT\_CLIENT\_COMM を上書きできます。



- DB2CLIENTADPT は、DFT\_CLIENT\_ADPT を上書きできます。

---

## データベースのカタログ作成

あるデータベースが省略時値とは異なるパス内にある場合、またはそのデータベース名と異なる別名を使用したい場合は、グローバル・データベースをカタログ化することができます。コマンド行プロセッサで CATALOG GLOBAL DATABASE コマンドを以下のように使用できます。

```
db2 CATALOG GLOBAL DATABASE database_global_name
    AS alias USING DIRECTORY DCE
```

別名は、データベースにアクセスするどのアプリケーション・プログラムによっても使用することができます。

たとえば、次のようにします。

```
db2 CATALOG GLOBAL DATABASE /.../cdscele112/subsys/database/dbmvs12 AS NYC3
    USING DIRECTORY DCE
```

---

## DCE ディレクトリー・サービスと機密保護

DB2 コネクト管理者として、ユーザー名とパスワードを検証する場所を決めることができます。DCE ディレクトリーを使用して、以下のように設定することによりこれを行うことができます。

- DB2 コネクト・ワークステーションを示すデータベース・ロケーター・オブジェクト内の通信プロトコルの機密保護タイプ。機密保護タイプは NONE を使用してください。
- データベース・オブジェクト内の認証タイプ
- データベース・オブジェクト内の通信プロトコルの機密保護タイプ
- 経路指定情報オブジェクト内の `authenticate at gateway` パラメーター

232ページの表13 および 233ページの表14 は、これらの値の可能な組み合わせと、各組み合わせについて検証が行われる場所を示します。これらの表に示される組み合わせだけが、DB2 コネクトによって DCE ディレクトリー・サービスの使用によりサポートされます。

表 13. APPC 接続を使用した場合の、DCE による有効な機密保護のシナリオ

ケース	サーバーのデータベース・オブジェクト		経路指定オブジェクト	検証
	認証	機密保護	DB2 コネクト・ゲートウェイでの認証 (1= 真、0= 偽)	
1	CLIENT	SAME	0	リモート・クライアント (または DB2 コネクト・ワークステーション)
2	CLIENT	SAME	1	DB2 コネクト・ワークステーション
3	SERVER	PROGRAM	0	ホストまたは AS/400 データベース・サーバー
4	SERVER	PROGRAM	1	DB2 コネクト・ワークステーションとホストまたは AS/400 データベース・サーバー
5	DCE	NONE	なし	DCE 機密保護サーバーで

注: リモート・クライアントが APPC 接続を介して DB2 コネクト エンタープライズ・エディション・ゲートウェイ・ワークステーションに接続されている場合は、ゲートウェイの DCE ロケーター・オブジェクトでの機密保護タイプを NONE に指定します。

表 14. TCP/IP 接続を使用した場合の、DCE による有効な機密保護のシナリオ

ケース	サーバーのデータ ベース・オブジェ クト	経路指定 オブジェクト	検証
	認証	DB2 コネクト エンタープライズ ・エディション・ ゲートウェイ での認証 (1= 真、0= 偽)	
1	CLIENT	0	リモート・クライア ント (または DB2 コネクト・ワークス テーション)
2	CLIENT	1	DB2 コネクト・ワー クステーション
3	SERVER	0	ホストまたは AS/400 データベース・サー バー
4	N/A	N/A	なし
5	DCE	N/A	DCE 機密保護サーバ ーで

次に、各組み合わせについて、より詳細に説明します。

- 1 番目は、ユーザー名とパスワードが、リモート・クライアントでのみ検証される場合です。(ローカル・クライアントの場合、ユーザー名とパスワードが検証されるのは DB2 コネクト・クライアントだけです。)

ユーザーは、最初にサインオンした場所で認証されるよう期待されています。ユーザー ID はネットワークを通して送信されますが、パスワードは送信されません。このタイプの機密保護は、すべてのクライアント・ワークステーションが適切な機密保護機構を持っている場合にのみ使われます。

- 2 番目は、ユーザー名とパスワードが DB2 コネクト・ワークステーションでのみ検証される場合です。パスワードは、ネットワークを介してリモート・クライアントから DB2 コネクト・サーバーへ (ホストまたは AS/400 データベース・サーバーへではない) に送信されます。
- 3 番目は、ユーザー名とパスワードがホストまたは AS/400 データベース・サーバーでのみ検証される場合です。パスワードは、ネットワークを介して

リモート・クライアントから DB2 コネクト・サーバーへ、次いで DB2 コネクト・ワークステーションからホストまたは AS/400 データベース・サーバーへ送信されます。

- 4 番目は、ユーザー名とパスワードが DB2 コネクト・ワークステーションとホストまたは AS/400 データベース・サーバーの両方で検証される場合です。パスワードは、ネットワークを介してリモート・クライアントから DB2 コネクト・サーバーへ、次いで DB2 コネクト・サーバーからホストまたは AS/400 データベース・サーバーへ送信されます。

検証が 2 つの場所で行われるため、同じセットのユーザー名とパスワードが DB2 コネクト・サーバーとホストまたは AS/400 データベース・サーバーの両方で維持されなければなりません。

- 5 番目は、DCE チケットが DCE 機密保護サーバーから得られる場合です。

**注:**

1. AIX ベースのシステムについては、機密保護タイプ SAME を使用するユーザーはすべて、**AIX システム・グループ**に属している必要があります。
2. リモート・クライアントを有する AIX ベースのシステムについては、DB2 コネクト・サーバー上で稼働する DB2 コネクト製品のインスタンスは、**AIX システム・グループ**に属している必要があります。
3. ホストまたは AS/400 データベース・サーバーへのアクセスは、それ自体の機密保護機構またはサブシステムによって制御されています。たとえば、仮想記憶通信アクセス方式 (VTAM) および資源アクセス管理機能 (RACF) がそれです。保護されたデータベース・オブジェクトへのアクセスは、**SQL GRANT** および **REVOKE** ステートメントによって制御されます。

---

## 付録E. バック・レベル・クライアント用ユーティリティのバインド

以前のリリースからのリモート・クライアントがある場合には、これらのクライアント上にあるユーティリティをホストまたは AS/400 データベース・サーバーにバインドする必要があります。

- 旧クライアントが、同じホストまたは AS/400 データベース・サーバーに対して以前のリリースの DB2 コネクトとともに使用されている場合は、追加のステップは必要ありません。
- 旧クライアントが DB2 コネクトとともに使用されていない場合は (たとえば、いくつかの OS/2 マシンがホストまたは AS/400 データベース・サーバーへの接続なしに接続されている場合)、以下のステップを行います。
  1. DB2 (OS/2 版) バージョン 1.0 または 1.2 のクライアントがある場合は、以下の行を使用してバインド・リスト・ファイルを作成します。

```
sqlabind.bnd+  
sqlueiwi.bnd+  
sqluigsi.bnd+  
sqluiici.bnd+  
sqluiict.bnd+  
sqluexpm.bnd+  
sqluimpm.bnd+  
sqlurexp.bnd+  
sqlarxcs.bnd+  
sqlarxrr.bnd+  
sqlarxur.bnd
```

そしてこれらのバインド・ファイルのおのおのをクライアントの 1 つから DB2 コネクト・ワークステーションへコピーします。

2. クライアント・アプリケーション・イネーブラー バージョン 1.0 または 1.2 がある場合は、以下の行を使用してバインド・リスト・ファイルを作成します。

```
db2ajgrt.bnd+  
db2clics.bnd+  
db2clpcs.bnd+
```

```
db2clpr.r.bnd+
db2clpur.bnd+
db2ueiwi.bnd+
db2uigsi.bnd+
db2uiici.bnd+
db2uiict.bnd+
db2uexpm.bnd+
db2uimpm.bnd+
db2urexp.bnd
```

そしてこれらのバインド・ファイルのおのおのをクライアントの 1 つから DB2 コネクト・ワークステーションへコピーします。

3. DB2 コネクト・サーバーで、各バインド・リスト・ファイルを各ホストまたは AS/400 データベース・サーバーのデータベースへバインドします。以下と同様のコマンドを発行します。

```
db2 connect to DBALIAS user USERID using PASSWORD
db2 bind path@bindfile.lst blocking all
      sqlerror continue messages bindfile.msg grant public
db2 connect reset
```

ここで、*DBALIAS*、*USERID*、および *PASSWORD* は、ホストまたは AS/400 データベース・サーバーのデータベースへ適用され、*bindfile* は、バインド・リスト・ファイルの名前であり、そして *path* はバインド・リスト・ファイルのロケーションです。

**bind** コマンドの *grant* オプションを使用することにより、**PUBLIC** に対し、または特定のユーザー名またはグループ ID に対し、**EXECUTE** 特権を付与することができます。**bind** コマンドの *grant* オプションを使用しない場合は、**GRANT EXECUTE (RUN)** を個別に各パッケージに対して付与しなければなりません。

バインド・ファイルについてのパッケージ名を見つけるには、次のコマンドを入力します。

```
ddcspkgn @bindfile.lst
```

---

## 付録F. CLISCHEMA キーワードによる CLI/ODBC アプリケーションのパフォーマンス調整

このセクションには、CLISCHEMA 初期化キーワードを使って ODBC/CLI アプリケーションのパフォーマンスを調整する際に役立つ新しい情報を記載します。この部分では、ネットワークまたはデータベース・パフォーマンスの調整に関する一般情報については記載しません (143ページの『第12章 パフォーマンス』を参照) この付録の情報の要約は、次のとおりです。

- 『ターゲット環境』
- 『CLI/ODBC』
- 238ページの『DB2 の CLISCHEMA 初期化キーワード』
- 241ページの『アプローチに関する提案事項』
- 242ページの『追加のヒント』
- 242ページの『db2ocat カタログ最適化ツール』
- 243ページの『その他の情報源』

---

### ターゲット環境

ここでは主に、DB2 ユニバーサル・データベース (OS/390 版) のユーザーを念頭に置いて情報を提供します。ターゲット環境の構成は以下のとおりです。

- DB2 ユニバーサル・データベース・クライアントとともに実行される CLI/ODBC アプリケーション
- DB2 コネクト バージョン 5 以降 (パーソナル・エディションまたはエンタープライズ・エディション)
- DB2 ユニバーサル・データベース (OS/390 版) バージョン 5.1 以降 (または特に指示がない限り DB2 (MVS/ESA 版))

---

### CLI/ODBC

CLI/ODBC とは、データベース・アプリケーションから呼び出し可能な SQL アプリケーション・プログラミング・インターフェースのことです。このインターフェースで、データベース関数呼び出しのように動的 SQL ステートメントを受け渡します。組み込み SQL とは異なり、ホスト変数やプリコンパイラは必要ありません。

アプリケーション・プログラムが CLI/ODBC を呼び出す際にまずしなければならないことは、ターゲット・データベースのいくつかのシステム・カタログ表に SQL 呼び出しをかけて、他のデータベースの内容に関する情報を取得することです。CLI/ODBC アプリケーションは、常にこの方法でシステム・カタログ表にアクセスします。接続しようとする宛先のデータベースに関する情報を収集する場合に使用できる API 呼び出しは 10 個あります。それらの API 呼び出しは以下のとおりです。

- SQLTables
- SQLColumns
- SQLSpecialcolumns
- SQLStatistics
- SQLPrimarykeys
- SQLForeignkeys
- SQLTablePrivileges
- SQLColumnPrivileges
- SQLProcedures
- SQLProcedureColumns

これらの API 呼び出しと表に関する詳細は、*コール・レベル・インターフェースの手引きおよび解説書* を参照してください。

省略時解釈では、データベースに接続すると、CLI/ODBC アプリケーションは、そのデータベースのすべてのデータベース表に関する情報についてシステム・カタログ表に照会します。特に大規模なシステムでは、この動作のためにネットワーク通信量が増え、アプリケーションの始動もかなり遅くなる可能性があります。

---

## DB2 の CLISHEMA 初期化キーワード

DB2 ユニバーサル・データベースでは、データベースに最初に接続した後の「情報収集」段階で初期 API 呼び出しが戻すデータの量を制限するために使用できるいくつかの CLI/ODBC 初期化キーワードが用意されています。これらのキーワードを設定するには、以下のようになります。

1. db2cli.ini ファイルを手動編集する。
2. クライアント構成支援機能 (サポートされているプラットフォームで) を使用して、データベースの ODBC/CLI 設定を変更する。
3. DBA コマンド行インターフェースを使用して、データベース CLI 構成を更新する。

キーワードは、以下のとおりです。



- DBNAME
- TABLETYPE
- SCHEMALIST
- SYSSCHEMA
- CLISCHEMA

CLISCHEMA の情報は例外ですが、これらのキーワードについては CLI/ODBC ヘルプと コール・レベル・インターフェースの手引きおよび解説書 に文書化されています。この後の部分では、CLISCHEMA の使い方についてのみ説明します。

おって コール・レベル・インターフェースの手引きおよび解説書 に追加される予定になっている CLISCHEMA のドキュメンテーションは、以下のとおりです。

```
db2cli.ini Keyword Syntax: CLISCHEMA = clischema
Default Setting:          No alternatives specified.
DB2 CLI/ODBC Settings Tab: Not present.
```

## 使用上の注意

CLISCHEMA オプションは、カタログ情報を収集するため DB2 CLI 呼び出しと ODBC 呼び出しが発行されたときに、SYSIBM (または SYSTEM、QSYS2) ではなく、代替スキーマ、表、および索引を検索するよう指示します。

たとえば、CLISCHEMA='SERGE' という指定にした場合は、通常ならシステム表を参照する内部の CLI/ODBC API 呼び出しが、以下のユーザー表を代わりに参照します。

- SERGE.TABLES
- SERGE.COLUMNNS
- SERGE.SPECIALCOLUMNNS
- SERGE.TSTATISTICS
- SERGE.PRIMARYKEYS
- SERGE.FOREIGNKEYS
- SERGE.TABLEPRIVILEGES
- SERGE.COLUMNNTABLES
- SERGE.PROCEDURES
- SERGE.PROCEDURESCOLUMNNS

データベース管理者がこれらのユーザー表を作成していないと、CLISCHEMA を使うことはできません。

**注:** DataPropagator は、CLISCHEMA のサポートを提供しているので、データベース管理者は以下の 3 つの方法でこのタスクを実行できるようになっています。

1. クライアントで db2cli.exe を使う。

2. DataPropagator を使用してサーバーで自動実行する。
3. サーバーで手動実行する。

クライアントでこのタスクを実行する方法について、以下に説明します。

## db2cli および bldschem ユーティリティー

CLISCHEMA で必要とされる、ユーザー表を設定するユーティリティーは、CLI コマンド行インターフェースの bldschem サポート・コマンドの形で提供されます。これは事前に文書化されていませんが、/samples/cli/db2cli.exe にあります。bldschem サポート・コマンドを含まない db2cli.exe の文書は、/samples/cli/INTCLI.DOC にあります。

たとえば、スキーマ所有者 (作成者) USERID が表名 STAFF を所有する場合に CLISCHEMA='SERGE' の処理に必要なユーザー表の集合をデータベース SAMPLE に作成するには、db2start を発行してデータベースを ODBC/CLI に登録してから、次のコマンドを実行します。

```
db2cli < addstaff.txt
```

"addstaff.txt" には、以下のスクリプトを入れます。

```
opt callerror on
opt echo on
quickc 1 1 sample userid password
#
# Repeat next line for each table to add.
#
bldschem 1 SERGE USERID STAFF
#
# Exit
#
killenv 1
```

このスクリプトは、表 USERID.STAFF のシステム・カタログ表データを取り込んだ、上記のリストに示されている索引付きの表の集合 SERGE.\* を作成します。たとえば、SERGE.TABLES には一致したそれぞれの項目の新しい行が取り込まれます。bldschem 追加呼び出しを発行すると、既存の行が置き換えられ、既存の SERGE.\* 実行可能プログラムに追加されます。

まとめとして、bldschem サポート・コマンドの構文は以下のようになります。

```
bldschem <handle_number> <value_of_CLISCHEMA> <schema_owner> <table_name>
```

値の意味は以下のとおりです。

- <handle\_number> should be 1
- <value\_of\_CLISCHEMA> should be the same as the schema name specified with the CLISCHEMA keyword

- <schema\_owner> is the creator of the table
- <table\_name> can be the name of a user table, a view, an alias, a synonym, or a system table name.  
(Wildcard characters are allowed).

この後、db2cli.exe で以下の例を実行すると、直前の例で作成したユーザー表 SERGE.\* に、 FRED と BERT がスキーマ所有者である各表のシステム・カタログ表のデータを反映する行が追加されます。

```
bldschem 1 SERGE FRED %  
bldschem 1 SERGE BERT %
```

この後、さらに CLISCHEMA CLI/ODBC キーワードを SERGE に設定して、SAMPLE データベースに対して ODBC/CLI アプリケーションの処理をかけると、システム・カタログ表ではなく SERGE.\* 表集合が参照されます。

---

## アプローチに関する提案事項

ほとんどの実稼働環境では、システム・カタログ表の省略時検索で大量のデータが戻ってくるので、CLI/ODBC アプリケーションがデータベースをオープンするたびにその処理速度がかなり遅くなることがあります。典型的なテスト・データベースでも、25 秒ぐらいかかってしまうことがあります。

最初に、上記の CLI キーワードをまったく設定しない状態で、この遅延時間、特に DB2 クライアントが新しいデータベースに最初に接続するときに発生する可能性のある長い遅延時間を差し引いて計ってください。自動バインド処理は数分かかることもあります。

この後どうするかは、データの構造と貴社の組織により異なります。DBNAME、SCHEMALIST、および TABLETYPE を組み合わせて、特定のアプリケーションまたはアプリケーション・グループの使用を念頭に置いて検索範囲を限定することもできます。たとえば、実動 DBA クライアントが通常所定の DBNAME とスキーマのもとにある表にアクセスする場合は、指定しやすくなります。

CLISCHEMA は、ほとんどのユーザーに最高のパフォーマンス上の利点を提供します。この理由で、実稼働環境では一般に CLISCHEMA の使用をお勧めします。CLI コマンド行インターフェース (db2cli.exe) と bldschem サポート・コマンドを使用して、CLISCHEMA のユーザー表を作成 / 修正する方がずっと簡単です。242ページの『db2ocat カタログ最適化ツール』も参照してください。

---

## 追加のヒント

CLISCHEMA キーワードは、db2cli.ini ファイルの DSN 名のセクションか共通セクションに追加する必要があります。セクションとは、大括弧で囲まれているテキストのことです。COMMON セクションは、大括弧に囲まれている「COMMON」というテキストで分かるようになっています。キーワードとセクション名は大文字小文字を区別しないので注意してください。

接続時に、まず DSN 名に有効なキーワードがあるかどうかをチェックし、なければ COMMON セクションを探します。このようにして、DSN 固有のキーワードとグローバル (クライアント) ・キーワードが両方可能になっています。

また、DBALIAS キーワードを使って、同じデータベースにマップされる異なる DSN (ODBC データ・ソース) を作成することもできます。(DSN 名の長さは 255 文字以内で、8 文字の dbname にマップされます)。

以下の例では、リストに示されていない TESTDB または DSN にユーザーが接続するたびに clischema=ODBCCAT を使うようになっています。TestDBcar2 に接続するときは clischema=odbccat2 を使いますが、testdb データベースには接続します。

以下に、db2cli.ini のサンプル・ファイルを示します。

```
[TESTDB]
[COMMON]
clischema=odbccat
[TestDBcat1]
DBALIAS=testdb
clischema=odbccat1
[TestDBcat2]
DBALIAS=testdb
clischema=odbccat2
```

---

## db2ocat カタログ最適化ツール

Windows 32 ビット・オペレーティング・システムでは、ODBC および JDBC アプリケーションでのシステム・カタログ検索を最適化するための新しいツール db2ocat が提供されています。

ポイント・アンド・クリック方式の db2ocat カタログ最適化ユーティリティーは、以下のサイトから db2ocat.zip をダウンロードすれば入手できます。

<ftp://ftp.software.ibm.com/ps/products/db2/tools>.

---

## その他の情報源

以下の情報源も参考になります。

- DataPropagator を使用したカタログ自動伝搬に関する資料:  
<http://www.ibm.com/software/data/db2/os390/odbcatl1g.html>
- DataPropagator を使わない手動によるアプローチに関する資料:  
<http://www.ibm.com/software/data/db2/os390/odbcmanu.html>



---

## 付録G. 追加情報や関連情報の情報源

---

### 他の関連資料

資料番号	書名
SG24-2006	<i>Migrating to DB2 Universal Database Version 5</i>
SG24-2213	<i>DB2 for OS/390 Version 5 Performance Topics</i>
SG24-4893	<i>DB2 Meets NT</i>
SG24-4894	<i>The Universal Connectivity Guide to DB2</i>
SG24-4693	<i>Getting Started with DB2 Stored Procedures</i>
SG24-2212	<i>DRDA Support for TCP/IP in DB2 Universal Database for OS/390 V5.1 and DB2 Universal Database V5.0</i>
SC88-6725	<i>AIX CICS/6000 アプリケーション・プログラミング・ガイド</i>
SC88-6734	<i>AIX CICS/6000 カスタマイゼーションおよび操作</i>
GC88-8519	<i>DB2 コネクト エンタープライズ・エディション (UNIX 版) 概説およびインストール</i>
GC88-8520	<i>DB2 コネクト エンタープライズ・エディション (OS/2 および Windows 版) 概説およびインストール</i>
GC88-8533	<i>DB2 コネクト パーソナル・エディション 概説およびインストール</i>
GG24-4155	<i>Distributed Relational Database Architecture: Using DDCS for AIX DRDA support with DB2 for MVS/ESA and DB2 Universal Database for AS/400</i>
GG24-4311	<i>Distributed Relational Database Architecture Cross Platform Connectivity and Application</i>
SC23-2443	<i>Encina for AIX Product Family Overview</i>





---

## 付録H. 特記事項

本書において、日本では発表されていない IBM 製品 (機械およびプログラム)、プログラミングまたはサービスについて言及または説明する場合があります。しかし、このことは、弊社がこのような IBM 製品、プログラミングまたはサービスを、日本で発表する意図があることを必ずしも示すものではありません。本書で、IBM ライセンス・プログラムまたは他の IBM 製品に言及している部分があっても、このことは当該プログラムまたは製品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。これらのプログラムまたは製品に代えて、IBM の知的所有権を侵害することのない機能的に同等な他社のプログラム、製品またはサービスを使用することができます。ただし、IBM によって明示的に指定されたものを除き、これらのプログラムまたは製品に関連する稼働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 および他社は、本書で説明する主題に関する特許権 (特許出願を含む)、商標権、または著作権を所有している場合があります。本書は、これらの特許権、商標権、および著作権について、本書で明示されている場合を除き、実施権、使用権等を許諾することを意味するものではありません。実施権、使用権等の許諾については、下記の宛先に、書面にてご照会ください。

〒106-0032 東京都港区六本木 3 丁目 2-31  
AP 事業所  
IBM World Trade Asia Corporation  
Intellectual Property Law & 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。

本書に含まれる情報には、技術的に不正確なもの、または誤植が含まれる場合があります。これらに対する変更は、定期的に行われます。これらの変更は、資料の改訂版に含まれます。IBM は、本書で説明している製品、プログラムに対して、予告なく改良、変更を加える場合があります。

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、お客様になんら義務も負わせない適切な方法で、使用もしくは配布することがあります。

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 Canada Limited  
Office of the Lab Director  
1150 Eglinton Ave. East  
North York, Ontario  
M3C 1H7  
CANADA

本プログラムに関する上記の情報は、適切な条件の下で使用することができませんが、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。

本書に含まれるパフォーマンス・データは、制御された環境下で決定されています。したがって、その他の稼働環境で得られる結果とは、かなり異なる可能性もあります。一部の測定値は、開発中のシステムを使用している場合があり、これらの測定値が一般的に提供可能なシステムで同様の数値になることを保証するものではありません。さらに、一部の測定値が推定されたものもあります。実測値と異なる場合があります。本書のユーザーは、使用される特定の環境での該当データを確認してください。

IBM 以外の製品については、当該製品の提供者から直接、出版されている資料または一般公開されている情報から入手しました。IBM は、これらの製品についてはテストを行っておらず、これらの IBM 以外の製品に関する性能、互換性またはその他の主張について確認することはできません。IBM 以外の製品の機能に対する質問は、それぞれの製品提供者にお問い合わせください。

IBM の将来の方向性または意図については、予告なしに変更または中止する場合があります。IBM の目的および目標のみを示しているものです。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれていますが、これは説明に具体性を与えるために記載されたものであり、それらの例には、個人、企業、ブランドの、あるいは製品などの名前が含まれている場合があります。それらの名前はすべて架空のものであり、また名称や住所が類似する企業が実在しても、それは偶然に過ぎません。

#### 著作権：

本書に含まれる情報には、サンプル・アプリケーション・プログラムがソース言語の形式で含まれており、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技法を示しています。お客様は、これらのサンプル・プログラムが書かれているオペレーティング・プラットフォームでアプリケーション・プログラミング・インターフェースが実行可能となるためのアプリケーション・プログラムを開発、使用、販売または配布もしくは転送する目的のためだけにのみ、サンプル・プログラムを、IBM に対する別途料金を支払うことなく、複製、変更、配布または転送することができます。これらのサンプルは、すべての条件下で十分にテストを行っていません。したがって、IBM は、これらのプログラムの信頼性、実用性または機能について、いかなる保証も負いません。

サンプル・プログラムまたはその改変版の複製物には、全部複製か部分複製かを問わず、次の著作権表示を必ず行うものとします。

© (お客様の会社名) (西暦年). このコードの一部は IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. \_年\_. All rights reserved.

---

## 商標

次のものは、IBM Corporation の米国およびその他の国における商標です。

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2
BookManager	OS/390
CICS	OS/400
C Set++	PowerPC
C/370	QBIC
DATABASE 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000
DataRefresher	S/370
DB2	SP
DB2 Connect	SQL/DS
DB2 Extenders	SQL/400
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational Database Architecture	SystemView
DRDA	VisualAge
eNetwork	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WebExplorer
	WIN-OS/2

次のものは、他社の商標または登録商標です。

Tivoli および NetView は、米国およびその他の国における Tivoli Systems Inc. の商標です。

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 がライセンスしている米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標または登録商標です。



# 索引

日本語、数字、英字、特殊文字の順に配列されています。なお、濁音と半濁音は清音と同等に扱われています。

## 【ア行】

アウトバウンド順序番号 (モニター) 116  
アクセス RDB コマンド 193  
宛先データベース名 87, 217  
アプリケーション  
    バインド 101  
アプリケーション開発 47, 148  
    DB2 アプリケーション開発クライアントの使用 27  
    ODBC の使用 27, 70  
アプリケーション名 (モニター) 113  
アプリケーション・サーバー  
    概要 36  
    構成 39  
    配置 39  
    DB2 コネクト・サポート 38  
    DRDA 定義 10  
アプリケーション・パフォーマンス  
    CLISHEMA キーワード 237  
アプリケーション・ハンドル 113  
アプリケーション・リクエスター  
    パラメーター 217  
    DRDA 定義 10  
アプリケーション・リクエスター名 88  
アンパーサンド、二重 (&&)、SQLCODE マッピング・ファイル内の 138  
暗黙接続 51  
移植、アプリケーションの 47  
移植に関する資料 67  
イベント  
    トレース 188

インストール  
    DB2 コネクト 6  
インポート  
    制限 123  
インポート・ユーティリティー  
    一般説明 122  
ウィザード  
    複数サイト更新 20  
エクスポート  
    制限 123  
エクスポート・ユーティリティー  
    一般説明 122  
エスケープ文字 95  
エラー 181  
応答時間 143

## 【カ行】

カーソル  
    確定 52  
    動的 52  
    未確定 52  
カーソル固定 55  
外部キー 54  
会計ストリング 64  
会計ストリング・フィールド 64  
拡張、ホストまたは AS/400 サーバーでのデータの 49  
拡張された動的 SQL ステートメント  
    サポートされない 64  
確定カーソル 52  
仮想記憶通信アクセス方式 (VTAM) 129, 234  
各国語サポート (NLS)  
    考慮事項 219  
    混合バイト・データ 49, 124  
    文字データの変換 219  
可変長ストリング 49  
可変長文字列 124  
環境変数  
    DB2ACCOUNT 65  
管理ユーティリティー 7  
キー  
    外部 54  
    基本 54  
キーワード  
    CLISHEMA 238, 240, 241, 242  
    DBALIAS 242  
記号宛先名 217  
    大文字小文字の区別 86  
機能強化  
    DB2 コネクト バージョン 5.0 212  
    DB2 コネクト バージョン 5.2 212  
    DB2 コネクト バージョン 6.1 211  
    DDCS バージョン 2.3 215  
    DDCS バージョン 2.4 214  
基本キー 54  
基本障害保守ログ 185  
機密保護 231  
    考慮事項 125  
    タイプ 86, 128, 217, 231  
APPC 128  
DCE 125  
GRANT 234  
GRANT ステートメント 130  
NONE 128, 129  
PROGRAM 128  
REVOKE 234  
REVOKE ステートメント 130  
SAME 128  
キャッシュ・ディレクトリー情報 154  
許可 ID (モニター) 113  
競合、システム・リソースの 170  
共通 SQL 7  
行レベルのロック 55  
国別コード  
    SQLCA の SQLERRMC フィールドにおける 51

- 国別コード・ページ・サポート 219
- 組み込み SQL 20
- クライアント DB 別名 (モニター) 115
- クライアント NNAME (モニター) 115
- クライアント順序番号 (モニター) 115
- クライアント製品 ID (モニター) 115
- クライアント・アプリケーション ID (モニター) 114
- グループ化要求
  - データベース 150
- 経路指定情報オブジェクト 223
- 経路指定要求 11
- コード化文字セット識別子 (CCSID) 219
- コード・セット
  - SQLCA の SQLERRMC フィールドにおける 51
- コード・ページ 219
  - 変換例外 222
  - SQLCA の SQLERRMC フィールドにおける 51
- コード・ページ ID (モニター) 116
- コア・ファイル 186
- 交換サーバー属性コマンド 193
- 構成
  - DB2 コネクト 6
  - ODBC ドライバー 74, 76
- 構成についての考慮事項
  - パスワード変更 135
- 構文
  - bldschm 240
- コマンド
  - コミット 195
  - ACCRDB 193, 194
  - ACCRDBRM 193, 194
  - BIND 107
  - EXCSAT 193
  - EXCSATRD 193, 194
  - EXCSQLSTT 63
  - FORCE 51
  - quit 122
  - REBIND PACKAGE 107
  - コマンド (続き)
    - terminate 122
  - コマンド行プロセッサ (CLP) 7, 121
    - パフォーマンス 152
    - REBIND PACKAGE コマンド 107
  - コミット・コマンド 195
  - 混合バイト・データ 49
    - インポート 124
    - エクスポート 124
  - コンマ、パラメーター・ストリング内の 89
  - コンマ・コンマ、パラメーター・ストリング内の 89
- [サ行]**
- 作業単位
  - 分散 14
  - リモート 11
- 作成者属性
  - パッケージ 52
- 参照保全 54
- 時間帯サポート 91
- 資源アクセス管理機能 (RACF) 129, 234
- 自己参照表 54
- システム資源
  - 競合 170
- システム・カタログ
  - 使用 56
- システム・データベース・ディレクトリー 85, 96
- 実行、アプリケーションの
  - データベース・クライアント 69
- 視点
  - システム・カタログ 56
- シフトアウト (SO) 文字およびシフトイン (SI) 文字 49, 124
- 集合 53
- 集合 ID 属性
  - パッケージ 52
  - DB2 ユニバーサル・データベース (AS/400 版) 53
- 修飾子属性
  - 異種プラットフォーム 52
- 修飾子属性 (続き)
  - パッケージ 52
- 終了作業単位応答メッセージ (ENDUOWRM) 195
- 受信バッファ (トレース) 187
- 述部論理 150
- 種類、SQL の 7
- 使用、DB2 コネクトの
  - シナリオ 23
  - トランザクション処理モニターとの 40
  - Tuxedo 43
  - XA 準拠トランザクション・マネージャ 43
- 照合順序
  - EBCDIC と ASCII 54
- 所有者属性
  - パッケージ 52
- 診断ツール 185
- 数値データ・タイプ 49
- 数値変換オーバーフロー 56
- ステートメント
  - 接続 51
  - 呼び出し 57
  - ACQUIRE 63
  - COMMIT 152
  - COMMIT WORK RELEASE 63
  - CREATE STORGROUP 49
  - CREATE TABLESPACE 49
  - DECLARE 63
  - DELETE 49
  - DESCRIBE 63, 152
  - EXECUTE IMMEDIATE 152
  - FOR FETCH ONLY 151
  - GRANT 50
  - INSERT 49, 50
  - LABEL ON 63
  - PREPARE 63, 152
  - ROLLBACK 51, 152
  - SELECT 49, 151, 152
  - SET CURRENT 63
  - UPDATE 49
- ストアード・プロシージャ
  - 一般説明 57
  - 概要 31



ストアド・プロシージャ・ビル  
ダー  
概要 59  
機能 59  
スループット  
トランザクション 143  
制限  
インポートとエクスポート 123  
接続コンセントレーター 161  
整数データ・タイプ 165  
静的 SQL 7, 152  
DB2 コネクト・サポート 47  
セクション番号 63  
設計、アプリケーションの 148  
接続  
暗黙接続 51  
CONNECT RESET ステートメン  
ト 51  
CONNECT TO ステートメント  
51  
null CONNECT 51  
接続、DRDA ホストへの  
DRDA ホストへの直接接続 25  
接続コンセントレーター  
概要 157  
構成パラメーター 159  
制限 161  
接続オーバーヘッド 158  
例 160  
XA トランザクション・サポート  
159  
接続サーバー  
DB2 コネクト エンタープライ  
ズ・エディション 27  
接続プール  
概要 31  
セットアップ  
DB2 コネクト 6  
前提条件  
DCE 125  
ゾーン 10 進数データ・タイプ 165  
相違点、異なる DB2 製品間の 48  
送信バッファ (トレース) 187

## [タ行]

対話式入力モード (CLP) 122

タイプ  
機密保護 128  
認証 126  
ROWID 50  
タイムアウト、ロックの 55  
チャージバック会計  
定義 64  
DB2 ユニバーサル・データベー  
ス (OS/390 版) 64  
チューニング  
アプリケーション・パフォーマンス  
237  
データベース 163  
ネットワーク・パフォーマンス  
166  
長フィールド 50  
直接接続  
ホスト・データベースへの 24  
直接的なデータベース・アクセス  
24  
ツール  
パフォーマンス 146  
メモリー使用状況 146  
CPU 使用状況 146  
データ制御言語 (DCL) 50  
データ操作言語 (DML) 49  
データ定義言語 (DDL) 49  
データ転送  
ホストおよびワークステーション  
間の 122  
データ転送速度 143  
パフォーマンス 169  
データベース  
オブジェクト 223  
グループ化要求 150  
チューニング 163  
名前 217  
パフォーマンス・ツール 146  
別名 218  
ロケーター・オブジェクト 223  
データベース接続サービス (DCS) デ  
ィレクトリー 85  
データベース別名 96  
データベース名 87, 96  
データベース・システム・モニター  
109

データベース・システム・モニタ  
ー・ユーティリティー 8  
データベース・ディレクトリー  
更新 85  
システム・データベース 85  
データベース接続サービス  
(DCS) 85  
ノード 85  
データベース・ディレクトリーの更  
新 85  
データ変換 164  
コード・ページ 219  
文字置換 222  
例外 222  
2 バイト文字 222  
CCSID 219  
データ・ソース 12  
データ・タイプ  
数値 49  
整数 165  
ゾーン 10 進数 165  
パック 10 進数 165  
浮動小数点 165  
変換 164  
CHAR 166  
VARCHAR 166  
データ・フロー 10, 143  
データ・ブロック 150  
定様式データ・オブジェクト内容体  
系 (FD:OCA) 10  
テリトリー  
SQLCA の SQLERRMC フィール  
ドにおける 51  
トークンおよび SQLCODE 137  
動的 SQL 7, 152  
CURRENTPACKAGESET 135  
DB2 コネクト・サポート 47  
動的カーソル 52  
登録  
ODBC ドライバー・マネージャ  
73  
トランザクション  
スループット 143  
トランザクション処理  
特性 41

トランザクション処理モニター  
例 42  
トレース・ユーティリティ 186  
構文 187  
出力 189  
出力ファイル 186

## [ナ行]

認証 96, 218  
検証 125  
認証タイプ  
省略時 126  
CLIENT 126  
DCE 126  
DCE ディレクトリー・サービス 231  
DCS 126  
DCS\_ENCRYPT 126  
SERVER 126  
SERVER\_ENCRYPT 126  
ネットワーク  
アダプターまたは通信制御装置 170  
信頼性 170  
チューニング 166  
通信量 170  
トポロジー 170  
ネットワーク・パフォーマンス・ツール 147  
ノード名 86, 96, 217, 218  
ノード・ディレクトリー 85, 86

## [ハ行]

ハードウェア  
ネットワーク・パフォーマンス 169  
バインド  
パッケージ 105  
必要な権限 102  
ユーティリティ 70  
ユーティリティおよびアプリケーション 101  
バインドに必要な権限 102  
バインド・リスト 101, 236

パスワード  
DCE ディレクトリー・サービス 231  
パスワード有効期限管理 (PEM) 92  
バック 10 進数データ・タイプ 165  
パッケージ  
属性 52  
ホストまたは AS/400 データベース・サーバーで作成される 105  
パッケージ属性  
作成者 52  
修飾子 52  
所有者 52  
パフォーマンス  
一般説明 143  
コマンド行プロセッサ 152  
チューニング 163  
ツール 146  
トラブルシューティング 171  
ネットワーク・ツール 147  
ネットワーク・ハードウェア 169  
ベンチマーク 145  
ボトルネック 145  
CLI アプリケーション 237  
CLISCHEMA キーワード 237  
DB2 (OS/390 版) 163  
ODBC および JDBC アプリケーションのパフォーマンスのチューニング 237  
PIU サイズ 176  
RU サイズ 176  
SNA チューニング・ヒント 172  
SNA の調整基準 175  
パラメーター  
AGENTPRI 155  
AUTHENTICATION 126  
BIDI 93  
DB\_Authentication 225  
DB\_Communication\_Protocol 225, 227  
DB\_Database\_Protocol 225  
DB\_Native\_Database\_Name 224  
DB\_Object\_Type 224, 227  
DB\_Principal 225  
DB\_Product\_Name 224

パラメーター (続き)  
DB\_Target\_Database\_Info 229  
DFT\_ACCOUNT\_STR 65  
DIR\_CACHE 154  
INTERRUPT\_ENABLED (切断) 90  
LOCALDATE 91  
MAXAGENTS 155, 159  
MAXDARI 154  
MAX\_COORDAGENTS 159  
NOMAP 89  
NUMDB 154  
NUM\_INITAGENTS 159  
NUM\_POOLAGENTS 159  
PRDDTA 64  
PRDID 194  
RQRIOBLK 153  
SYSPLEX 91  
パラメーター・ストリング 217  
日付および時間帯サポート 91  
複合 SQL  
NOT ATOMIC 61, 149  
複数サイト更新 14  
コントロール・センター 20  
サポート 61  
テスト 21  
複数サイト更新ウィザード 20  
浮動小数点データ・タイプ 165  
プリコンパイラー  
サポート 49  
DB2 コネクト・サポート 52  
プログラミング情報 47  
プログラムに関する考慮事項 47  
ホストまたは AS/400 環境での 47  
プロセス状況ユーティリティ 186, 193  
ブロック化 52  
データ 150  
ブロック・サイズ 153  
分離レベル 56, 57  
分散環境 47  
分散データ管理 187  
分散データ管理 (DDM) 10  
分散要求 12

分散リレーショナル・データベース  
体系 (DRDA)  
アプリケーション・サーバー 10  
アプリケーション・リクエスター  
10  
概念 9  
使用される体系 10  
資料 22  
データ・フロー 10  
分類順序  
照合順序 54  
定義 54  
ページング  
ブロック・サイズ 153  
ページ・レベルのロック 55  
変換  
文字 49  
変換、データの 164  
ベンチマーク  
パフォーマンス 145  
ホスト製品 ID (モニター) 116  
ホストまたは AS/400 サーバーとワ  
ークステーションとの違い 63  
ホスト・アプリケーション ID (モニ  
ター) 114  
ホスト・データベース名 (モニタ  
ー) 116  
ボトルネック  
トランザクション 145

## [マ行]

マッピング、SQLCODE 137  
未確定カーソル 52  
メモリー使用状況ツール 146  
文字データ表現体系 (CDRA) 10  
文字変換 49  
モニター  
DB2 コネクト・ゲートウェイで  
の接続 109  
問題解決 181  
数値変換オーバーフロー 56  
問題判別 181

## [ヤ行]

ユーザー定義タイプ  
DB2 コネクトでサポートされて  
いる 50  
ユーザー定義の照合順序 54  
ユーザー名 231  
ユーティリティ  
インポート 122  
エクスポート 122  
管理 7, 121  
データベース・システム・モニタ  
ー 8  
トレース 186  
バインド 70, 101  
プロセス状況 193  
bldschm 240  
db2cli 240  
db2ocat 242  
ddcspkgn 106, 108  
ddcstrc 186  
ps 193

## [ラ行]

リモート作業単位 11  
両方向 CCSID サポート 93  
両方向言語サポート 222  
例  
接続コンセントレーター 160  
XA コンセントレーター 160  
連鎖 55  
ロック  
行レベル 55  
タイムアウト 55  
ページ・レベル 55

## [数字]

2 フェーズ・コミット 14  
TCP/IP 接続によって使用される  
resynch ポート 87  
64 ビット整数 (BIGINT) データ・タ  
イプ  
DB2 コネクト バージョン 7 で  
サポートされている 50

## A

ACCRDB コマンド 193, 194  
ACCRDBRM コマンド 193, 194  
ACCSEC 194  
ACQUIRE ステートメント 63  
AGENTPRI パラメーター 155  
API  
データベース・ディレクトリーの  
更新 97  
APPC  
記号宛先名 217  
AR 名 88  
ARI (DB2 (VSE および VM  
版)) 51  
AS 宛先データベース名 87  
ASCII  
混合バイト・データ 49  
分類順序 54  
AS/400  
DRDA 9  
ATOMIC 複合 SQL  
サポートされない 61  
DB2 コネクトでサポートされな  
い 149  
AUTHENTICATION パラメーター  
126  
AUTHENTICATION=CLIENT 134

## B

BIDI パラメーター 93  
BIND コマンド  
構文 107  
BINDADD 特権 102  
bldschm 240  
構文 240

## C

CALL USING DESCRIPTOR ステ  
ートメント (OS/400) 58  
CALL ステートメント  
異種プラットフォーム 57  
CCSID 222  
CGI プログラミング  
限界 29

CGI プログラミング (続き)  
 利点 29

CHAR データ・タイプ 166

CHGPWD\_SDN パラメーター 92

CICS 48

CLI  
 ユーティリティ 240

CLIENT 認証タイプ 126

CLISHEMA キーワード 238, 240, 241, 242

CLI/ODBC アプリケーション  
 CURRENTPACKAGESET 135

CLI/ODBC アプリケーション・パフォーマンス  
 アプリケーション・パフォーマンス 237

COMMIT WORK RELEASE ステートメント  
 サポートされない 63

COMMIT ステートメント  
 静的にバインドされた 152

CPU 使用状況ツール 146

CREATE IN COLLECTION  
 NULLID 102

CREATE STORGROUP ステートメント  
 サポート 49

CREATE TABLESPACE ステートメント  
 サポート 49

CURRENTPACKAGESET 135

## D

D (切断) パラメーター 89

DataPropagator 240

DB2 (MVS/ESA 版) または DB2 ユニバーサル・データベース (OS/390 版)  
 ブートストラップ・データ・セット 87

DOMAIN 87

RESPORT 87

TCPPOPT 87

DB2 (MVS/ESA 版) または DB2 ユニバーサル・データベース (OS/390 版) 上のブートストラップ・データ・セット  
 BSDS パラメーター 87

DB2 コネクト エンタープライズ・エディション  
 接続サーバーとしての 27

DB2 コネクト バージョン 5.0  
 機能強化 212

DB2 コネクト バージョン 5.2  
 機能強化 212

DB2 コネクト バージョン 6.1  
 機能強化 211

DB2 コネクト RQRIOLBK サイズ 176

DB2 ユニバーサル・データベース (OS/390 版)  
 検査済みの TCP/IP 134

DYNAMICRULES(BIND) 135

DB2ACCOUNT 環境変数 65

db2cli.exe ユーティリティ 240

db2cli.ini ファイル  
 ODBC および JDBC アプリケーションのパフォーマンスのチューニング 238

DB2CONNECT\_IN\_APP\_PROCESS 環境変数 109

db2ocat  
 カタログ最適化ユーティリティ 242

DBALIAS キーワード 242

DB\_Authentication パラメーター 225

DB\_Communication\_Protocol パラメーター 225, 227

DB\_Database\_Protocol パラメーター 225

DB\_Native\_Database\_Name パラメーター 224

DB\_Object\_Type パラメーター 224, 227

DB\_Principal パラメーター 225

DB\_Product\_Name パラメーター 224

DB\_Target\_Database\_Info パラメーター 229

DCE  
 機密保護 125

セットアップ情報 125

ソフトウェア前提条件 125

ディレクトリー・サービス 223

認証タイプ 126

DCL (データ制御言語) 50

DCS  
 認証タイプ 126

DCS ディレクトリー 87  
 宛先データベース名 87  
 アプリケーション・リクエスト名 88  
 データベース名 87  
 パラメーター・ストリングの指定 95

AR 名 88

AS 宛先データベース名 87

BIDI パラメーター 93

CHGPWD\_SDN パラメーター 92

LOCALDATE パラメーター 91

SYSPLEX パラメーター 91

dcs1ari.map ファイル 137

dcs1dsn.map ファイル 137

dcs1qsq.map ファイル 138

DCS\_ENCRYPT 認証タイプ 126

DDCS バージョン 2.3  
 機能強化 215

DDCS バージョン 2.4  
 機能強化 214

ddcs400.lst ファイル 101

ddcsmvslst ファイル 101

ddcstrc ユーティリティ 186  
 構文 187  
 出力 189  
 出力ファイル 186

ddcstrc.dmp ファイル 189

ddcsvm.lst ファイル 101

ddcsvslst ファイル 101

DDL (データ定義言語) 49

DECLARE PROCEDURE ステートメント (OS/400) 58

DECLARE STATEMENT  
 サポート 63

DECLARE ステートメント 63

DELETE  
サポート 49  
DESCRIBE ステートメント 63, 152  
サポート 63  
DFT\_ACCOUNT\_STR 構成パラメーター 65  
DIR\_CACHE パラメーター 154  
DML (データ操作言語) 49  
DSN (DB2 ユニバーサル・データベース (OS/390 版)) 51  
DSS タイプ (トレース) 187

## E

EBCDIC  
混合バイト・データ 49  
分類順序 54  
ENDUOWRM メッセージ 195  
EXCSAT コマンド 193  
EXCSATRD コマンド 193, 194  
EXCSQLSTT コマンド 63  
EXECUTE IMMEDIATE ステートメント 152  
EXTNAM オブジェクト 193

## F

FOR FETCH ONLY, SELECT ステートメントの 151  
FORCE コマンド 51, 113

## G

GRANT ステートメント  
機密保護 130, 234  
GROUP BY 文節  
分類順序 54

## I

IBM SQL 7  
IBM WebSphere  
概要 34  
INSERT ステートメント  
サポート 49, 50

INTERRUPT\_ENABLED (切断) パラメーター 90  
ISO/ANS SQL92 54

## J

Java  
アプリケーション・サーバー  
DB2 コネクトの使用 32  
プログラムの実行 77  
JDBC  
アプリケーション・パフォーマンス 237  
カタログ最適化ユーティリティ 242  
プログラムの実行 77

## L

LABEL ON ステートメント 63  
LANGLEVEL SQL92E プリコンパイル・オプション 54  
LIST DCS APPLICATIONS コマンド 113  
LOB データ・タイプ  
DB2 コネクト バージョン 7 でサポートされている 50  
LOCALDATE パラメーター 91

## M

MAXAGENTS パラメーター 155, 159  
MAXDARI パラメーター 154  
MAX\_COORDAGENTS パラメーター 159  
Microsoft ODBC Driver Manager 72  
Microsoft Windows アプリケーション 27  
MVS  
DRDA 9

## N

Net.Data  
概要 33

Net.Data (続き)  
機能 34  
NOMAP パラメーター 89, 137  
NONE 機密保護タイプ 128, 129  
NOT ATOMIC 複合 SQL 61, 149  
NULLID, OS/400 用 102  
NUMDB パラメーター 154  
NUM\_INITAGENTS パラメーター 159  
NUM\_POOLAGENTS パラメーター 159

## O

ODBC  
アプリケーション・パフォーマンス 237  
インターフェース 27  
ドライバ・マネージャーの登録 73  
プログラムの実行 70  
ODBC アプリケーション  
CURRENTPACKAGESET 135  
odbcad32.exe 72  
ODBC/CLI  
カタログ最適化ユーティリティ 242  
ORDER BY 文節  
分類順序 54  
OS/390  
DRDA 9  
OS/400  
DRDA 9

## P

PC/IXF ファイル形式 123  
PIU 176  
PRDDTA パラメーター (DRDA) 64  
PRDID パラメーター 194  
PREPARE ステートメント 152  
サポート 63  
PROGRAM 機密保護タイプ 128  
ps (プロセス状況) ユーティリティ 186, 193  
PUT ステートメント  
サポートされない 63

- Q**
- QSQ (DB2 ユニバーサル・データベース (AS/400 版) ) 51
- quit コマンド (CLP) 122
- R**
- RACF 234
- RDBNAM オブジェクト 193
- REBIND PACKAGE コマンド (CLP) 107
- REVOKE ステートメント  
機密保護 130, 234  
ステートメント 50
- ROLLBACK WORK RELEASE  
サポートされない 63
- ROLLBACK コマンド  
静的にバインドされた 152
- ROLLBACK ステートメント 51
- ROWID データ・タイプ  
DB2 コネクト バージョン 7 で  
サポートされている 50
- RQRIOBLK サイズ 176
- RQRIOBLK パラメーター 153
- RQRIOBLK フィールド 52
- RU サイズ 176
- S**
- SAME 機密保護タイプ 128
- SECCHK 194
- SELECT ステートメント 151, 152  
サポート 49
- SERVER 認証タイプ 126
- SERVER\_ENCRYPT 認証タイプ 126
- SET CURRENT PACKAGESET 135
- SET CURRENT ステートメント  
サポート 63
- SHOW DETAIL モニター・オプション 114
- SNA 管理サービス体系 (MSA) 10
- SNA パフォーマンス  
チューニング・ヒント 172
- SOCKS  
必須環境変数 86
- SQL
- 静的 152
- 動的 152
- SQL ステートメント  
組み込み 20  
サポート 63  
分類 47
- SQL1338 戻りコード 86
- SQL92 54
- SQLCA  
データのバッファ 187  
SQLCODE フィールド 187  
SQLERRMC フィールド 51, 61  
SQLERRP フィールド 51
- SQLCODE  
異種プラットフォーム 55  
スタンドアロン 54  
マッピング 137  
SQLCA 内のフィールド 187
- SQLCODE および SQLSTATE に関する相違点 55
- SQLCODE マッピング・ファイル  
アスタリスク 138  
構文 139  
cc 139  
i 140  
P 139  
s 140  
U 139  
W 138  
\* (アスタリスク) 138
- SQLDA、最適割り振りサイズ 152
- SQLERRMC フィールド、SQLCA の 51, 61
- SQLERRP フィールド、SQLCA の 51
- sqlsact API 65
- SQLSTATE  
クラス・コード 139  
スタンドアロン 54  
相違 55  
SQLCA の SQLERRMC フィールドにおける 61
- SQL/DS  
DRDA 9
- SRVNAM オブジェクト 193
- SYSIBM.SYSPROCEDURES カタログ (OS/390) 58
- SYSPLEX パラメーター 91
- T**
- TCP/IP  
検査済みの機密保護 134  
サービス名 87  
サービス名またはポート番号 217  
ホスト接続の構成 25  
ホスト名 217  
リモート・ホスト名 87, 217  
ACCSEC 194  
DOMAIN 87  
RESPORT 87  
resynch ポート 87  
SECCHK 194  
TCPPOPT 87
- terminate コマンド (CLP) 122
- Tuxedo  
DB2 コネクトとの 43
- U**
- UPDATE ステートメント  
サポート 49
- V**
- VALIDATE RUN  
DB2 コネクト・サポート 52
- VALNSPRM 値 194
- VARCHAR データ・タイプ 166
- VM  
DRDA 9
- VSE  
DRDA 9
- VTAM 234
- W**
- Web アプリケーション  
概要 29
- WebSphere  
アドバンスド版 36

WebSphere (続き)

エンタープライズ版 36

概要 34

機能 34

スタンダード版 35

Windows アプリケーション 27

## X

XA インターフェース

定義 43

XA コンセントレーター

例 160

XA 準拠トランザクション・マネー

ジャー

定義 43

XA 準拠リソース・マネージャー

43

XA トランザクション・サポート

接続コンセントレーター 159

X/Open 分散トランザクション処理

(DTP) モデル

概要 43

## [特殊文字]

&&、SQLCODE マッピング・ファイル内の 138

\* (アスタリスク)、AIX 版での CLP での 122

, (コンマ)、パラメーター・ストリング内の 89

,, (コンマ・コンマ)、パラメーター・ストリング内の 89

" (二重引用符)、AIX 版での CLP での 122

¥ (円記号)、OS/2 で 188





---

## IBM と連絡をとる

技術上の問題がある場合は、時間をとって「問題判別の手引き」に定義されている処置を検討し、それらの提案を実行した後で、DB2 顧客サービスに連絡をとってください。この資料には、DB2 顧客サービスがお客さまを支援するために必要とする情報が説明されています。

---

### 製品情報

以下の情報は英語で提供されます。内容は英語版製品に関する情報です。

#### **<http://www.ibm.com/software/data/>**

DB2 World Wide Web ページには、ニュース、製品説明、研修スケジュールなどの DB2 に関する最新情報が提供されています。ただし、提供されている情報は英語です。

#### **<http://www.ibm.com/software/data/db2/library/>**

「DB2 Product and Service Technical Library」では、よくされる質問 (FAQ)、修正内容、資料、および最新の DB2 技術情報などの情報へのアクセスが提供されています。

**注:** この情報のご提供は英語のみとなりますのでご注意ください。

#### **<http://www.elink.ibm.com/pbl/pbl/>**

「International Publications」注文用 Web サイトでは、マニュアルの注文方法についての情報を提供しています。ただし、提供されている情報は英語です。

#### **<http://www.ibm.com/education/certify/>**

IBM の「Professional Certification Program」Web サイトでは、DB2 を含むさまざまな IBM 製品の認証テストの情報を提供しています。ただし、提供されている情報は英語です。

#### **<ftp.software.ibm.com>**

匿名でログオンしてください。ディレクトリー /ps/products/db2 には、DB2 および多数の他製品に関連したデモ、修正プログラム、情報、およびツールがあります。ただし、提供されている情報は英語です。

**comp.databases.ibm-db2, bit.listserv.db2-l**

これらのインターネット・ニュースグループは、ユーザーが DB2 製品に関する自分の経験について話し合うために利用できます。ただし、提供されている情報は英語です。

**CompuServe: GO IBMDB2**

このコマンドを入力すると、IBM DB2 Family forum にアクセスできます。すべての DB2 製品が、このフォーラムでサポートされています。ただし、提供されている情報は英語です。

米国以外の国で IBM に連絡する方法については、*IBM Software Support Handbook* の Appendix A を参照してください。この資料にアクセスするには、Web ページ: <http://www.ibm.com/support/> にアクセスし、ページの最下部にある「IBM Software Support Handbook」リンク・ボタンを選択します。

**注:** 国によっては、IBM が承認している販売業者が、IBM サポート・センターの代わりにそれら販売業者のサポート・センターに連絡する場合があります。





Printed in Japan

SC88-8521-00



日本アイ・ビー・エム株式会社

〒106-8711 東京都港区六本木3-2-12