

DB2[®] ユニバーサル・データベース



コマンド解説書

バージョン 7

DB2[®] ユニバーサル・データベース



コマンド解説書

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、655ページの『付録D. 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：	SC09-2951-00 IBM® DB2® Universal Database Command Reference 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
本書の対象読者	vii
本書の構成	vii

第1章 システム・コマンド	1
コマンドの説明の編成方法	2
db2admin - DB2 管理サーバー	3
db2adutl - TSM アーカイブ・イメージによる作業	5
db2advis - DB2 索引アドバイザー	10
db2audit - 監査機能管理者ツール	13
db2atld - 自動ローダー	14
db2batch - ベンチマーク・ツール	15
db2bfd - バインド・ファイル記述ツール	22
db2cc - コントロール・センターの開始	23
db2cdbcr - データウェアハウジング用の制御データベースの作成	25
db2cfexp - 接続構成エクスポート・ツール	26
db2cfimp - 接続構成インポート・ツール	28
db2cidmg - リモート・データベース移行	29
db2ckbkp - バックアップの検査	31
db2ckmig - データベース事前移行ツール	34
db2cli - DB2 対話機能 CLI	35
db2cmd - DB2 コマンド・ウィンドウのオープン	37
db2dclgn - 宣言生成プログラム	39
db2drdat - DRDA トレース	42
db2empfa - 複数ページ・ファイル割り振りの使用可能化	44
db2eva - イベント・アナライザー	45
db2evmon - 事象モニター生産性向上ツール	47
db2exfmt - Explain 表フォーマット・ツール	49
db2expln - DB2 SQL Explain ツール	50
db2flsn - ログ順序番号の検出	51
db2fs - First Steps	53
db2gov - DB2 管理プログラム	54
db2govlg - DB2 管理プログラム・ログ照会	56
db2icrt - インスタンスの作成	57
db2idrop - インスタンスの除去	60
db2ilist - インスタンスのリスト	62
db2imigr - インスタンスの移行	63

db2ipxad - IPX/SPX インターネットワーク・アドレスの獲得	64
db2iupdt - インスタンスの更新	65
db2ldcfg - LDAP 環境の構成	68
db2licm - ライセンス管理ツール	69
db2look - DB2 統計および DDL 抽出ツール	71
db2move - データベース移動ツール	77
db2mscs - Windows NT フェールオーバー・ユーティリティのセットアップ	82
db2nchg - データベース区画サーバー構成の変更	83
db2ncrt - インスタンスへのデータベース区画サーバーの追加	85
db2ndrop - インスタンスからのデータベース区画サーバーの除去	88
db2perfc - データベース・パフォーマンス値のリセット	90
db2perfi - パフォーマンス・カウンター登録ユーティリティ	91
db2perfr - パフォーマンス・モニター登録ツール	92
db2profc - DB2 SQLJ プロファイル・カスタマイザー	93
db2profp - DB2 SQLJ プロファイル・プリンター	96
db2rbind - すべてのパッケージの再バインド	98
db2sampl - サンプル・データベースの作成	100
db2set - DB2 プロファイル・レジストリー・コマンド	102
db2sql92 - SQL92 準拠 SQL ステートメント・プロセッサ	106
db2start - DB2 の開始	109
db2stop - DB2 の停止	110
db2sync - DB2 シンクロナイザーの開始	111
db2tbst - 表スペース状態の獲得	112
db2trc - トレース	113
db2uiddl - V5 セマンティクスへの固有索引変換の準備	117
db2untag - コンテナ・タグの解放	119

第2章 コマンド行プロセッサ (CLP)	121
---------------------------------------	------------

コマンド行プロセッサの呼び出し	122	GET CLI CONFIGURATION	274
コマンド行プロセッサのオプション	125	GET CONNECTION STATE	277
コマンド行プロセッサの戻りコード	133	GET DATABASE CONFIGURATION	279
コマンド行プロセッサの使用	135	GET DATABASE MANAGER	
コマンド・ファイルでのコマンド行プロセ		CONFIGURATION	292
ッサの使用	136	GET DATABASE MANAGER MONITOR	
コマンド行プロセッサの設計	136	SWITCHES	307
CLP 使用上の注意	139	GET INSTANCE	309
第3章 CLP コマンド	143	GET MONITOR SWITCHES	310
DB2 CLP コマンド	143	GET SNAPSHOT	312
ACTIVATE DATABASE.	148	HELP	330
ADD DATALINKS MANAGER	150	IMPORT	332
ADD NODE.	152	INITIALIZE TAPE	359
ATTACH.	154	INVOKE STORED PROCEDURE.	360
BACKUP DATABASE	157	LIST ACTIVE DATABASES	362
BIND	165	LIST APPLICATIONS	364
CALL	184	LIST COMMAND OPTIONS	367
CATALOG APPC NODE	186	LIST DATABASE DIRECTORY	369
CATALOG APPCLU NODE	189	LIST DATALINKS MANAGERS	373
CATALOG APPN NODE	191	LIST DCS APPLICATIONS.	374
CATALOG DATABASE.	194	LIST DCS DIRECTORY.	378
CATALOG DCS DATABASE	199	LIST DRDA INDOUBT TRANSACTIONS	380
CATALOG GLOBAL DATABASE	202	LIST HISTORY	382
CATALOG IPX/SPX NODE	204	LIST INDOUBT TRANSACTIONS	386
CATALOG LDAP DATABASE	208	LIST NODE DIRECTORY	391
CATALOG LDAP NODE	213	LIST NODEGROUPS.	394
CATALOG LOCAL NODE	215	LIST NODES	397
CATALOG NAMED PIPE NODE.	218	LIST ODBC DATA SOURCES	398
CATALOG NETBIOS NODE	221	LIST PACKAGES/TABLES	400
CATALOG ODBC DATA SOURCE	224	LIST TABLESPACE CONTAINERS	403
CATALOG TCP/IP NODE	226	LIST TABLESPACES.	405
CHANGE DATABASE COMMENT	230	LOAD.	411
CHANGE ISOLATION LEVEL	232	LOAD QUERY.	453
CREATE DATABASE	235	MIGRATE DATABASE	455
DEACTIVATE DATABASE.	243	PRECOMPILE PROGRAM	457
DEREGISTER	245	PRUNE HISTORY/LOGFILE	487
DESCRIBE	247	QUERY CLIENT	489
DETACH.	250	QUIESCE TABLESPACES FOR TABLE	491
DROP DATABASE	251	QUIT	495
DROP NODE VERIFY	254	REBIND	496
ECHO.	256	RECONCILE	500
EXPORT	257	REDISTRIBUTE NODEGROUP	505
FORCE APPLICATION	267	REFRESH LDAP	509
GET ADMIN CONFIGURATION	269	REGISTER	511
GET AUTHORIZATIONS	272	REORGANIZE TABLE	517
		REORGCHK.	520

RESET ADMIN CONFIGURATION	528	UPDATE MONITOR SWITCHES	613
RESET DATABASE CONFIGURATION	531	UPDATE HISTORY FILE	615
RESET DATABASE MANAGER CONFIGURATION	533	第4章 コマンド行構造化照会言語ステートメントの使用	617
RESET MONITOR	535	付録A. 構文図の読み方	627
RESTART DATABASE	537	付録B. 命名規則	631
RESTORE DATABASE	539	付録C. DB2 ライブラリーの用法	633
REWIND TAPE	549	DB2 PDF ファイルおよびハードコピー版資料	633
ROLLFORWARD DATABASE.	550	DB2 情報	633
RUNSTATS	561	PDF 資料の印刷	645
SET CLIENT	566	印刷資料の注文方法	645
SET RUNTIME DEGREE	570	DB2 オンライン文書	646
SET TABLESPACE CONTAINERS	572	オンライン・ヘルプへのアクセス	646
SET TAPE POSITION	575	オンライン情報の表示	648
START DATABASE MANAGER	576	DB2 ウィザードの使用	650
STOP DATABASE MANAGER	582	文書サーバーのセットアップ	652
TERMINATE	586	オンライン情報の検索	653
UNCATALOG DATABASE.	587	付録D. 特記事項	655
UNCATALOG DCS DATABASE	589	商標	658
UNCATALOG LDAP DATABASE	591	索引	661
UNCATALOG LDAP NODE	593	IBM と連絡をとる	675
UNCATALOG NODE.	595	製品情報	675
UNCATALOG ODBC DATA SOURCE	597		
UPDATE ADMIN CONFIGURATION	599		
UPDATE CLI CONFIGURATION.	602		
UPDATE COMMAND OPTIONS	604		
UPDATE DATABASE CONFIGURATION	606		
UPDATE DATABASE MANAGER CONFIGURATION	608		
UPDATE LDAP NODE	610		

本書について

本書は、データベース管理機能を実行するためのシステム・コマンドおよび DB2 ユニバーサル・データベース・コマンド行プロセッサ (CLP) の使用方法について説明します。

本書の対象読者

本書の読者は、データベース管理について理解しており、構造化照会言語 (SQL) の知識があることが前提となっています。

本書の構成

本書では、CLP を使用する際に必要となる参照情報を提供します。

以下のトピックについて説明します。

『第1章 システム・コマンド』

データベース・マネージャーにアクセスするためにオペレーティング・システム・コマンド・プロンプトまたはシェル・スクリプトに入力できるコマンドを記述します。

『第2章 コマンド行プロセッサ (CLP)』

コマンド行プロセッサを呼び出し、使用方法と CLP オプションについて説明します。

『第3章 CLP コマンド』

すべてのデータベース・マネージャー・コマンドを説明します。

『第4章 コマンド行構造化照会言語ステートメントの使用』

コマンド行から SQL ステートメントを使用する方法を示します。

『付録A. 構文図の読み方』

構文図の規則を説明します。

『付録B. 命名規則』

データベースおよび表などのオブジェクトを命名するときの規則について説明します。

第1章 システム・コマンド

この章では、データベース・マネージャーへのアクセスおよび保守のために、オペレーティング・システムのコマンド・プロンプトで入力するか、またはシェル・スクリプトに含めることが可能なコマンドについて説明します。

注: ディレクトリー・パス中の斜線 (/) は UNIX ベースのシステムだけに用いられるもので、OS/2 および Windows オペレーティング・システムの円記号 (¥) に相当します。

コマンドの説明の編成方法

各コマンドの短い説明の後に、以下の項目の一部またはすべてが続きます。

効力範囲

インスタンス内でのコマンド操作の効力範囲。単一ノード・システムでは、効力範囲はその単一ノードに限定されます。マルチノード・システムでの効力範囲は、ノード構成ファイル (db2nodes.cfg) に定義されている論理ノードすべてです。

許可

コマンドを正常に呼び出すために必要な権限。

必須接続

データベース、インスタンス、なし、または接続の確立のどれかです。機能が正常に作動するために、データベース接続またはインスタンス接続が必要かどうか、または接続は必要ないかを示します。特定のコマンドを発行する前に、データベースまたはインスタンスへの明示的な接続が必要である場合もあります。データベース接続またはインスタンス接続を必要とするコマンドは、ローカルまたはリモートのどちらかで実行することができます。データベース接続とインスタンス接続のいずれも必要ではないコマンドはリモートには実行できません。そのため、そのようなコマンドをクライアント環境で発行すると、コマンドの影響はそのクライアント内にしか及びません。データベース接続およびインスタンス接続の詳細については、[管理の手引き](#)を参照してください。

コマンド構文

構文図については、627ページの『付録A. 構文図の読み方』を参照してください。

コマンド・パラメーター

コマンドとともに使用可能なパラメーターの説明。

使用上の注意

その他の情報。

以下も参照

関連情報の相互参照です。

db2admin - DB2 管理サーバー

このユーティリティーは、DB2 管理サーバーの管理に使用します。DB2 管理サーバーの詳細については、[管理の手引き](#) を参照してください。

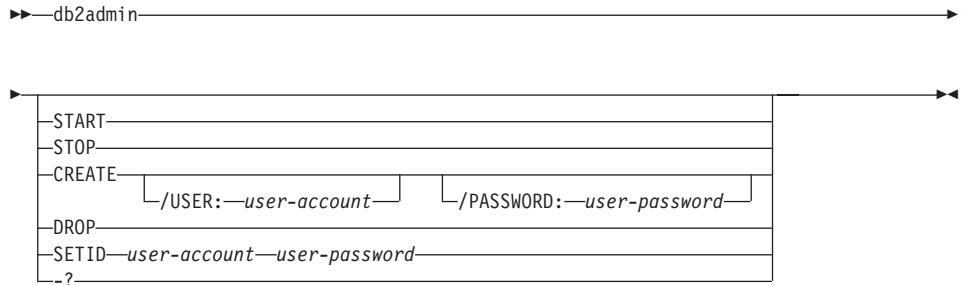
許可

Windows NT 上のローカル管理者、または UNIX ベースのシステムおよび OS/2 上の SYSADM。

必須接続

なし

コマンド構文



コマンド・パラメーター

注: パラメーターを指定せず、かつ DB2 管理サーバーが存在する場合、このコマンドは DB2 管理サーバー・インスタンスの名前を戻します。

START

DB2 管理サーバーを開始します。

STOP DB2 管理サーバーを停止します。

CREATE /USER: user-account /PASSWORD: user-password

DB2 管理サーバーを作成します。ユーザー名およびパスワードを指定した場合、DB2 管理サーバー・インスタンスがこのユーザー・アカウントに関連付けられます。指定した値が無効であると、ユーティリティーは認証エラーを戻します。指定したユーザー・アカウントは有効な SQL 識別子でなければならず、セキュリティ・データベース内になければなりません。DB2 管理サーバーの機能すべてにアクセスできるように、ユーザー・アカウントを指定することをお勧めします。

db2admin - DB2 管理サーバー

注: このパラメーターは UNIX ではサポートされていません。

DROP DB2 管理サーバー・インスタンスを削除します。

注: このパラメーターは UNIX ベースのシステムではサポートされていません。

SETID user-account/user-password

DB2 管理サーバー・インスタンスに関連付けられたユーザー・アカウントを設定または修正します。

-? ヘルプ情報を表示します。このオプションを指定すると、他のすべてのオプションは無視され、ヘルプ情報だけが表示されます。

db2adutl - TSM アーカイブ・イメージによる作業

Tivoli Storage Manager (以前の AD SM) を使用して保管した、バックアップ、ログ、およびロード・コピー・イメージの、照会、抽出、検査、および削除をユーザーに許可します。

UNIX ベースのシステムでは、このユーティリティーは INSTHOME/sql lib/misc ディレクトリーにあり、OS/2 および Windows オペレーティング・システムでは %sql lib%misc ディレクトリーにあります。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

許可

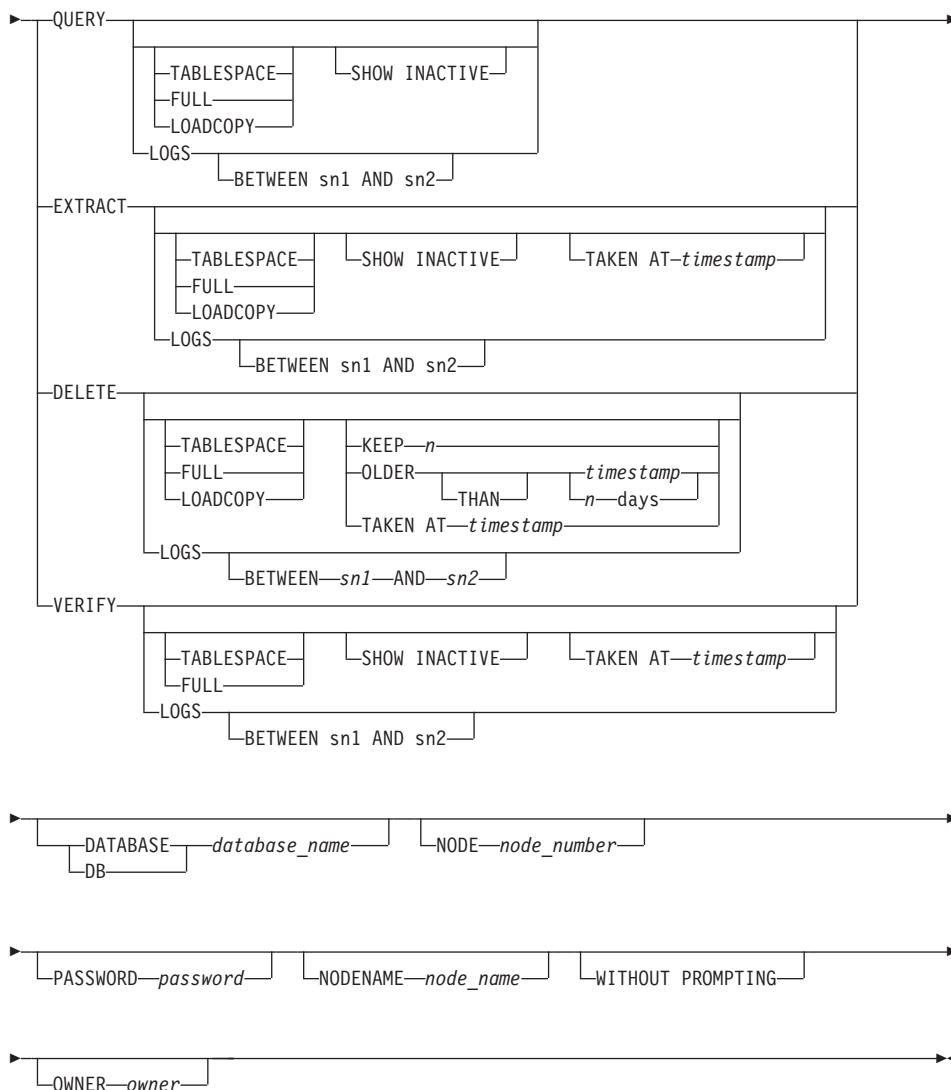
なし

必須接続

なし

コマンド構文

▶▶—db2adutl—————→



コマンド・パラメーター

QUERY

TSM サーバーで DB2 オブジェクトを照会します。

EXTRACT

TSM サーバーからの DB2 オブジェクトを、ローカル・マシンにある現行ディレクトリーにコピーします。

DELETE

バックアップ・オブジェクトを非活動化するか、または TSM サーバーにあるログ・アーカイブを削除します。

VERIFY

サーバーにあるバックアップ・コピーに対して整合性検査を実行します。

注: このパラメーターを指定すると、バックアップ・イメージ全体がネットワークを介して転送されます。

TABLESPACE

表スペース・バックアップ・イメージだけを組み込みます。

FULL 完全データベース・バックアップ・イメージだけを組み込みます。

LOADCOPY

ロード・コピー・イメージだけを組み込みます。

LOGS ログ・アーカイブ・イメージだけを組み込みます。

BETWEEN *sn1* AND *sn2*

ログ順序番号 1 とログ順序番号 2 との間のログの使用を指定します。

SHOW INACTIVE

非活動化されているバックアップ・オブジェクトを組み込みます。

TAKEN AT *timestamp*

タイム・スタンプを基準としてバックアップ・イメージを指定します。

KEEP *n*

タイム・スタンプで最新の *n* 個を除き、指定したタイプのすべてのオブジェクトを非活動化します。

OLDER THAN *timestamp* or *n* days

timestamp または *n* 日より前のタイム・スタンプが付けられているオブジェクトを非活動化することを指定します。

DATABASE *database_name*

指定したデータベース名に関連したオブジェクトだけを対象にします。

NODE *node_number*

指定したノード番号で作成されたオブジェクトだけを対象にします。

PASSWORD *password*

このノードの TSM クライアント・パスワードを指定します (要求される場合)。データベースが指定されたもののパスワードが提供されない

db2adutl - TSM アーカイブ・イメージによる作業

場合には、 *tsm_password* データベース構成パラメーターに指定した値が TSM に渡されます。渡されない場合には、パスワードは使用されません。

NODENAME *node_name*

特定の TSM ノード名に関連したイメージだけを対象にします。

WITHOUT PROMPTING

オブジェクトの削除の前に、確認を求めるプロンプトが出ないようにします。

OWNER *owner*

指定した所有者により作成されたオブジェクトだけを対象にします。

例

以下に示すのは、db2 backup database rawsampl use tsm の出力例です。

```
Backup successful. The timestamp for this backup is : 19970929130942
```

```
db2adutl query
Query for database RAWSAMPL
```

```
Retrieving full database backup information.
  full database backup image: 1, Time: 19970929130942,
                                Oldest log: S0000053.LOG, Sessions used: 1
  full database backup image: 2, Time: 19970929142241,
                                Oldest log: S0000054.LOG, Sessions used: 1
```

```
Retrieving table space backup information.
  table space backup image: 1, Time: 19970929094003,
                                Oldest log: S0000051.LOG, Sessions used: 1
  table space backup image: 2, Time: 19970929093043,
                                Oldest log: S0000050.LOG, Sessions used: 1
  table space backup image: 3, Time: 19970929105905,
                                Oldest log: S0000052.LOG, Sessions used: 1
```

```
Retrieving log archive information.
  Log file: S0000050.LOG
  Log file: S0000051.LOG
  Log file: S0000052.LOG
  Log file: S0000053.LOG
  Log file: S0000054.LOG
  Log file: S0000055.LOG
```

以下に示すのは、db2adutl delete full taken at 19950929130942 db rawsampl の出力例です。

```
Query for database RAWSAMPL
```

```
Retrieving full database backup information. Please wait.
```

```
full database backup image: RAWSAMPL.0.db26000.0.19970929130942.001
```

Do you want to deactivate this backup image (Y/N)? y

Are you sure (Y/N)? y
db2adutl query

Query for database RAWSAMPL

Retrieving full database backup information.

full database backup image: 2, Time: 19950929142241,
Oldest log: S0000054.LOG, Sessions used: 1

Retrieving table space backup information.

table space backup image: 1, Time: 19950929094003,
Oldest log: S0000051.LOG, Sessions used: 1

table space backup image: 2, Time: 19950929093043,
Oldest log: S0000050.LOG, Sessions used: 1

table space backup image: 3, Time: 19950929105905,
Oldest log: S0000052.LOG, Sessions used: 1

Retrieving log archive information.

Log file: S0000050.LOG

Log file: S0000051.LOG

Log file: S0000052.LOG

Log file: S0000053.LOG

Log file: S0000054.LOG

Log file: S0000055.LOG

db2advis - DB2 索引アドバイザー

1 つ以上の SQL ステートメントに作成する索引についてユーザーにアドバイスをします。関連 SQL ステートメントのグループは、ワークロードと呼ばれます。ユーザーは、ワークロード中の各ステートメントの重要性をランク付けし、ワークロード中の各ステートメントが実行される頻度を指定することができます。各表ごとに推奨される索引、それらに応じて派生する統計、およびそれぞれを作成する DDL は、ユーザー作成の表 ADVISE_INDEX に書き込まれます。

注: 構造型列は、このコマンドの実行時には考慮されません。

許可

データベースへの読み取りアクセス。 Explain 表への読み取りおよび書き込みアクセス。

必須接続

なし。このコマンドは、データベース接続を確立します。

コマンド構文

```
db2advis -d database-name -w workload-name -s "statement" -i filename  
-a userid /passwd -l disk-limit -t max-advise-time  
-h
```

コマンド・パラメーター

-d database-name

接続の確立先のデータベースの名前を指定します。

-w workload-name

索引がアドバイスされるワークロードの名前を指定します。この名前は ADVISE_WORKLOAD 表で使用されます。

-s "statement"

索引がアドバイスされる単一の SQL ステートメントのテキストを指定します。ステートメントは必ず二重引用符で囲んでください。

-i filename

1 つ以上の SQL ステートメントが入っている入力ファイルの名前を指定します。省略時は標準入力です。注釈テキストは、各行の先頭に 2 つのハイフンを付けて -- <注釈> で表します。ステートメントは必ずセミコロンで区切ってください。

ワークロード中の各ステートメントが実行される頻度は、次の行を入力ファイルに挿入することによって変更できます。

```
--#SET FREQUENCY <x>
```

頻度は、ファイル中何回でも更新できます。

-a userid/passwd

データベースへの接続に使用する名前およびパスワード。パスワードが指定される場合、斜線 (/) を含めなければなりません。

-l disk-limit

既存のスキーマですべての索引に使用可能な最大 MB を指定します。省略時値は、区画 (64 GB) ごとの索引の最大サイズにおける、データベース・マネージャー限度です。

-t max-advise-time

最大許可時間 (分) を指定し、操作を完了します。省略時値は 10 です。無制限の時間は、ゼロの値によって指定されます。

-h ヘルプ情報を表示します。このオプションを指定すると、他のすべてのオプションは無視され、ヘルプ情報だけが表示されます。

例

次の例では、ユーティリティーは PROTOTYPE データベースに接続し、解決策において制限なしで ADDRESSES 表に索引を推奨します。

```
db2advis -d prototype -s "select * from addresses a
where a.zip in ('93213', '98567', '93412')
and (company like 'IBM%' or company like '%otus')"
```

次の例では、ユーティリティーは PROTOTYPE データベースに接続し、ワークロード名が "production" に等しい ADVISE_WORKLOAD 表の照会で、53 MB を超えない索引を推奨します。解決策を見つけるための最大許可時間は 20 分です。

```
db2advis -d prototype -w production -l 53 -t 20
```

db2advis - DB2 索引アドバイザー

最後の例では、db2advis.in という名前の入力ファイルに SQL ステートメント、および各ステートメントが実行される頻度の指定が含まれています。

```
--#SET FREQUENCY 100
SELECT COUNT(*) FROM EMPLOYEE;
SELECT * FROM EMPLOYEE WHERE LASTNAME='HAAS';
--#SET FREQUENCY 1
SELECT AVG(BONUS), AVG(SALARY) FROM EMPLOYEE
GROUP BY WORKDEPT ORDER BY WORKDEPT;
```

ユーティリティーは SAMPLE データベースに接続し、入力ファイル内の照会によって参照される各表ごとに索引を推奨します。 解決策を見つけるための最大許可時間は 5 分です。

```
db2advis -d sample -f db2advis.in -t 5
```

使用上の注意

動的 SQL ステートメントの場合、ステートメントが実行される頻度は、次のようにモニターから獲得されます。

1. 次のように発行します。

```
db2 reset monitor for database <database-alias>
```

適切な時間間隔で待機します。

2. 次のように発行します。

```
db2 get snapshot for dynamic sql on <database-alias>
write to file
```

3. 次のように発行します。

```
db2 "insert into advise_workload
(select 'myworkload', 0, stmt_text, cast(generate_unique() as
char(254)), num_executions, 1, num_executions, 0, 0
from table(SYSFUN.SQLCACHE_SNAPSHOT()) as correlations_name"
```

ワークロードの各 SQL ステートメントの省略時頻度は 1 で、省略時の重要度も 1 です。 generate_unique() 関数は、その SQL ステートメントのより分かりやすい説明になるように、ユーザーによって更新できるステートメントに、固有の識別子を割り当てます。

db2audit - 監査機能管理者ツール

DB2 には、未知または予期しないデータ・アクセスの検出を支援する監査機能が備わっています。DB2 監査機能は、事前定義された一連のデータベース事象の監査証跡を生成し、その保守を許可します。この機能で生成されたレコードは、監査ログ・ファイルに保持されます。これらのレコードを分析すると、システムの誤用を識別する使用パターンが明らかになります。識別することができれば、システムのそのような誤用を削減または除去する処置をとることができます。監査機能はインスタンス・レベルで動作し、すべてのインスタンス・レベルの活動とデータベース・レベルの活動を記録します。

監査機能の許可ユーザーは、**db2audit** を使用することにより、監査機能内で以下の処置を制御することができます。

- DB2 インスタンス内で監査可能事象の記録を開始する。
- DB2 インスタンス内で監査可能事象の記録を停止する。
- 監査機能の振る舞いを構成する。
- 記録する監査可能事象の区分を選択する。
- 現在の監査構成の説明を要求する。
- 保留中の監査レコードをインスタンスからフラッシュし、監査ログに書き込む。
- 形式設定して監査ログからコピーすることにより、監査レコードをフラット・ファイルまたは ASCII 区切りファイルに抽出する。抽出を行う理由は 2 つのうちどちらかです。ログ・レコードの分析を準備するためか、ログ・レコードのプルーニングを準備するためです。
- 現在の監査ログから監査レコードをプルーニングする。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

db2atld - 自動ローダー

db2atld - 自動ローダー

自動ローダーは、MPP 環境でデータを区分化またはロードするツールです。このユーティリティーは以下の処理を行います。

- あるシステム (たとえば、MVS) から AIX システム (RS/6000 または SP2) へデータを転送する
- データを並行して区分化する
- 対応している複数のノードでデータを同時にロードする

このコマンドの詳細については、[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。

以下も参照

411ページの『LOAD』

db2batch - ベンチマーク・ツール

フラット・ファイルまたは標準入力のどちらかから SQL ステートメントを読み取り、ステートメントを動的に準備および記述し、応答セットを戻します。

このユーティリティには、EEE 環境での並列エクスポート機能もあります。この機能では、エクスポートするデータを定義する照会を簡単に実行でき、個々の区画で、その区画にあるエクスポート・データの部分が入るファイルを作成することができます。

許可

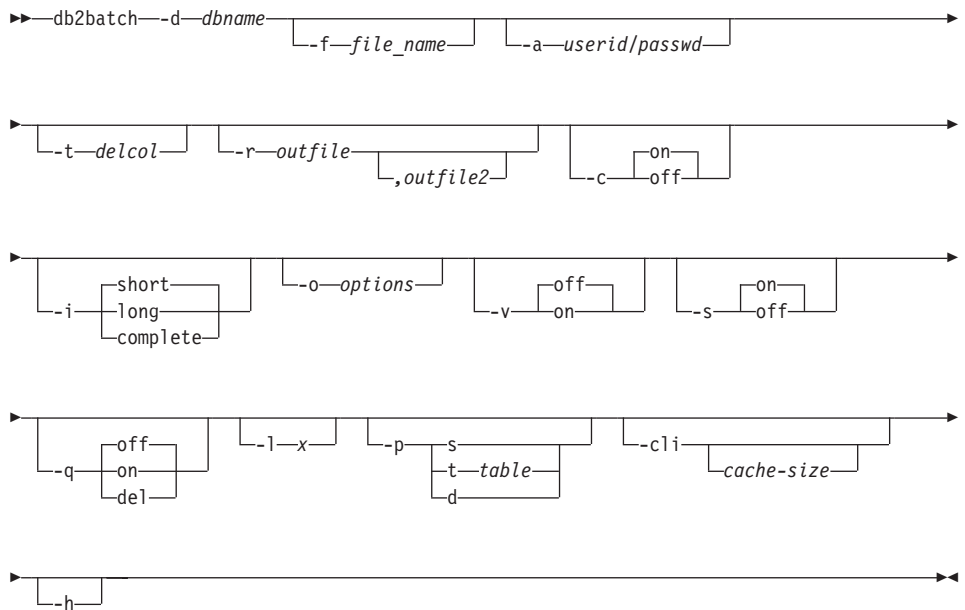
読み取られる SQL ステートメントが必要とするものと同じ権限レベル。

並列モードでは、ユーザーは **db2_all** を実行する権限がなければなりません。詳しくは、[管理の手引き](#) を参照してください。

必須接続

なし。このコマンドは、データベース接続を確立します。

コマンド構文



コマンド・パラメーター

-d dbname

SQL ステートメントが適用されるデータベースの別名。省略時値は、**DB2DBDFT** 環境変数の値です。

-f file_name

SQL ステートメントが入っている入力ファイルの名前。省略時は標準入力です。

注釈テキストは、各行の先頭に 2 つのハイフンを付けて `-- <注釈>` で表します。注釈を出力にも含めるときは、次のように注釈にマークを付けます。 `--#COMMENT <注釈>`。

ブロック は、一まとまりとして処理されるいくつかの SQL ステートメントからなっています。つまり、ステートメントで使用する情報を 1 つずつ収集するのではなく、すべてのステートメントに必要な情報を一度に収集します。照会ブロックの開始は、`--#BGBLK` で表します。照会ブロックの終了は、`--#EOBLK` で表します。

1 つまたは複数の制御オプションを指定するには、次のようにします。`--#SET <制御オプション> <値>`。有効な制御オプションは、以下のとおりです。

ROWS_FETCH

応答セットから取り出す行数。有効な値は `-1 ~ n` です。省略時値は `-1` (すべての行を取り出す) です。

ROWS_OUT

取り出された行のうち出力へ送られる行数。有効な値は `-1 ~ n` です。省略時値は `-1` (取り出された行をすべて出力へ送る) です。

PERF_DETAIL

戻されるパフォーマンス情報のレベルを指定します。有効な値は以下のとおりです。

- 0 時間情報を戻さない。
- 1 経過時間のみ戻す。
- 2 経過時間と CPU 時間を戻す。
- 3 モニター情報の要約を戻す。
- 4 データベース・マネージャー、データベース、アプリケーション、およびステートメントのスナップショットを戻す。(自動コミットがオフになっており、かつ

ステートメント・ブロックではなく単一ステートメントを処理している場合は、ステートメントのスナップショットが戻されます。)

- 5** データベース・マネージャー、データベース、アプリケーション、およびステートメントのスナップショットを戻す。(自動コミットがオフになっており、かつステートメント・ブロックではなく単一ステートメントを処理している場合は、ステートメントのスナップショットが戻されます。) バッファ・プール、表スペース、および FCM も戻します (FCM スナップショットは複数モード環境でのみ使用可能)。

省略時値は 1 です。2 以上の値は、DB2 バージョン 2 サーバーでのみ有効です。

DELIMITER

1 文字か 2 文字のステートメント終結区切り文字です。省略時値はセミコロン (;) です。

SLEEP

スリープの秒数。有効な値は 1 ~ n です。

PAUSE

継続するかどうかの入力を要求するプロンプトをユーザーに出します。

TIMESTAMP

タイム・スタンプを生成します。

-a userid/passwd

データベースへの接続に使用する名前およびパスワード。斜線 (/) を含めなければなりません。

-t delcol

1 文字の列区切り記号を指定します。

注: タブの列区切り文字を含めるには、-t TAB を使用します。

-r outfile

照会結果が入る出力ファイル。任意指定の出力ファイル 2 には、結果の要約が入ります。省略時値は標準出力です。

-c

各 SQL ステートメントの実行による変更を自動的にコミットします。

-i

経過時間の間隔 (秒単位)。

short カーソルのオープン、取り出しの完了、およびカーソルのクローズの所要時間です。

long ある照会の開始から次の照会の開始までの経過時間です。PAUSE と SLEEP の時間、およびコマンド・オーバーヘッドを含みます。

complete

準備、実行、および取り出しの時間。別個に表示します。

-o options

制御オプション。有効なオプションは以下のとおりです。

f rows_fetch

応答セットから取り出す行数。有効な値は $-1 \sim n$ です。省略時値は -1 (すべての行を取り出す) です。

r rows_out

取り出された行のうち出力へ送られる行数。有効な値は $-1 \sim n$ です。省略時値は -1 (取り出された行をすべて出力へ送る) です。

p perf_detail

戻されるパフォーマンス情報のレベルを指定します。有効な値は以下のとおりです。

- 0** 時間情報を戻さない。
- 1** 経過時間のみ戻す。
- 2** 経過時間と CPU 時間を戻す。
- 3** モニター情報の要約を戻す。
- 4** データベース・マネージャー、データベース、アプリケーション、およびステートメントのスナップショットを戻す。(自動コミットがオフになっており、かつステートメント・ブロックではなく単一ステートメントを処理している場合は、ステートメントのスナップショットが戻されます。)
- 5** データベース・マネージャー、データベース、アプリケーション、およびステートメントのスナップショットを戻す。(自動コミットがオフになっており、かつステートメント・ブロックではなく単一ステートメントを処理している場合は、ステートメントのスナップショットが戻されます。) バッファ・プール、表ス

ベース、および FCM も戻します (FCM スナップショットは複数モード環境でのみ使用可能)。

o query_optimization_class

照会最適化クラスを設定する。有効な値の詳細については、[管理の手引き](#) を参照してください。

e explain_mode

db2batch 実行時の explain モードを設定する。このコマンドを使用する前に、`Explain` 表を作成しておく必要があります。有効な値は以下のとおりです。

- 0** 照会のみ実行 (省略時値)。
- 1** `Explain` 表のみ移植。このオプションは、`Explain` 表を移植し、`Explain` スナップショットを取得させます。
- 2** `Explain` 表を移植し、照会を実行する。このオプションは、`Explain` 表を移植し、`Explain` スナップショットを取得させます。

- v** 冗長。照会処理中に標準エラーに情報を送信します。省略時値は OFF です。
- s** 要約表。照会または照会のブロックごとに、要約表を提供します。その表には、経過時間 (選択時)、CPU 時間 (選択時)、取り出された行数、印刷された行数が含まれます。経過時間および CPU 時間が収集された場合、その算術平均および幾何平均が算出されます。
- q** 照会の出力。有効な値は以下のとおりです。
 - on** 照会の区切られていない 出力のみを印刷する。
 - off** 照会と関連情報すべての出力を印刷する。これが省略時値です。
 - del** 照会の区切られた 出力のみを印刷する。
- l x** 終了文字を指定します。
- p** 並列 (EEE のみ)。このモードでは `SELECT` ステートメントしかサポートされていません。出力名には、完全修飾パスがなければなりません。有効な値は以下のとおりです。
 - s** 単一表または照合結合照会。 `SELECT` ステートメントには列関数だけを含めることはできません。これは、照会に追加される `NODENUMBER` 関数の要件です。このオプションを指定すると、`NODENUMBER` 関数が

照会の WHERE 節に追加され、一時表は作成されません。このオプションは、照会の FROM 節に単一表が入っている場合、または FROM 節に含まれる単一表が連結されている場合にのみ有効です。

このオプションが指定され、照会に GROUP BY 節が含まれる場合、GROUP BY に指定された列は、表区分キーのスーパーセットでなければなりません。

t table

エクスポートを取り込むステージング表として使用する既存の表の名前を指定します。照会に FROM 節の複数の表が入っていて、その表が連結されていない場合、結果セットは指定した表に挿入され、SELECT が同時にすべての区画に対して発行され、エクスポート・データが入るファイルが生成されます。

d

INSERT INTO ステートメントで使用するシステム表を IBMDEFAULTGROUP に作成します。照会に FROM 節の複数の表が入っていて、その表が連結されていない場合、結果セットは指定した表に挿入され、SELECT が同時にすべての区画に対して発行され、エクスポート・データが入るファイルが生成されます。

ローカル 出力ファイルを指定した (**-r** オプションを使用する) 場合、各ノードからの出力は、各ノードで同じ名前を持つ別々のファイルに入られます。NFS マウント・ファイル・システム上のファイルが指定された場合、出力はすべてこのファイルに入られます。

-cli CLI モードで **db2batch** を実行する。省略時では、組み込み動的 SQL を使用するようになっています。ステートメント・メモリーは、*cache-size* パラメーターを使用して、手動で設定できます。

cache-size

ステートメント・メモリーのサイズ。ステートメントの数で表されます。省略時値は 25 です。ユーティリティーがすでに準備済みになっている SQL ステートメントを検出した場合、古いプランを再利用します。このパラメーターは、**db2batch** を CLI モードで実行する場合だけ設定できます。

-h ヘルプ情報を表示します。このオプションを指定すると、他のすべてのオプションは無視され、ヘルプ情報だけが表示されます。

使用上の注意

SQL ステートメントの長さは、65 535 文字まで可能です。入力ファイル中ではテキスト行が 3 898 文字を超えることはできないので、長いステートメントは数行に分割する必要があります。ステートメントの末尾は区切り文字 (省略時値はセミコロン) にしなければなりません。

SQL ステートメントは、反復可能読取り (RR) 分離レベルで実行されます。

以下も参照

db2batch の使用の詳細については、[管理の手引き](#) を参照してください。

並列エクスポートの詳細な説明については、[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。

106ページの『db2sql92 - SQL92 準拠 SQL ステートメント・プロセッサ』

db2bfd - バインド・ファイル記述ツール

db2bfd - バインド・ファイル記述ツール

バインド・ファイルの内容を表示します。このユーティリティーは、バインド・ファイルを作成する際に使用したプリコンパイル・オプションを表示するだけでなく、バインド・ファイル内の SQL ステートメントを調べ、検査するためにも使用できます。アプリケーションのバインド・ファイルに関連した問題を判別するのに役立ちます。

許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

filespec

内容が表示されるバインド・ファイルの名前です。

- h ヘルプ情報を表示します。このオプションを指定すると、他のすべてのオプションは無視され、ヘルプ情報だけが表示されます。
- b バインド・ファイル・ヘッダーを表示します。
- s SQL ステートメントを表示します。
- v ホスト変数宣言を表示します。

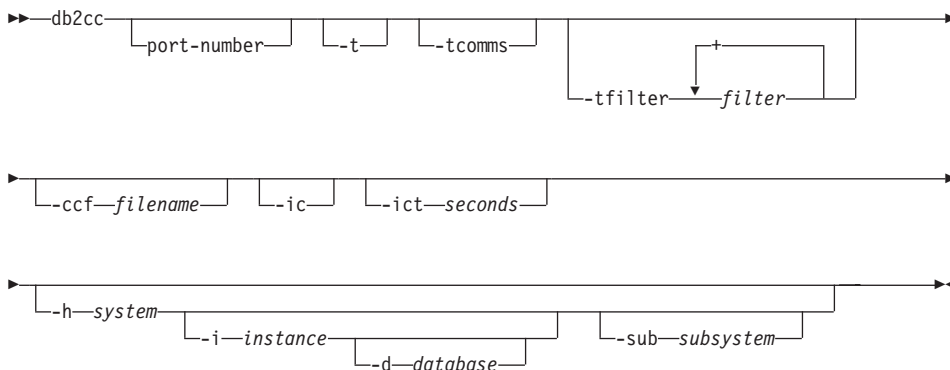
db2cc - コントロール・センターの開始

コントロール・センターを開始します。コントロール・センターは、データベース・オブジェクト (データベース、表、およびパッケージなど) とそれらの相互関係を管理するために使用する、グラフィカル・インターフェースです。

許可

sysadm

コマンド構文



コマンド・パラメーター

port-number

予約済みの DB2JD ポート番号を指定します。たとえば、6790 となります。省略時値は 6789 です。

-t 初期化コードに対する NavTrace をオンにします。

-tcomms

トレースを通信イベントのみを対象とするように制限します。

-tfilter filter

トレースを、指定したフィルター (1 つまたは複数) を含む項目のみを対象とするように制限します。

-ccf filename

指定したファイル名で初期設定したコマンド・センターをオープンします。

-ic インフォメーション・センターをオープンします。

db2cc - コントロール・センターの開始

-ict seconds

アイドル接続タイマー。指定した秒数が経過すると、コントロール・センターによって保守されているプールにあるアイドル接続はクローズされます。省略時タイマーは 30 分です。

-h system

システムに関連してコントロール・センターをオープンします。

-i instance

インスタンスに関連してコントロール・センターをオープンします。

-d database

データベースに関連してコントロール・センターをオープンします。

-sub subsystem

サブシステムに関連してコントロール・センターをオープンします。

使用上の注意

注: 以下のコマンド、

- 269ページの『GET ADMIN CONFIGURATION』
- 528ページの『RESET ADMIN CONFIGURATION』
- 599ページの『UPDATE ADMIN CONFIGURATION』

は、通常、コマンド・センターからは発行できません。コマンド・センターから発行できるのは、コマンド・センターが DB2 管理サーバー・インスタンスから開始された場合だけです。他のインスタンスからこれらのコマンドを実行するには、コマンド行プロセッサ (CLP) を使用してください。

コントロール・センターの全般的な情報については、[管理の手引き](#)を参照してください。詳細は、コントロール・センターのオンライン・ヘルプ機能で提供されます。

db2cdbcr - データウェアハウジング用の制御データベースの作成

現行のインスタンスに制御データベースを作成します。制御データベースは、定義されたデータベース・オブジェクトについてのメタ・データを保管します。このコマンドは、データウェアハウジングの実行時に使用されます。

データウェアハウジングの詳細については、[データウェアハウスセンター 管理の手引き](#)を参照してください。

許可

インスタンスの所有者により実行されなければなりません。

コマンド構文

```
▶▶ db2cdbcr [-d] [-n CDB_name] ▶▶
```

コマンド・パラメーター

-d 一般的な標準デバッグ・オプション。

-n CDB_name
作成される制御データベースの名前。

注: 指定したデータベースが存在しない場合、**db2cdbcr** がそれを作成します。

db2cfexp - 接続構成エクスポート・ツール

接続構成情報を、エクスポート・プロファイルにエクスポートします。後でそのプロファイルは、類似のインスタンス・タイプの別の DB2 ユニバーサル・データベース (UDB) ワークステーション・インスタンスでインポートすることができます。

このユーティリティーは、接続構成情報を、構成プロファイルというファイルにエクスポートします。これは、指定されるエクスポート・オプションの要件を満たすのに必要な構成情報をすべてパッケージする、非対話式ユーティリティーです。エクスポートできる項目は次のとおりです。

- データベース情報 (DCS および ODBC 情報を含む)
- ノード情報
- プロトコル情報
- データベース・マネージャー構成設定
- UDB レジストリー設定
- 共通 ODBC/CLI 設定

このユーティリティーは、特に DB2CCA GUI がインストールされていないワークステーションで、接続構成情報をエクスポートする場合や、複数の同様のリモート UDB クライアントがインストール、構成、および維持される状況において役立ちます。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*

コマンド構文

```
▶▶ db2cfexp filename [TEMPLATE | BACKUP | MAINTAIN] ▶▶
```

コマンド・パラメーター

filename

ターゲット・エクスポート・ファイルの完全修飾名を指定します。このファイルは、構成プロファイルと呼ばれます。

TEMPLATE

同じインスタンス・タイプの他のインスタンス用のテンプレートとして使用される、構成プロファイルを作成します。プロファイルには次のことに関する情報が含まれます。

- 関連 ODBC および DCS 情報を含めたすべてのデータベース
- エクスポートされるデータベースに関連したすべてのノード
- 共通 ODBC/CLI 設定
- データベース・マネージャー構成の共通クライアント設定
- UDB レジストリーの共通クライアント設定

BACKUP

ローカル・バックアップの目的で、UDB インスタンスの構成プロファイルを作成します。このプロファイルには、このローカル・インスタンスのみに関係のある、特定の性質の情報も含め、インスタンス構成情報がすべて含まれています。プロファイルには次のことに関する情報が含まれます。

- 関連 ODBC および DCS 情報を含めたすべてのデータベース
- エクスポートされるデータベースに関連したすべてのノード
- 共通 ODBC/CLI 設定
- データベース・マネージャー構成のすべての設定
- UDB レジストリーのすべての設定
- すべてのプロトコル情報

MAINTAIN

他のインスタンスを維持または更新するために、データベースおよびノードに関連した情報だけを含む構成プロファイルを作成します。

db2cfimp - 接続構成インポート・ツール

接続構成情報を、構成プロファイルというファイルからインポートします。これは、構成プロファイル中で見つかるすべての情報をインポートしようとする、非対話式ユーティリティーです。

構成プロファイルには、次のような接続項目が含まれる場合があります。

- データベース情報 (DB2 コネクトおよび ODBC 情報を含む)
- ノード情報
- プロトコル情報
- データベース・マネージャー構成設定
- ユニバーサル・データベース (UDB) レジストリー設定
- 共通 ODBC/CLI 設定

このユーティリティーを使用すると、前に構成された同様の別のインスタンスから、接続情報を複製することが可能になります。このユーティリティーは、複数の同様のリモート UDB クライアントが、インストール、構成、および維持される状況において、DB2CCA GUI ツールがないワークステーションで特に役立ちます。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*

コマンド構文

▶—db2cfimp—*filename*————▶

コマンド・パラメーター

filename

インポートされる構成プロファイルの完全修飾名を指定します。有効なインポート構成プロファイルは、DB2 UDB または DB2 コネクトの接続構成エクスポート方式で作成されたプロファイル、またはサーバー・アクセス・プロファイルです。

db2cidmg - リモート・データベース移行

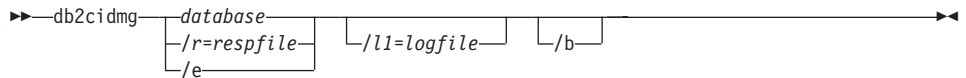
構成、インストール、および配布 (CID) アーキテクチャー環境で、リモート操作による自動移行をサポートします。

許可

以下のどれかが必要です。

- *sysadm*
- *dbadm*

コマンド構文



コマンド・パラメーター

database

移行するデータベースの別名を指定します。この別名が指定されていない場合、プログラムを呼び出すためには、応答ファイルまたは /e が指定されていなければなりません。データベースの別名は宛先ワークステーション上でカタログ作成しておかなければならないので、注意してください。ただし、別名はローカル・データベースのもでもリモート・データベースのもでも構いません。

- /r** CID 移行に使用する応答ファイルを指定します。応答ファイルとは、移行するデータベースのリストが入っている ASCII ファイルのことです。これが指定されていない場合、プログラムを呼び出すためには、データベース別名または /e が指定されていなければなりません。
- /e** システム・データベース・ディレクトリーでカタログ作成されている単一データベースをすべて移行します。 /e を指定しない場合は、データベース別名か応答ファイルを指定しなければなりません。
- /l1** 移行作業が完了したら、リモート・ワークステーションからのエラー・ログ情報を複写することが可能なファイルのパス名を指定します。応答ファイルに複数のデータベースを指定する場合、各データベースの移行ログ情報は、ファイルの最後に追加されます。 /l1 が指定されているかどうかにかかわらず、 DB2CIDMG.LOG という名前のログ・ファイルが生成され、データベースの移行を実行したワークステーションのファイル・システムに保持されます。

db2cidmg - リモート・データベース移行

- /b** データベースにあるパッケージはすべて、移行が完了すると再バインドされます。

db2ckbkp - バックアップの検査

このユーティリティを使用すると、バックアップ・イメージの保全性をテストして、それが復元可能かどうかを判別することができます。また、バックアップ・ヘッダーに保管されているメタ・データを表示して、特定のバックアップ・イメージについての情報を判別するために使用することもできます。イメージの 1 つまたは複数の部分を検査できます。

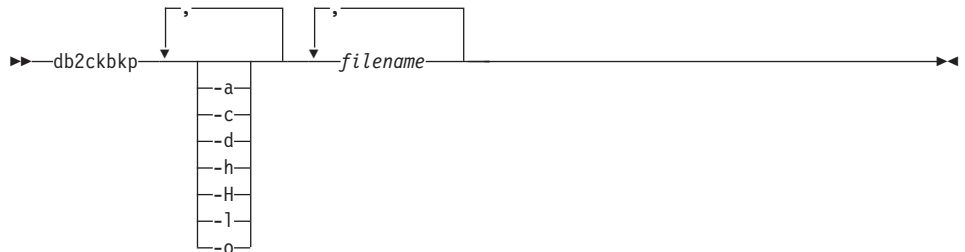
許可

このユーティリティにはすべてのユーザーがアクセス可能ですが、イメージ・バックアップに対してこのユーティリティを実行するには、それらの読み取り許可がなければなりません。

必須接続

なし

コマンド構文



コマンド・パラメーター

- a 使用可能なすべての情報を表示します。
- c チェックビットおよびチェックサムの結果を表示します。
- d DMS 表スペース・データ・ページのヘッダーからの情報を表示します。
- h メディア・ヘッダー情報を表示します。これには、復元ユーティリティで要求されるイメージの名前またはパスも含まれます。
- H メディア・ヘッダー情報のみを表示します。

注:

1. このオプションは、イメージの妥当性検査は実行しません。

db2ckbcp - バックアップの検査

2. このオプションを指定しない場合には、妥当性検査はイメージ全体に対して実行されます。

- l ログ・ファイル・ヘッダー・データを表示します。
- o オブジェクト・ヘッダーからの詳細情報を表示します。

filename

バックアップ・イメージ・ファイルの名前。 1 つ以上のファイルを一度に検査できます。

注:

1. 完全バックアップが複数のオブジェクトで構成されている場合には、同時にすべてのオブジェクトを **db2ckbcp** を使用して妥当性検査する場合にのみ、妥当性検査は正常に実行できます。
2. イメージの複数の部分を検査する場合には、最初のバックアップ・イメージ・オブジェクト (.001) を最初に指定しなければなりません。

例

```
db2ckbcp SAMPLE.0.krodger.NODE0000.CATN0000.19990817150714.*
[1] Buffers processed: ##
[2] Buffers processed: ##
[3] Buffers processed: ##
Image Verification Complete - successful.
db2ckbcp -h SAMPLE2.0.krodger.NODE0000.CATN0000.19990818122909.001
=====
MEDIA HEADER REACHED:
=====
Server Database Name          -- SAMPLE2
Server Database Alias        -- SAMPLE2
Client Database Alias        -- SAMPLE2
Timestamp                     -- 19990818122909
Node                           -- 0
Instance                      -- krodger
Sequence Number              -- 1
Release ID                   -- 900
Database Seed                 -- 65E0B395
DB Comment's Codepage (Volume) -- 0
DB Comment (Volume)          --
DB Comment's Codepage (System) -- 0
DB Comment (System)          --
Authentication Value         -- 255
Backup Mode                   -- 0
Backup Type                   -- 0
Backup Gran.                  -- 0
Status Flags                  -- 11
System Cats inc               -- 1
Catalog Node Number          -- 0
DB Codeset                    -- ISO8859-1
```

```
DB Territory                --
Backup Buffer Size          -- 4194304
Number of Sessions         -- 1
Platform                   -- 0
The proper image file name would be:
SAMPLE2.0.krodger.NODE0000.CATN0000.19990818122909.001
```

```
[1] Buffers processed: ####
Image Verification Complete - successful.
```

使用上の注意

複数のセッションを使用してバックアップを作成した場合には、**db2ckbkp** は同時にすべてのファイルを検査できます。順序番号 001 のセッションが、最初に指定されるファイルであることを確認してください。

このユーティリティーは、テープ上のバックアップ・イメージも検査できます。これは、復元操作の場合のようにテープを準備し、テープ装置名を指定してユーティリティーを実行することにより行えます。たとえば、UNIX ベースのシステムでは以下のようにします。

```
db2ckbkp -h /dev/rmt0
```

Windows NT では以下のようにします。

```
db2ckbkp -d ¥$.¥tape1
```

バックアップが TSM に常駐する場合には、5ページの『db2adutl - TSM アーカイブ・イメージによる作業』を参照してください。

db2ckmig - データベース事前移行ツール

db2ckmig - データベース事前移行ツール

データベースが移行可能であることを検査します。このツールの使用法の詳細については、概説およびインストール 資料のいずれかを参照してください。

許可

sysadm

必須接続

なし

コマンド構文

```
db2ckmig -database- -l filename -u userid -p password
```

コマンド・パラメーター

database

走査するデータベース名の別名を指定します。

- e 走査対象のローカルにカタログ作成されたデータベースをすべて指定します。
- l データベースの走査で生成されたエラーおよび警告のリストを保持するログ・ファイルを指定します。
- u システム管理者のユーザー ID を指定します。
- p システム管理者のユーザー ID のパスワードを指定します。

使用上の注意

データベースの状態の検査は、以下の手順で行います。

1. インスタンス所有者としてログオンする。
2. **db2ckmig** コマンドを発行する。
3. ログ・ファイルをチェックする。エラーがあれば、概説およびインストール資料のいずれかを参照して、推奨されている修正処置を調べてください。

注: ログ・ファイルは、**db2ckmig** コマンドの実行時に起きたエラーを表示します。ログがヌルであることをチェックしてから、移行プロセスを継続してください。

db2cli - DB2 対話機能 CLI

CLI で、設計およびプロトタイピングのための対話式コール・レベル・インターフェイス環境を開始します。これはデータベース・インスタンス所有者のホーム・ディレクトリーの `sqllib/samples/cli/` サブディレクトリーにあります。

許可

なし

必須接続

なし

コマンド構文

▶—db2cli—▶

コマンド・パラメーター

なし

使用上の注意

DB2 対話機能 CLI は、CLI 関数呼び出しの設計、プロトタイプ作成、テストに使用できる一連のコマンドからなっています。このツールはプログラマーの便宜を図ったテスト用ツールで、ツールのパフォーマンスについては IBM の保証はありません。DB2 対話機能 CLI はエンド・ユーザーを対象にしていないので、広範囲にわたるエラー・チェック機能はありません。

2 種類のコマンドがサポートされています。

CLI コマンド

IBM CLI がサポートする各関数呼び出しに対応している (同じ名前の) コマンド。

サポート・コマンド

CLI 関数には等価なものがないコマンド。

コマンドは対話式で発行することも、またはファイル内から発行することもできます。同様に、コマンド出力は端末に表示することも、ファイルに書き込むこともできます。CLI コマンド・ドライバーの便利な機能は、セッション中

db2cli - DB2 対話機能 CLI

に入力されたコマンドをすべてキャプチャーし、それをファイルに書き込み、それによってコマンド・スクリプトを作成することです。このスクリプトは後で再実行することができます。

このユーティリティーの詳細については、データベース・インスタンス所有者のホーム・ディレクトリーの `sqllib/samples/cli/` サブディレクトリーにある `intcli.doc` ファイルを参照してください。

db2cmd - DB2 コマンド・ウィンドウのオープン

CLP 可能 DB2 ウィンドウをオープンし、DB2 コマンド行環境を初期化します。このコマンドを発行することは、「DB2 コマンド・ウィンドウ (DB2 Command Window)」アイコンをクリックすることと同じです。

このコマンドは、Windows NT、Windows 95、Windows 98、および Windows 2000 でのみ使用可能です。

許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

- c コマンドを実行してから終了します。たとえば、"db2cmd /c dir" というコマンドを使用すると、"dir" コマンドがコマンド・ウィンドウに呼び出されてから、コマンド・ウィンドウがクローズします。
- w cmd.exe プロセスが終了するまで待機します。たとえば、"db2cmd /c /w dir" というコマンドを使用すると、"dir" コマンドが呼び出され、コマンド・ウィンドウがクローズするまで、db2cmd.exe は終了しません。
- i 同じコンソールを共用し、ファイル・ハンドルを継承して、コマンド・ウィンドウを実行します。たとえば、"db2cmd /c /w /i db2 get dbm cfg > myoutput" を使うと、cmd.exe が起動して db2 コマンドを実行し、完了を待機します。新しいコンソールは割り当てられず、stdout がファイル "myoutput" にパイプ接続されます。
- t "DB2 CLP" をコマンド・ウィンドウのタイトルに使用する代わりに、

db2cmd - DB2 コマンド・ウィンドウのオープン

起動ウィンドウからタイトルを継承します。これは、たとえば "db2cmd /t" を起動する異なるタイトルでアイコンを設定したい場合などに役立ちます。

注: すべてのスイッチは、コマンドが実行される前に表示されるはずですが。たとえば、db2cmd /t db2 などです。

使用上の注意

DB21061E (「コマンド行環境が初期化されていない。」) が、CLP で使用可能になった DB2 ウィンドウを表示する際に戻される場合、または Windows 95 か Windows 98 で CLP コマンドを実行する場合、オペレーティング・システムの環境スペースが足りない可能性があります。config.sys ファイルで SHELL 環境セットアップ・パラメーターを検査し、状況に応じて値を増やしてください。たとえば、次のようにします。

```
SHELL=C:¥COMMAND.COM C:¥ /P /E:32768
```


db2dclgn - 宣言生成プログラム

指定されたデータベース表に宣言を生成し、文書中でそれらの宣言を検索する必要を省きます。生成された宣言は、必要に応じて変更できます。サポートされるホスト言語は C/C++、COBOL、JAVA、および FORTRAN です。

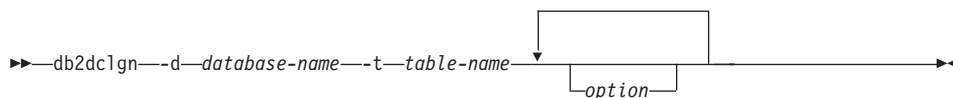
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

-d database-name

接続の確立先のデータベースの名前を指定します。

-t table-name

宣言を生成するために列情報が検索される表の名前を指定します。

option

以下のどれかです (複数の場合もある)。

-a action

宣言が追加されるか、置換されるかを指定します。有効な値は ADD および REPLACE です。省略時値は ADD です。

-b lob-var-type

LOB 列に生成される変数のタイプを指定します。有効な値は以下のとおりです。

LOB (省略時値)

たとえば、C では SQL TYPE は CLOB(5K) x です。

LOCATOR

たとえば、C では SQL TYPE は CLOB_LOCATOR x です。

FILE たとえば、C では SQL TYPE は CLOB_FILE x です。

- c 接頭部 (-n) の指定時に、列名が、フィールド名で接尾部として使用されるかどうかを指定します。接頭部が指定されない場合、このオプションは無視されます。省略時の動作は、列名を接尾部としては使用せず、代わりに 1 で始まる列番号を使用します。
- i 標識変数が生成されるかどうか指定します。ホスト構造は C および COBOL でサポートされるので、列の数に等しいサイズの標識表が生成されますが、JAVA および FORTRAN の場合は、個々の標識変数が各列ごとに生成されます。標識表および変数の名前は、それぞれ表名および列名と同じで、"IND-" (COBOL の場合) または "ind_" (その他の言語の場合) という接頭部が付きます。省略時の動作は、標識変数を生成しません。

-l language

宣言が生成されるホスト言語を指定します。有効な値は C、COBOL、JAVA、および FORTRAN です。省略時の動作は、C 宣言を生成することで、C++ にも有効です。

-n name

それぞれのフィールド名に接頭部を指定します。接頭部は、-c オプションが使用される場合に指定する必要があります。指定されないと、列名がフィールド名として使用されます。

-o output-file

宣言用の出力ファイルの名前を指定します。省略時の動作は、生成されたホスト言語を反映した拡張子を使った、基本ファイル名として表名を使用します。

- .h (C の場合)
- .cbl (COBOL の場合)
- .java (JAVA の場合)
- .f (FORTRAN (UNIX) の場合)
- .for (FORTRAN (INTEL) の場合)

-p password

データベースへの接続に使用するパスワードを指定します。ユーザー ID を指定する場合に指定する必要があります。省略時の動作では、接続の確立時にパスワードを提供しません。

-r remarks

列の注釈が使用可能である場合、宣言内のコメントとして使用される、フィールドのより詳細な記述を提供するかどうかを指定します。

-s structure-name

宣言内のすべてのフィールドをグループ化するために生成される構造名を指定します。省略時の動作では、修飾なしの表名を使用します。

-u userid

データベースへの接続に使用するユーザー ID を指定します。パスワードを指定する場合に指定する必要があります。省略時の動作では、接続の確立時にユーザー ID を提供しません。

-v

ユーティリティの状況 (たとえば接続状況) が表示されるかどうかを指定します。省略時の動作では、エラー・メッセージのみを表示します。

-w DBCS-var-type

sqldbcchar または wchar_t が、C で GRAPHIC/VARGRAPHIC/DBCLOB 列に使用されるかどうかを指定します。

-y DBCS-symbol

G または N が、COBOL で DBCS シンボルとして使用されるかどうかを指定します。

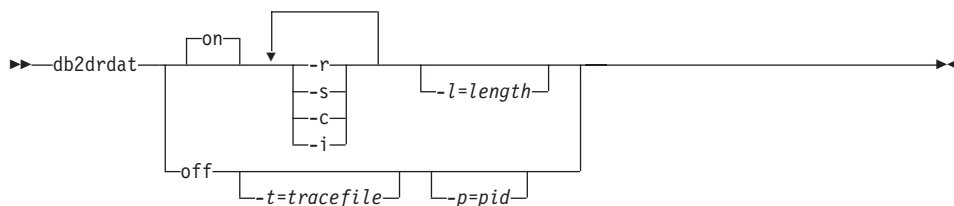
db2drdat - DRDA トレース

DB2 UDB DRDA アプリケーション・リクエスター (AR) および DRDA アプリケーション・サーバー (AS) 間で交換された DRDA データ・ストリームをキャプチャーできます。このツールは、アプリケーションの実行に必要な送信および受信の回数を判別することによって、問題判別でよく使用されますが、クライアント / サーバー環境でのパフォーマンス調整にも使用することができます。

許可

なし

コマンド構文



コマンド・パラメーター

- on** AS トレース事象 (指定しない場合はすべて) をオンにします。
- off** AS トレース事象をオフにします。
- r** DRDA AR から受信した DRDA 要求をトレースします。
- s** DRDA AR に送信された DRDA 応答をトレースします。
- c** ホスト・システムの DRDA サーバーから受信した SQLCA をトレースします。これは、様式化して読みやすくした非ヌル SQLCA です。
- i** トレース情報にタイム・スタンプを含めます。
- l** トレース情報を格納するために使用されるバッファのサイズを指定します。
- p** このプロセスについてのみ事象をトレースします。 **-p** を指定しない場合、サーバー上の着信 DRDA 接続をもつエージェントがすべてトレースされます。

注: トレースする *pid* は、364ページの『LIST APPLICATIONS』によって戻される *agent* フィールドにあります。

- t** トレースの宛先を指定します。ファイル名で、完全なパス名が指定されていない場合、脱落情報は現在パスから取られます。

注: *tracefile* が指定されていない場合、メッセージは現行ディレクトリ内の *db2drdat.dmp* に送られます。

使用上の注意

db2drdat が活動中には、**db2trc** コマンドを発行しないでください。
(**db2trc** コマンドの詳細については、*問題判別の手引き* を参照してください。)

db2drdat は、以下の情報を *tracefile* に書き込みます。

1. -r
 - DRDA 要求のタイプ
 - 受信バッファ
2. -s
 - DRDA 応答 / オブジェクトのタイプ
 - 送信バッファ
3. CPI-C エラー情報
 - 重大度
 - 使用したプロトコル
 - 使用した API
 - ローカル LU 名
 - 失敗した CPI-C 機能
 - CPI-C 戻りコード

コマンドは終了コードを戻します。ゼロ値はコマンドが正常に完了したことを示します。非ゼロ値はコマンドが正常に完了しなかったことを示します。

注: **db2drdat** がすでに存在するファイルに出力を送信する場合、ファイルの許可で古いファイルの消去が禁止されているのではない限り、古いファイルは消去されます。古いファイルの消去が禁止されている場合は、オペレーティング・システムがエラーを返します。

db2empfa - 複数ページ・ファイル割り振りの使用可能化

データベースの複数ページ・ファイル割り振りを使用可能にします。SMS 表スペースでの複数ページ・ファイル割り振りを使用可能にすると、ディスク・スペースは、一度に 1 ページではなく、1 エクステントに割り振られます。

効力範囲

このコマンドは、それが実行されたノードに対してだけ影響を与えます。

許可

sysadm

必須接続

なし。このコマンドは、データベース接続を確立します。

コマンド構文

▶▶db2empfa—*database-alias*————▶▶

コマンド・パラメーター

database-alias

複数ページ・ファイル割り振りを使用可能にするデータベースの別名を指定します。

使用上の注意

このユーティリティーは以下の処理を行います。

- (適用可能な) ノードのデータベース区画に排他モードで接続する
- すべての SMS 表スペースでは、ヌル・ページを割り振り、1 エクステントより大きなデータおよび索引ファイルすべてで最後のエクステントを埋め込む
- データベース構成パラメーター *multipage_alloc* の値を YES に変更する
- 切断する

db2empfa は、ノードのデータベース区画に排他モードで接続するので、カタログ・ノードまたは他のどのノードでも同時に実行することはできません。

db2eva - イベント・アナライザー

イベント・アナライザーを開始することにより、DB2 事象モニターが生成しファイルに送ったパフォーマンス・データをトレースできるようにします。事象モニターについての詳細は、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。

許可

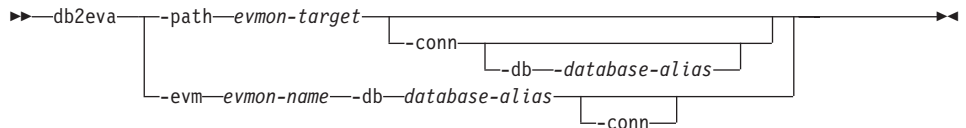
なし。ただし、データベースに接続しており、かつカタログから選択している場合 (-evm、-db、および -conn) には、以下のいずれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

必須接続

なし

コマンド構文



コマンド・パラメーター

-path evmon-target

事象モニター・トレース・ファイルを含むディレクトリーを指定します。

-conn **db2eva** に、**-db** で指定したデータベースへの接続を保守するよう要求します。**-db** を使用していない場合は、事象モニター・トレース・ヘッダーに指定されたデータベースへの接続を保守するよう要求します。接続を保守することにより、イベント・アナライザーは、トレース・ファイルには含まれない情報（たとえば、静的 SQL のテキスト）を獲得できるようになります。**-conn** を指定した場合、ステートメント事象レコードにはパッケージ作成者、パッケージ、およびセクション番号が入ります。**db2eva** はデータベース・システム・カタログ (sysibm.sysstmt) からテキストを検索できます。

db2eva - イベント・アナライザー

-db database-alias

事象モニター用に定義したデータベースの名前を指定します。 `-path` を指定してある場合、事象モニター・トレース・ヘッダーにあるデータベース名は一時変更されます。

-evm evmon-name

トレースが分析される事象モニターの名前です。

使用上の注意

必須の接続はありませんが、`-conn`、または `-evm` と `-db` オプションを使用している場合、**db2eva** はデータベースへの接続を試行します。データベースにアクセスでき、かつ適切な許可がある場合には、静的ステートメントの SQL テキストも表示できます。しかし、必要なアクセス権や権限がない場合、動的ステートメントのテキストしか表示できません。

以下の 2 通りの方法で、事象モニター・トレースを読み取ることができます。

1. トレース・ファイルのあるディレクトリーを指定する (`-path` オプションを使用)。こうすると、サーバーからトレース・ファイルを移動させ、それらのファイルをローカルに分析できるようになります。事象モニターが除去されている場合でも、この方法は使用できます。
2. データベース名と事象モニター名を指定することにより、トレース・ファイルの自動位置指定を行う。こうすると、イベント・アナライザーはデータベースに接続し、`select target from sysibm.syseventmonitors` を発行することによって、事象モニターがトレース・ファイルを書き込むディレクトリーを位置指定します。`-conn` を指定しなかった場合、その後接続は解放されます。事象モニターが除去されている場合には、この方法は使用できません。

注: イベント・アナライザーは、活動中の事象モニターが生成したデータを分析するのに使用できます。しかし、事象モニターはそれらのデータをディスクに書き込む前にバッファーに入れるので、一部の情報が失われてしまう可能性があります。事象モニターをオフにして、強制的にバッファーをフラッシュしてください。

db2evmon - 事象モニター生産性向上ツール

事象モニター・ファイルと名前付きパイプをフォーマットし、それを標準出力に書き込みます。

許可

なし。ただし、データベースに接続している場合には (-evm, -db,)、以下のいずれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

必須接続

なし

コマンド構文

```
▶▶ db2evmon [ -db database-alias -evm event-monitor-name ] [ -path event-monitor-target ] ▶▶
```

コマンド・パラメーター

-db database-alias

表示するデータのあるデータベースを指定します。このパラメーターには、大文字小文字の区別があります。

-evm event-monitor-name

事象モニターの 1 パート名です。普通の、または区切り SQL 識別子です。このパラメーターには、大文字小文字の区別があります。

-path event-monitor-target

事象モニター・トレース・ファイルを含むディレクトリーを指定します。

使用上の注意

データがファイルに書き込まれている場合、このツールは標準出力を使用した表示のためにファイルをフォーマットします。この場合、最初にモニターがオ

db2evmon - 事象モニター生産性向上ツール

ンになり、次にこのツールによってファイル中の事象データが表示されます。このツールを実行した後にファイルに書き込まれたデータをすべて表示させるには、 **db2evmon** を再発行します。

データがパイプに書き込まれている場合、事象が起きた時点で、このツールは標準出力を使用した表示のために出力をフォーマットします。この場合、ツールが開始した後 に、モニターがオンになります。

db2exfmt - Explain 表フォーマット・ツール

Explain 表の内容をフォーマットします。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

以下も参照

50ページの『db2expln - DB2 SQL Explain ツール』

db2expln - DB2 SQL Explain ツール

DB2 共通サーバー・システム・カタログに保管されているパッケージにある、静的 SQL ステートメント用のアクセス・プランの種類を記述します。データベース名、パッケージ名、パッケージ作成者、およびセクション番号を指定すると、このツールはこれらのカタログ内の情報を解釈して記述します。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

以下も参照

49ページの『[db2exfmt - Explain 表フォーマット・ツール](#)』

db2flsn - ログ順序番号の検出

指定されたログ順序番号 (LSN) で識別されるログ・レコードを含むファイルの名前を戻します。

許可

なし

コマンド構文

```
▶▶ db2flsn  ▶▶
```

コマンド・パラメーター

-q ログ・ファイル名だけが印刷されます。エラー・メッセージや警告メッセージは印刷されず、状況は戻りコードを介してのみ判別できます。有効なエラー・コードは以下のとおりです。

- -100 無効な入力
- -101 LFH ファイルをオープンできない
- -102 LFH ファイルの読み取りに失敗した
- -103 無効な LFH
- -104 データベースが回復可能
- -105 LSN が大きすぎる
- -500 論理エラー

他の有効な戻りコードは以下のとおりです。

- 0 正常な実行
- 99 警告: 結果は、分かっている最後のログ・ファイル・サイズに基づいている

input_LSN

ストリング付きの内部 (6 バイト) 16 進数値を表す 12 バイトのストリング。

例

```
db2flsn 000000BF0030
Given LSN is contained in log file S0000002.LOG
db2flsn -q 000000BF0030
S0000002.LOG
db2flsn 000000BE0030
```

db2flsn - ログ順序番号の検出

```
Warning: the result is based on the last known log file size.  
The last known log file size is 23 4K pages starting from log extent 2.  
Given LSN is contained in log file S0000001.LOG  
db2flsn -q 000000BE0030  
S0000001.LOG
```

使用上の注意

ログ・ヘッダー制御ファイル `sqllogct1.lfh` が現行ディレクトリになければなりません。このファイルはデータベース・ディレクトリにあるので、データベース・ディレクトリからこのツールを実行するか、このツールが実行されるディレクトリに制御ファイルをコピーすることができます。

このツールは、`logfilsiz` データベース構成パラメーターを使用します。DB2は、このパラメーターの最新の3つの値と、各 `logfilsiz` 値によって作成された最初のログ・ファイルを記録します。このため、`logfilsiz` が変更されても、ツールは正しく動作することができます。指定された LSN の日付が最新の `logfilsiz` 値の日付よりも前の場合、ツールはこの値を使用し、警告を戻します。このツールは、UDB バージョン 5.2 より前のデータベース・マネージャーでも使用できます。その場合、正しい結果 (`logfilsiz` の値が変更されない場合に得られる) についても警告が戻されます。

このツールは、回復可能データベースでのみ使用することができます。データベースが回復可能なのは、`logretain` または `userexit` をオンにして構成されている場合です。

以下も参照

管理 API 解説書の『`sqlurlog` - ログの非同期読み取り』API

db2fs - First Steps

First Steps GUI を立ち上げます。これには、ユーザーが DB2 の学習および使用を始めるにあたって必要な機能へのリンクがあります。

UNIX ベースのシステムでは、db2fs は sqllib/bin ディレクトリーにあります。OS/2 では、db2fs.cmd は \$DB2PATH¥bin ディレクトリーにあり、Windows オペレーティング・システムでは、db2fs.bat は \$DB2PATH¥bin ディレクトリーにあります。

許可

sysadm

コマンド構文

▶▶—db2fs—▶▶

コマンド・パラメーター

なし

db2gov - DB2 管理プログラム

データベースに対して実行しているアプリケーションの振る舞いをモニターし変更します。省略時では、デーモンは各論理ノードで開始されます。しかし、フロントエンド・ユーティリティーを使用して、特定のノードで単一デーモンを開始し、そのノードのデータベース区画に対する活動をモニターすることができます。

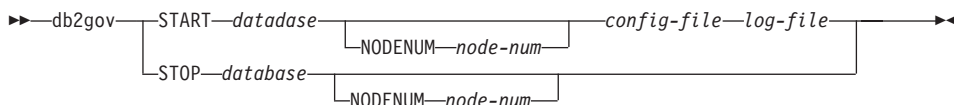
このコマンドの詳細については、[管理の手引き](#) を参照してください。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*

コマンド構文



コマンド・パラメーター

START database

管理プログラム・デーモンを開始して、指定されたデータベースをモニターします。データベース名またはデータベース別名のいずれかを指定できます。指定する名前は、管理プログラム構成ファイルに指定する名前と同じでなければなりません。

注: モニターされる各データベースにつき 1 つのデーモンが実行されます。区分データベース環境では、各データベース区画につき 1 つのデーモンが実行されます。複数のデータベースに対して管理プログラムが実行されている場合には、データベース・サーバーでも複数のデーモンが実行されます。

NODENUM node-num

管理プログラム・デーモンを開始または停止するデータベース区画を指定します。この番号は、ノード構成ファイルで指定した番号と同じでなければなりません。

config-file

データベースをモニターする際に使用する構成ファイルを指定します。

構成ファイルの省略時の位置は、sql1lib ディレクトリーです。指定したファイルがこのディレクトリーにない場合、フロントエンドは、指定したこの名前をファイルの完全名であるとみなします。

log-file

管理プログラムがログ記録を書き込むファイルの基本名を指定します。ログ・ファイルは、sql1lib ディレクトリーのログ・サブディレクトリーに保管されます。管理プログラムが実行されているデータベース区画の数は、自動的にログ・ファイル名に追加されます。たとえば、mylog.0, mylog.1, mylog.2 となります。

STOP database

指定したデータベースをモニターしている管理プログラム・デーモンを停止します。区分データベース環境では、フロントエンド・ユーティリティーは、ノード構成ファイルを読み取ることによって、すべてのデータベース区画の管理プログラムを停止します。

db2govlg - DB2 管理プログラム・ログ照会

指定したタイプのレコードを管理プログラム・ログ・ファイルから抽出します (54ページの『db2gov - DB2 管理プログラム』を参照)。DB2 管理プログラムは、データベースに対して実行しているアプリケーションの振る舞いをモニターし変更します。

許可

なし

コマンド構文

```
▶ db2govlg log-file [nodenum node-num] [rectype record-type] ▶
```

コマンド・パラメーター

log-file

照会するログ・ファイル (複数可) のベース名。

nodenum node-num

管理プログラムを実行しているノードの番号。

rectype record-type

照会するレコードのタイプです。次のレコード・タイプが有効です。

- START
- FORCE
- NICE
- ERROR
- WARNING
- READCFG
- STOP
- ACCOUNT

以下も参照

54ページの『db2gov - DB2 管理プログラム』

db2icrt - インスタンスの作成

DB2 インスタンスを作成します。

UNIX ベースのシステムでは、このユーティリティーは DB2DIR/instance ディレクトリにあります。ここで、DB2DIR の部分は AIX では /usr/lpp/db2_07_01 になり、Linux では /usr/IBMdb2/V7.1 になり、他のすべての UNIX ベースのシステムでは /opt/IBMdb2/V7.1 になります。OS/2 または Windows オペレーティング・システムでは、このツールは %sqllib%bin サブディレクトリにあります。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

許可

UNIX ベースのシステムでは root でアクセスし、Windows または OS/2 オペレーティング・システムではローカル管理者でアクセスします。

コマンド構文

UNIX ベースのシステムの場合

```

▶▶ db2icrt [-h] [-d] [-a AuthType] [-p PortName]
           [-s InstType] [-u FencedID] InstName

```

Windows オペレーティング・システムと OS/2 の場合

```

▶▶ db2icrt [-s InstType] [-u UserName, Password] [-p InstProfPath]
           [-c ClusterName] [-h HostName] [-r PortRange] InstName

```

コマンド・パラメーター

UNIX ベースのシステムの場合

-h または **-?**

使用情報を表示します。

-d デバッグ・モードをオンにします。

db2icrt - インスタンスの作成

-a AuthType

インスタンスの認証タイプ (SERVER、CLIENT、または DCS) を指定します。省略時値は SERVER です。

-p PortName

インスタンスが使用するポート名または番号を指定します。

-s InstType

作成するインスタンスのタイプを指定します (EEE、EE、または CLIENT)。

-u Fenced ID

隔離したユーザー定義関数および隔離したストアード・プロシージャを実行するユーザー ID の名前を指定します。

InstName

インスタンスの名前を指定します。

Windows オペレーティング・システムと OS/2 の場合

-s InstType

作成するインスタンスのタイプを指定します。有効な値は以下のとおりです。

Client クライアントのインスタンスを作成するために使用します。

Standalone

ローカル・クライアントでデータベース・サーバーのインスタンスを作成するために使用します。

Satellite

サテライト・データベース・サーバーのインスタンスを作成するために使用します。

EE ローカルおよびリモート・クライアントでデータベース・サーバーのインスタンスを作成するために使用します。

EEE 区分データベース・サーバーのインスタンスを作成するために使用します。

-u Username, Password

DB2 サービスのアカウント名およびパスワードを指定します。区分データベース・インスタンスを作成する際には、このオプションは必須です。

-p InstProfPath

インスタンス・プロファイル・パスを指定します。区分データベース・インスタンスを作成する際には、このオプションは必須です。

-h HostName

現行のマシンに対して複数の省略時の TCP/IP ホスト名がある場合、それらを指定変更します。TCP/IP ホスト名は、省略時ノード (node 0) を作成する際に使用されます。このオプションは、区分データベース・インスタンスに対してのみ有効です。

-c ClusterName

MSCS クラスタ名を指定します。このオプションは、MSCS をサポートする DB2 インスタンスを作成するために指定します。

-r PortRange

MPP モードで実行する場合に、区分データベース・インスタンスによって使用される TCP/IP ポートの範囲を指定します。このオプションを指定した場合、ローカル・マシンのサービス・ファイルは、以下の項目で更新されます。

```
DB2_InstName      baseport/tcp
DB2_InstName_END  endport/tcp
```

InstName

インスタンスの名前を指定します。

db2idrop - インスタンスの除去

db2idrop - インスタンスの除去

57ページの『db2icrt - インスタンスの作成』によって作成された DB2 インスタンスを除去します。インスタンスのリストからインスタンス項目を除去します。

UNIX ベースのシステムでは、このユーティリティーは DB2DIR/instance ディレクトリーにあります。ここで、DB2DIR の部分は AIX では /usr/lpp/db2_07_01 になり、Linux では /usr/IBMdb2/V7.1 になり、他のすべての UNIX ベースのシステムでは /opt/IBMdb2/V7.1 になります。OS/2 または Windows オペレーティング・システムでは、このツールは %sqllib%bin サブディレクトリーにあります。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

許可

UNIX ベースのシステムでは root でアクセスし、Windows または OS/2 オペレーティング・システムではローカル管理者でアクセスします。

コマンド構文

UNIX ベースのシステムの場合

```
▶▶ db2idrop [-h|-?] InstName ▶▶
```

Windows オペレーティング・システムと OS/2 の場合

```
▶▶ db2idrop [-f] InstName ▶▶
```

コマンド・パラメーター

UNIX ベースのシステムの場合

-h または **-?**

使用情報を表示します。

InstName

インスタンスの名前を指定します。

Windows オペレーティング・システムと OS/2 の場合

- f** 強制アプリケーション・フラグを指定します。このフラグを指定すると、このインスタンスを使用しているすべてのアプリケーションが強制的に終了させられます。

InstName

インスタンスの名前を指定します。

db2ilist - インスタンスのリスト

db2ilist - インスタンスのリスト

システムで使用可能なインスタンスをすべてリストします。

UNIX ベースのシステムでは、このユーティリティーは DB2DIR/instance ディレクトリーにあります。ここで、DB2DIR の部分は AIX では /usr/lpp/db2_07_01 になり、Linux では /usr/IBMd2/V7.1 になり、他のすべての UNIX ベースのシステムでは /opt/IBMd2/V7.1 になります。OS/2 または Windows オペレーティング・システムでは、このツールは %sqllib%bin サブディレクトリーにあります。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

許可

UNIX ベースのシステムでは root でアクセスします。Windows および OS/2 オペレーティング・システムでは、許可は必要ありません。

コマンド構文

▶—db2ilist—▶

コマンド・パラメーター

なし

db2imigr - インスタンスの移行

データベース・マネージャーのインストールの後に、既存のインスタンスを移行します。

UNIX ベースのシステムでは、このユーティリティーは DB2DIR/instance ディレクトリーにあります。ここで、DB2DIR の部分は AIX では /usr/lpp/db2_07_01 になり、Linux では /usr/IBMdb2/V7.1 になり、他のすべての UNIX ベースのシステムでは /opt/IBMdb2/V7.1 になります。

このコマンドの詳細な説明は、概説およびインストール 資料のいずれかを参照してください。

db2iupdt - インスタンスの更新

UNIX ベースのシステムの場合、このコマンドは、新規システム構成の取得を可能にしたり、特定のプロダクト・オプションのインストールまたは除去に関連した関数にアクセスするために、指定された DB2 インスタンスを更新します。

Windows NT および Windows 2000 オペレーティング・システムでは、このコマンドは、区分データベース・システムで使用する単一区画インスタンスを更新します。

UNIX ベースのシステムでは、このユーティリティーは DB2DIR/instance ディレクトリにあります。ここで、DB2DIR の部分は AIX では

/usr/lpp/db2_07_01 になり、Linux では /usr/IBMdb2/V7.1 になり、他のすべての UNIX ベースのシステムでは /opt/IBMdb2/V7.1 になります。

Windows NT および Windows 2000 オペレーティング・システムでは、このツールは %sqllib%bin サブディレクトリにあります。

注: このコマンドは OS/2 では使用できません。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

許可

UNIX ベースのシステムでは root でアクセスし、Windows NT および Windows 2000 オペレーティング・システムではローカル管理者でアクセスします。

コマンド構文

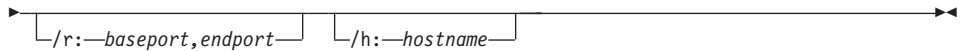
UNIX ベースのシステムの場合

```
db2iupdt [-h] [-d] [-k] [-s] [-a AuthType]
```

```
[-u FencedID] [-e InstName]
```

Windows NT と Windows 2000 の場合

```
db2iupdt InstName /u:username,password [/p:instance profile path]
```



コマンド・パラメーター

UNIX ベースのシステムの場合

-h または **-?**

使用情報を表示します。

-d デバッグ・モードをオンにします。

-k 更新時に現行のインスタンス・タイプが変更されないようにします。

-s 既存の SPM ログ・ディレクトリーを無視します。

-a AuthType

インスタンスの認証タイプ (SERVER、CLIENT、または DCS) を指定します。省略時値は SERVER です。

-u Fenced ID

隔離したユーザー定義関数および隔離したストアード・プロシージャを実行するユーザー ID の名前を指定します。

InstName

インスタンスの名前を指定します。

-e すべてのインスタンスを更新します。

Windows NT と Windows 2000 の場合

InstName

インスタンスの名前を指定します。

/u:username,password

DB2 サービスのアカウント名およびパスワードを指定します。

/p:instance profile path

更新されたインスタンス用の新しいインスタンス・プロファイル・パスを指定します。

/r:baseport,endport

MPP モードで実行する場合に、区分されたデータベース・インスタンスによって使用される TCP/IP ポートの範囲を指定します。このオプションを指定した場合、ローカル・マシンのサービス・ファイルは、以下の項目で更新されます。

```
DB2_InstName      baseport/tcp  
DB2_InstName_END  endport/tcp
```

/h:hostname

現行のマシンに対して複数の省略時の TCP/IP ホスト名がある場合、それらを指定変更します。

db2ldcfg - LDAP 環境の構成

現行のログオン・ユーザー用の Lightweight Directory Access Protocol (LDAP) ユーザー識別名 (DN) およびパスワードを、IBM LDAP クライアントを使用する LDAP 環境で構成します。

許可

none

必須接続

なし

コマンド構文

```
db2ldcfg -u user's Distinguished Name -w password -r
```

コマンド・パラメーター

-u user's Distinguished Name

LDAP ディレクトリーにアクセスする際に使用する LDAP ユーザーの識別名を指定します。

-w password

パスワードを指定します。

-r マシン環境からユーザーの DN およびパスワードを除去します。

使用上の注意

IBM LDAP クライアントを使用する LDAP 環境では、現行のログオン・ユーザー用の省略時 LDAP ユーザーの DN およびパスワードを構成できます。一度構成すると、LDAP ユーザーの DN およびパスワードがこのユーザーの環境に保管され、それらは DB2 が LDAP ディレクトリーにアクセスする際に必ず使用されます。こうすると、LDAP コマンドまたは API を発行する際に、LDAP ユーザーの DN およびパスワードを指定する必要はなくなります。ただし、コマンドまたは API が発行される際に LDAP ユーザーの DN およびパスワードが指定されると、省略時設定は上書きされてしまいます。

このコマンドは、IBM LDAP クライアントを使用する場合にのみ実行できます。Microsoft LDAP クライアントでは、現行のログオン・ユーザーの認証が使用されます。

db2licm - ライセンス管理ツール

コントロール・センターがない場合に基本ライセンス機能を実行します。ローカル・システムにインストールされたライセンスおよびポリシーを追加、除去、リスト、および変更します。

許可

UNIX ベースのシステムでは、root 権限が必要です。OS/2 または Windows オペレーティング・システムでは、許可は必要ありません。

必須接続

なし

コマンド構文

```

▶▶—db2licm—▶▶
  -a—filename—
  -l—
  -p—prod-password— [REGISTERED] [CONCURRENT]
  -r—prod-password—
  -u—prod-password—num-users—
  -n—prod-password—num-processors—
  -e— [HARD] [SOFT]
  -v
  -h—
  -?—

```

コマンド・パラメーター

-a filename

製品のライセンスを追加します。有効なライセンス情報を含むファイル名を指定します。

-l 全ての製品を、使用可能なライセンス情報と共にリストします。

-p prod-password REGISTERED CONCURRENT

システムで使用するライセンス・ポリシー・タイプを更新します。REGISTERED、CONCURRENT、または両方のキーワードを指定できます。

-r prod-password

製品のライセンスを除去します。ライセンスを除去した後、製品は

db2licm - ライセンス管理ツール

"Try & Buy" (お試し) モードで機能します。特定の製品のパスワードを入手するには、 **-l** オプションを指定してコマンドを起動してください。

-u prod-password num-users

顧客が購入したユーザー・ライセンスの数を更新します。ユーザーの数、およびライセンスが購入された製品のパスワードを指定してください。

-n prod-password num-processors

カスタマーに DB2 を使用するライセンスが与えられている処理装置の数を更新します。

-e システム上の制約ポリシーを更新します。有効な値は、HARD および SOFT です。 HARD は、ライセンスなしの要求が許可されないことを指定します。 SOFT は、ライセンスなしの要求がログに記録されるが、制限はされないことを指定します。

-v バージョン情報を表示します。

-h/-? ヘルプ情報を表示します。このオプションを指定すると、他のすべてのオプションは無視され、ヘルプ情報だけが表示されます。

例

```
db2licm -a db2entr.lic
db2licm -p db2udbwe registered concurrent
db2licm -r db2udbee
db2licm -u db2udbwe 10
db2licm -n db2udbee 8
```


db2look - DB2 統計および DDL 抽出ツール

要求された DDL ステートメントを抽出して、テスト・データベース上の実動データベースのデータベース・オブジェクトを再生成します。このツールを使用すると、テスト・データベース内のオブジェクトの統計を複製するために使用される必須 UPDATE ステートメントを生成するだけでなく、更新データベース構成パラメーターと更新データベース・マネージャー構成パラメーター、および **db2set** ステートメントも生成して、テスト・データベースの登録変数および構成パラメーター設定を、実動データベースの設定に適合させることができます。

テスト・システムに実動システムのデータのサブセットを含めておくと、便利なが多くあります。しかし、そのようなテスト・システム用に選択したアクセス・プランが、必ずしも実動システム用に選択したアクセス・プランと同じであるとは限りません。テスト・システム用のカタログ統計と構成パラメーターの両方が、実動システムのものと同じように更新されていなければなりません。このツールを使用すると、アクセス・プランが、実動システムで使用するものと類似しているテスト・データベースを作成することが可能になります。

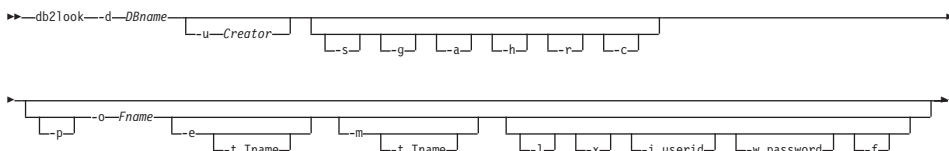
許可

システム・カタログに対する SELECT 特権

必須接続

なし。このコマンドは、データベース接続を確立します。

コマンド構文



コマンド・パラメーター

-d DBname

照会する実動データベースの別名。DBname は、DB2 UDB (UNIX 版、Windows 版、OS/2 版)、または DB2 UDB (OS/390 版) データベースの名前にすることができます。DBname が DB2 UDB (OS/390 版) データベースの場合には、**db2look** ユーティリティーは、

db2look - DB2 統計および DDL 抽出ツール

OS/390 オブジェクト用の DDL および UPDATE 統計ステートメントを抽出します。これらの DDL および UPDATE 統計ステートメントは、DB2 UDB データベースには適用できますが、DB2 (OS/390 版) データベースには適用できません。これは、OS/390 オブジェクトを抽出して、それらを DB2 UDB データベースで再作成しようとするユーザーに役立ちます。

DBname が OS/390 データベースの場合、**db2look** 出力は以下のものに制限されます。

- 表、索引、および視点用の DDL の生成
- 表、列、列配布および索引用の UPDATE 統計ステートメントの生成

-u Creator

作成者 ID。出力をこの作成者 ID があるオブジェクトだけに制限します。オプション **-a** を指定した場合、このパラメーターは無視されます。**-u** と **-a** のどちらも指定しない場合には、環境変数 **USER** が使用されます。

-s ポストスクリプト・ファイルを生成します。

注:

1. このオプションは、すべての LaTeX ファイルと `.tmp` ポストスクリプト・ファイルを除去します。
2. 必要な非 IBM 製のソフトウェアは LaTeX と dvips です。
3. `psfig.tex` ファイルは、LaTeX 入力パスに置いておくことが必要です。

-g 索引の取り出しページ・ペアを示すためにグラフを使用します。

注:

1. このオプションは、LaTeX ファイルだけでなく、`filename.ps` ファイルを生成します。
2. 必要な非 IBM 製のソフトウェアは Gnuplot です。
3. `psfig.tex` ファイルは、LaTeX 入力パスに置いておくことが必要です。

-a このオプションが指定されている場合には、特定の作成者 ID で作成されたオブジェクトだけに出力が制限されることはありません。すべてのユーザーによって作成されたすべてのオブジェクトが対象になります。たとえば、このオプションと **-e** オプションが共に指定される場合、データベース内のすべてのオブジェクト用の DDL ステートメン

トが抽出されます。このオプションと `-m` オプションが共に指定される場合、データベース内のすべてのユーザー作成表および索引用の UPDATE 統計ステートメントが抽出されます。

注: `-u` と `-a` のどちらも指定しない場合には、環境変数 **USER** が使用されます。UNIX ベースのシステムでは、この変数を明示的に設定する必要はありません。しかし Windows NT の場合、**USER** 環境変数に省略時値がありません。このプラットフォームでは、**SYSTEM** 変数の中のユーザー変数を設定するか、または `set USER=<username>` をセッションに発行する必要があります。

-h ヘルプ情報を表示します。このオプションを指定すると、他のすべてのオプションは無視され、ヘルプ情報だけが表示されます。

-r このオプションと `-m` オプションを共に指定する場合には、**db2look** は `RUNSTATS` コマンドを生成しません。省略時アクションでは、`RUNSTATS` コマンドを生成します。`-m` オプションを指定しない場合、`-r` オプションは無視されます。

-c このオプションを `-m` オプションと共に指定する場合には、**db2look** は `COMMIT`、`CONNECT`、および `CONNECT RESET` ステートメントを生成しません。省略時アクションでは、これらのステートメントを生成します。`-m` オプションを指定しない場合、`-c` オプションは無視されます。

-t Tname

表名。特定の表への出力を制限します。

-p 平文テキスト形式を使用します。

-o Fname

LaTeX 形式を使用する場合、`filename.tex` に出力を書き込みます。平文テキスト形式を使用する場合、`filename.txt` に出力を書き込みます。このオプションを指定しない場合、出力は標準出力に書き込まれます。

-e データベース・オブジェクト用の DDL ステートメントを抽出します。このオプションは、`-m` オプションと一緒に使用できます。`-e` オプションを使用する場合には、以下のデータベース・オブジェクト用の DDL を抽出します。

- 表
- 視点
- 自動要約表 (AST)

db2look - DB2 統計および DDL 抽出ツール

- 別名
- 索引
- トリガー
- ユーザー定義特殊タイプ
- 基本キー、RI、および CHECK 制約
- ユーザー定義構造タイプ
- ユーザー定義関数
- ユーザー定義方式
- ユーザー定義変換

注: **db2look** によって生成される DDL を使用して、ユーザー定義関数を正常に再作成することができます。ただし、ユーザー定義関数を使用可能であるためには、特定のユーザー定義関数 (EXTERNAL NAME 節など) が参照するユーザー・ソース・コードが使用可能でなければなりません。

-m 必要な UPDATE ステートメントを生成して、表、列、および索引についての統計を複製します。-p、-g、および -s オプションは、-m オプションが指定された場合は無視されます。

-l このオプションを指定すると、**db2look** ユーティリティーは、ユーザー定義表スペース、ノード・グループ、およびバッファーク・プール用の DDL を生成します。以下のデータベース・オブジェクト用の DDL は、-l オプションを使用すると抽出されます。

- ユーザー定義表スペース
- ユーザー定義ノード・グループ
- ユーザー定義バッファーク・プール

-x このオプションを指定すると、**db2look** ユーティリティーは、権限 DDL (GRANT ステートメントなど) を生成します。

-i userid

リモート・データベースで作業する場合には、このオプションを使用してください。

-w password

-i オプションと共にこのパラメーターを使用すると、リモート・システムに常駐するデータベースに対して **db2look** が実行可能になります。**db2look** では、リモート・システムにログオンするために、ユーザー ID およびパスワードが使用されます。

-f このオプションは、構成パラメーターおよび登録変数を抽出するために使用します。

注: DB2 照会最適化プログラムに影響を与える構成パラメーターおよび登録変数だけが抽出されます。

例

データベース DEPARTMENT でユーザー valid によって作成されたオブジェクト用の DDL ステートメントを生成します。 **db2look** の出力は、以下のようにしてファイル db2look.sql に送信します。

```
db2look -d department -u valid -e -o db2look.sql
```

UPDATE ステートメントを生成して、データベース DEPARTMENT でユーザー valid によって作成された表および索引の統計を複製します。出力は、以下のようにしてファイル db2look.sql に送信します。

```
db2look -d department -u valid -m -o db2look.sql
```

ユーザー valid によって作成されたオブジェクト用の DDL ステートメントおよび UPDATE ステートメントの両方を生成して、同じユーザーによって作成された表および索引についての統計を複製します。 **db2look** の出力は、以下のようにしてファイル db2look.sql に送信します。

```
db2look -d department -u valid -e -m -o db2look.sql
```

データベース DEPARTMENT ですべてのユーザーによって作成されたオブジェクトの DDL ステートメントを生成します。 **db2look** の出力は、以下のようにしてファイル db2look.sql に送信します。

```
db2look -d department -a -e -o db2look.sql
```

すべてのユーザー定義のノード・グループ、バッファ・プール、および表スペース用の DDL ステートメントを生成します。 **db2look** の出力は、以下のようにしてファイル db2look.sql に送信します。

```
db2look -d department -l -o db2look.sql
```

データベースおよびデータベース・マネージャー構成パラメーター用の UPDATE ステートメント、およびデータベース DEPARTMENT にある登録変数用の **db2set** ステートメントを生成します。 **db2look** の出力は、以下のようにしてファイル db2look.sql に送信します。

```
db2look -d department -f -o db2look.sql
```

db2look - DB2 統計および DDL 抽出ツール

データベース DEPARTMENT にあるすべてのオブジェクト用の DDL、データベース DEPARTMENT にあるすべての表および索引についての統計を複製するための UPDATE ステートメント、GRANT 権限ステートメント、データベースおよびデータベース・マネージャー構成パラメーター用の UPDATE ステートメント、登録変数用の **db2set** ステートメント、およびデータベース DEPARTMENT にあるすべてのユーザー定義のノード・グループ、バッファ・プール、および表スペース用の DDL を生成します。出力は、ファイル db2look.sql に送信されます。

```
db2look -d department -a -e -m -l -x -f -o db2look.sql
```

db2move - データベース移動ツール

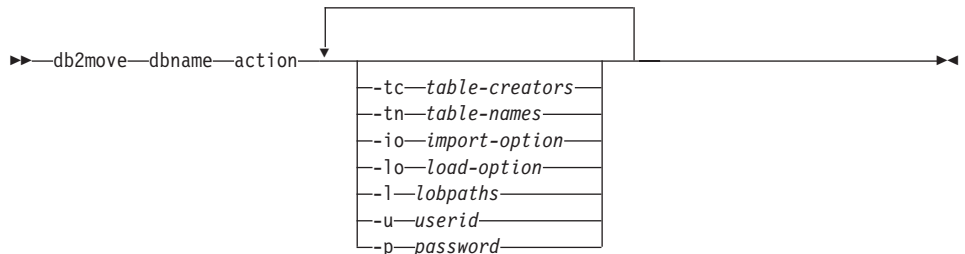
このツールは、ワークステーション上にある DB2 データベース間で、大量の表の移動を容易にします。また、特定のデータベースのシステム・カタログ表を照会し、すべてのユーザー表のリストをコンパイルします。そして、これらの表を PC/IXF 形式でエクスポートします。PC/IXF ファイルは、同じシステム上の別のローカル DB2 データベースにインポートまたはロードするか、または別のワークステーション・プラットフォームに転送し、そのプラットフォームで DB2 データベースにインポートまたはロードすることができます。

注: 構造型列がある表は、このツールを使用しても移動しません。

許可

このツールは、ユーザーから要求されるアクションにしたがって、DB2 エクスポート、インポート、およびロード API を呼び出します。したがって、要求元ユーザー ID には、これらの API に求められる正しい権限がなければなりません。この権限がないと、要求は失敗します。

コマンド構文



コマンド・パラメーター

dbname

データベースの名前。

action EXPORT、IMPORT、または LOAD のどれかです。

-tc 表の作成者。省略時値はすべての作成者です。

これは EXPORT アクションのみです。指定されると、このオプションでリストされる作成者が作成する表のみがエクスポートされます。指定されない場合、省略時ではすべての作成者を使用します。複数の作成者を指定する場合、それぞれをコンマで区切る必要があります。作成者 ID 間に空白を入れることはできません。指定できる作成者の最大

数は 10 です。このオプションは、エクスポートするために表を選択する場合に、『-tn』 表名オプションを指定して使用することができます。

アスタリスク (*) は、ストリング中のどこにでも入れられるワイルドカード文字として使用できます。

-tn 表名。省略時値はすべてのユーザー表です。

これは EXPORT アクションのみです。指定されると、指定されたストリング中の名前と完全に一致する名前を持つ表のみがエクスポートされます。指定されない場合、省略時ではすべてのユーザー表を使用します。複数の表名を指定する場合、それぞれをコンマで区切る必要があります。表名間にブランクを入れることはできません。指定できる表名の最大数は 10 です。このオプションは、エクスポートするために表を選択する場合に、『-tc』 表作成者オプションを指定して使用することができます。 **db2move** は、名前が指定された表名と一致し、かつ作成者が指定された表作成者と一致する表のみをエクスポートします。

アスタリスク (*) は、ストリング中のどこにでも入れられるワイルドカード文字として使用できます。

-io インポート・オプション。省略時値は REPLACE_CREATE です。

有効なオプションは、INSERT、INSERT_UPDATE、REPLACE、CREATE、および REPLACE_CREATE です。

-lo ロード・オプション。省略時値は INSERT です。

有効なオプションは、INSERT および REPLACE です。

-l LOB パス。省略時値は現行ディレクトリーです。

このオプションは、LOB ファイルが (EXPORT の一部として) 作成されるか、または (IMPORT または LOAD の一部として) 検索される絶対パス名を指定します。複数の LOB パスを指定する場合、それぞれをコンマで区切る必要があります。LOB パス間にブランクを入れることはできません。最初のパスでスペースを使い尽くした場合 (EXPORT 中)、またはパスでファイルが見つからない場合 (IMPORT または LOAD 中)、2 番目のパスが使用される、という方法でパスが使用されます。

アクションが EXPORT で LOB パスが指定される場合、LOB パス・ディレクトリーのすべてのファイルが削除され、ディレクトリーは除去され、新しいディレクトリーが作成されます。指定されない場合、現行ディレクトリーが LOB パスに使用されます。

- u ユーザー ID。省略時値はログオン・ユーザー ID です。
ユーザー ID とパスワードはどちらも任意指定です。しかし、一方を指定した場合、他方も必ず指定する必要があります。コマンドがリモート・サーバーに接続するクライアント上で実行される場合、ユーザー ID とパスワードを指定することが必要です。
- p パスワード。省略時値はログオン・パスワードです。
ユーザー ID とパスワードはどちらも任意指定です。しかし、一方を指定した場合、他方も必ず指定する必要があります。コマンドがリモート・サーバーに接続するクライアント上で実行される場合、ユーザー ID とパスワードを指定することが必要です。

例

- db2move sample export
この例では、SAMPLE データベースのすべての表をエクスポートします。すべてのオプションに省略時値が使用されます。
- db2move sample export -tc userid1,us*rid2 -tn tname1,*tname2
この例では、『userid1』 または 『us%rid2』 のようなユーザー ID で作成され、『tname1』 という名前、または 『%tname2』 のような表名を持つすべての表がエクスポートされます。
- db2move sample import -l D:¥LOBPATH1,C:¥LOBPATH2
この例は、OS/2 または Windows オペレーティング・システムのみにも適用されます。コマンドは、SAMPLE データベースのすべての表をインポートします。LOB パス 『D:¥LOBPATH1』 および 『C:¥LOBPATH2』 で、LOB ファイルが検索されます。
- db2move sample load -l /home/userid/lobpath,/tmp
この例は、UNIX ベースのシステムのみにも適用されます。コマンドは、SAMPLE データベースのすべての表をロードします。
/home/userid/lobpath サブディレクトリーと tmp サブディレクトリーの両方で、LOB ファイルが検索されます。
- db2move sample import -io replace -u userid -p password
この例では、SAMPLE データベースのすべての表が REPLACE モードでインポートされます。指定されたユーザー ID およびパスワードが使用されず。

使用上の注意

このツールはユーザーが作成した表をエクスポート、インポート、またはロードします。データベースが、あるオペレーティング・システムから別のオペレーティング・システムに複製される場合、**db2move** によって表の移動が容易になります。表と関連する他のすべてのオブジェクト、たとえば別名、視点、トリガーなども移動する必要があります。71ページの『db2look - DB2 統計および DDL 抽出ツール』によって、データベースからデータ定義言語 (DDL) ステートメントを抽出することにより、これらのオブジェクトのいくつかを容易に移動することができます。

エクスポート、インポート、またはロード API が **db2move** によって呼び出されると、FileTypeMod パラメーターが lobsinfile に設定されます。つまり、LOB データが PC/IXF ファイルとは別に保持されます。LOB ファイルでは、26 000 のファイル名が使用可能です。

LOAD アクションは、データベースおよびデータ・ファイルが常駐するマシンでローカルに実行する必要があります。ロード API が **db2move** によって呼び出されると、CopyTargetList パラメーターが NULL に設定されます。つまり、コピーは行われません。logretain がオンである場合、ロード操作を後でロールフォワードすることはできません。ロードされた表が常駐する表スペースはバックアップ保留状態にされ、アクセスできません。表スペースをバックアップ保留状態から解除するには、全データベースのバックアップまたは表スペースのバックアップが必要です。

バージョン 5.2 クライアントからバージョン 6 データベースに対して発行される場合、このツールは、長さが 18 文字を超える表名または列名はサポートしません。

EXPORT 使用時に必要とされるファイル / 生成されるファイル

- 入力: なし。
- 出力:

EXPORT.out EXPORT アクションの結果の要約。

db2move.lst オリジナル表名のリスト、その対応する PC/IXF ファイル名 (tabnnn.ixf)、およびメッセージ・ファイル名 (tabnnn.msg)。このリスト、エクスポートされた PC/IXF ファイル、および LOB ファイル (tabnnnc.yyy) は、**db2move** IMPORT または LOAD アクションへの入力として使用されます。

tabnnn.ixf 特定の表の、エクスポートされる PC/IXF ファイル。

tabnnn.msg 対応する表のエクスポート・メッセージ・ファイル。

tabnnnc.yyy 特定の表の、エクスポートされる LOB ファイル。

『nnn』は表番号です。『c』はアルファベットの文字です。『yyy』は 001 ~ 999 の範囲内の数値です。

これらのファイルは、エクスポートされている表に LOB データが入っている場合のみ作成されます。作成されると、これらの LOB ファイルは『lobpath』ディレクトリーに入れます。LOB ファイルには、合計 26 000 の可能な名前があります。

system.msg ファイルまたはディレクトリー・コマンドを作成または削除するための、システム・メッセージを含むメッセージ・ファイル。これは、アクションが EXPORT で、LOB パスが指定される場合のみ使用されます。

IMPORT 使用時に必要とされるファイル / 生成されるファイル

- 入力:

db2move.lst EXPORT アクションからの出力ファイル。

tabnnn.ixf EXPORT アクションからの出力ファイル。

tabnnnc.yyy EXPORT アクションからの出力ファイル。

- 出力:

IMPORT.out IMPORT アクションの結果の要約。

tabnnn.msg 対応する表のインポート・メッセージ・ファイル。

LOAD 使用時に必要とされるファイル / 生成されるファイル

- 入力:

db2move.lst EXPORT アクションからの出力ファイル。

tabnnn.ixf EXPORT アクションからの出力ファイル。

tabnnnc.yyy EXPORT アクションからの出力ファイル。

- 出力:

LOAD.out LOAD アクションの結果の要約。

tabnnn.msg 対応する表の LOAD メッセージ・ファイル。

db2mcs - Windows NT フェールオーバー・ユーティリティーのセットアップ

MSCS を使用する Windows NT 環境でフェールオーバーをサポートするための DB2 インフラストラクチャーを作成します。このユーティリティーを使用すると、単一区画環境と区分データベース環境の両方でフェールオーバーが可能になります。

このユーティリティーの使用の詳細については、[管理の手引き](#) を参照してください。

許可

ユーザーは、MSCS クラスタ内の各マシンの管理者グループに属するドメイン・ユーザー・アカウントにログオンする必要があります。

コマンド構文

```
db2mcs [-f:input_file]
```

コマンド・パラメーター

-f:input_file

MSCS ユーティリティーによって使用される DB2MSCS.CFG 入力ファイルを指定します。このパラメーターが指定されない場合、DB2MSCS ユーティリティーは、現行のディレクトリーにある DB2MSCS.CFG ファイルを読み取ります。

db2nchg - データベース区画サーバー構成の変更

データベース区画サーバー構成を変更します。これには、あるマシンから別のマシンへのデータベース区画サーバー（ノード）の移動、マシンの TCP/IP ホスト名の変更、データベース区画サーバー（ノード）用の別の論理ポート番号または別のネットワーク名の選択も含まれます。このコマンドが使用できるのは、データベース区画サーバーが停止している場合だけです。

このコマンドは Windows NT および Windows 2000 でのみ使用可能です。

許可

ローカル管理者

コマンド構文

```

▶▶ db2nchg /n:—node_number— /i:—instance_name— /u:—user,password—
    /p:—logical_port— /h:—hostname— /m:—machine_name—
    /g:—network_name—
  
```

コマンド・パラメーター

/n:node_number

変更するデータベース区画サーバー構成のノード番号を指定します。

/i:instance_name

このデータベース区画サーバーが参加するインスタンスを指定します。パラメーターが指定されていない場合、省略時値は現行のインスタンスになります。

/u:username,password

ユーザー名およびパスワードを指定します。パラメーターが指定されない場合、既存のユーザー名とパスワードが設定されます。

/p:logical_port

データベース区画サーバー用の論理ポートを指定します。データベース区画サーバーを別のマシンに移動させるには、このパラメーターを指定する必要があります。パラメーターが指定されない場合、論理ポート番号は変更されません。

db2nchg - データベース区画サーバー構成の変更

/h:host_name

内部通信用に FCM によって使用される TCP/IP ホスト名を指定します。パラメーターが指定されない場合、ホスト名は変更されません。

/m:machine_name

データベース区画サーバーが常駐するマシンを指定します。インスタンスに既存のデータベースがない場合にのみ、データベース区画サーバーを移動させることができます。

/g:network_name

データベース区画サーバーのネットワーク名を変更します。このパラメーターは、マシンに複数の IP アドレスがある場合に、特定の IP アドレスをデータベース区画サーバーに適用するために使用できます。ネットワーク名または IP アドレスを指定できます。

例

インスタンス TESTMPP に参加する、ノード 2 に割り当てられている論理ポートを論理ポート 3 に変更するには、以下のコマンドを入力します。

```
db2nchg /n:2 /i:TESTMPP /p:3
```

以下も参照

85ページの『db2ncrt - インスタンスへのデータベース区画サーバーの追加』

db2nrcrt - インスタンスへのデータベース区画サーバーの追加

データベース区画サーバー (ノード) をインスタンスに追加します。

このコマンドは Windows NT および Windows 2000 でのみ使用可能です。

効力範囲

すでにインスタンスが存在しているマシンにデータベース区画サーバーが追加される場合には、データベース区画サーバーはマシンへの論理ノードとして追加されます。インスタンスが存在していないマシンにデータベース区画サーバーが追加される場合には、インスタンスが追加され、そのマシンは新しい物理ノードになります。インスタンスにデータベースがある場合には、このコマンドを使用してはなりません。代わりに、db2start addnode nodenum コマンドを使用する必要があります。こうすると、新しいデータベース区画サーバーにデータベースが確実に正しく追加されます。データベースが作成されたインスタンスにノードを追加するには、[管理の手引き](#) を参照してください。

注: db2nodes.cfg ファイルは編集するべきではありません。このファイルを変更すると、区分データベース・システムに不整合が生じる可能性があるためです。

許可

新しいノードが追加されるマシンに対するローカル管理者権限。

コマンド構文

```

▶▶ db2nrcrt /n:—node_number /u:—username,password [ /i:—instance_name ]
    [ /m:—machine_name ] [ /p:—logical_port ] [ /h:—host_name ]
    [ /g:—network_name ] [ /o:—instance_owing_machine ]

```

コマンド・パラメーター**/n:node_number**

データベース区画サーバーを識別する固有のノード番号。 1～999 の範囲の番号を指定できます。

db2ncrt - インスタンスへのデータベース区画サーバーの追加

/u:domain_name¥username,password

DB2 のドメイン、ログオン・アカウント名およびパスワードを指定します。

/i:instance_name

インスタンス名を指定します。パラメーターが指定されていない場合、省略時値は現行のインスタンスになります。

/m:machine_name

ノードが常駐する Windows NT ワークステーションのコンピューター名を指定します。データベース区画サーバーをリモート・マシン上に追加している場合、このパラメーターは必須です。

/p:logical_port

データベース区画サーバーに使用する論理ポート番号を指定します。このパラメーターが指定されていない場合、割り当てられる論理ポート番号は 0 です。

注: 論理ノードを作成する際には、このパラメーターを指定しなければならず、使用していない論理ポート番号を選択しなければなりません。以下の制限事項に注意してください。

- すべてのマシンには、論理ポートが 0 のデータベース区画サーバーがなければなりません。
- ポート番号は、x:¥winnnt¥system32¥drivers¥etc¥ ディレクトリーで FCM 通信に予約されているポートの範囲内でなければなりません。たとえば、4 個のポートが現行のインスタンスに予約されている場合には、最大のポート番号は 3 になります。ポート 0 は、省略時の論理ノード用に使用されます。

/h:host_name

内部通信用に FCM によって使用される TCP/IP ホスト名を指定します。データベース区画サーバーをリモート・マシン上に追加する場合、このパラメーターは必須です。

/g:network_name

データベース区画サーバーのネットワーク名を指定します。パラメーターが指定されていない場合、システムで検出される最初の IP アドレスが使用されます。このパラメーターは、マシンに複数の IP アドレスがある場合に、特定の IP アドレスをデータベース区画サーバーに適用するために使用できます。ネットワーク名または IP アドレスを入力できます。

db2ncrt - インスタンスへのデータベース区画サーバーの追加

/o:instance_owning_machine

インスタンスを所有しているマシンのコンピューター名を指定します。省略時値はローカル・マシンです。インスタンス所有マシンではない任意のマシンで **db2ncrt** コマンドが呼び出される場合、このパラメーターは必須です。

例

インスタンス所有のマシン SHAYER 上で、インスタンス TESTMPP に新しいデータベース区画サーバーを追加する場合、新しいデータベース区画サーバーがノード 2 と認識されており、論理ポート 1 を使用している場合には、次のコマンドを入力します。

```
db2ncrt /n:2 /u:QBPAULZ¥paulz,g1reeky /i:TESTMPP /m:TEST /p:1 /o:SHAYER
```

以下も参照

83ページの『db2nchg - データベース区画サーバー構成の変更』

db2ndrop - インスタンスからのデータベース区画サーバーの除去

db2ndrop - インスタンスからのデータベース区画サーバーの除去

データベースのないインスタンスからデータベース区画サーバー（ノード）を除去します。データベース区画サーバーが除去された場合には、このノード番号を新しいデータベース区画サーバーで再使用できます。このコマンドは、データベース区画サーバーが停止している場合にのみ使用できます。

このコマンドは、Windows NT および Windows 2000 でのみ使用可能です。

許可

ノードを除去するマシンに対するローカル管理者権限。

コマンド構文

```
db2ndrop /n:—node_number /i:—instance_name
```

コマンド・パラメーター

/n:node_number

データベース区画サーバーを識別する固有のノード番号。

/i:instance_name

インスタンス名を指定します。パラメーターが指定されていない場合、省略時値は現行のインスタンスになります。

例

```
db2ndrop /n:2 /i=KMASCI
```

使用上の注意

インスタンスの所有するデータベース区画サーバー（ノード 0）がインスタンスから除去されると、このインスタンスは使用できなくなります。インスタンスを除去するには、**db2idrop** コマンドを使用します。

このインスタンスにデータベースがある場合には、このコマンドを使用してはなりません。代わりに、db2stop drop nodenum コマンドを使用する必要があります。こうすると、区分データベース・システムからデータベース区画サーバーを確実に除去することができます。データベースが存在するインスタンスでノードを除去するには、[管理の手引き](#) を参照してください。

db2ndrop - インスタンスからのデータベース区画サーバーの除去

注: db2nodes.cfg ファイルは編集するべきではありません。このファイルを変更すると、区分データベース・システムに不整合が生じる可能性があるためです。

複数の論理ノードを実行しているマシンから、論理ポート 0 に割り当てられたノードを除去するには、他の論理ポートに割り当てられている他のすべてのノードを最初に除去する必要があります。各データベース区画サーバーには、論理ポート 0 に割り当てられているノードが必ず必要です。

以下も参照

83ページの『db2nchg - データベース区画サーバー構成の変更』

db2perfc - データベース・パフォーマンス値のリセット

db2perfc - データベース・パフォーマンス値のリセット

Windows NT パフォーマンス・モニターで使用されます。1 つ以上のデータベースのパフォーマンス値をリセットします。アプリケーションが DB2 モニター API を呼び出したときに戻される値は、DB2 が開始して以来の累積値になります。しかしたいの場合、パフォーマンス値をリセットし、テストを実行し、再び値をリセットしてからテストを再実行する方がよいでしょう。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

db2perfi - パフォーマンス・カウンター登録ユーティリティ

DB2 (Windows NT 版) パフォーマンス・カウンター用の DLL を登録します。これは、DB2 および DB2 コネクトのパフォーマンス情報を、Windows NT パフォーマンス・モニターにアクセス可能にするために実行する必要があります。また、これによって Win32 パフォーマンス API を使用する他の Windows NT アプリケーションが、パフォーマンス・データを入手することもできます。

このコマンドの詳細については、[管理の手引き](#) を参照してください。

db2perfr - パフォーマンス・モニター登録ツール

Windows NT パフォーマンス・モニターで使用されます。管理者ユーザー名およびパスワードを DB2 で登録します。これは、Windows NT パフォーマンス・オブジェクトが、別の DB2 (Windows NT 版) マシンから認識される前に実行する必要があります。(省略時の Windows NT パフォーマンス・モニター・ユーザー名 SYSTEM は、DB2 予約語なので使用できません。)

このコマンドの詳細については、[管理の手引き](#) を参照してください。

db2prof - DB2 SQLJ プロファイル・カスタマイザー

組み込み SQL ステートメントを含む SQLJ プロファイルを処理します。省略時では、データベース内に DB2 パッケージが作成されます。このユーティリティーは、実行時に使用する DB2 固有の情報をプロファイルに入れます。このユーティリティーは、SQLJ アプリケーションが変換された後の、アプリケーションが実行される前に実行する必要があります。

許可

以下のどれかが必要です。

- *sysadm* または *dbadm* の権限
- パッケージが存在しない場合は、BINDADD 特権および以下のどちらかが必要です。
 - パッケージのスキーマ名が存在しない場合は、データベースに対する IMPLICIT_SCHEMA 権限
 - パッケージのスキーマ名が存在する場合は、スキーマに対する CREATEIN 特権
- パッケージが存在する場合は、スキーマに対する ALTERIN 特権
- パッケージに対する BIND 特権 (パッケージが存在する場合)

ユーザーには、アプリケーション内の静的 SQL ステートメントをコンパイルするのに必要な特権もすべて必要になります。グループに付与された特権は、静的ステートメントの許可検査では使用されません。ユーザーに *sysadm* 権限があってもバインドを完成させる明示特権がない場合、データベース・マネージャーは、明示的な *dbadm* 権限を自動的に付与します。

必須接続

このコマンドは、データベース接続を確立します。

コマンド構文

```

▶▶ db2prof -user=username -password=password

```

```

▶▶ -preoptions="precompile-options" -url=JDBC-url-profilename

```

コマンド・パラメーター

-user= username

プロファイルのカスタマイズを実行するためにデータベースに接続するときに使用する名前を指定します。

-password= password

ユーザー名のパスワードを指定します。

-prepoptions= "precompile-options"

DB2 プリコンパイラーで使用されるプリコンパイル・オプションのリストを指定します。サポートされるプリコンパイル・オプションのリストについては、[アプリケーション開発の手引き](#) を参照してください。

プリコンパイル・パッケージ「PACKAGE USING package-name」は、プリコンパイラーが生成するパッケージの名前を指定します。名前を入力しないと、プロファイルの名前 (拡張子を除いて大文字に変換したもの) が使われます。最大長は 8 文字です。

プリコンパイル・オプション「BINDFILE USING bind-file」は、プリコンパイラーが生成するバインド・ファイルの名前を指定します。ファイル名には、.bnd 拡張子が付いていなければなりません。ファイル名を入力しないと、プリコンパイラーはプロファイルの名前を使用し、それに .bnd 拡張子を付けてファイル名とします。パスを指定しないと、バインド・ファイルは現行ディレクトリーに作成されます。

-url= JDBC-url

データベース接続の設定に使用する JDBC URL を指定します。

profilename

SQL ステートメントが保管されるプロファイルの名前を指定します。SQLJ ファイルが Java ファイルに変換されるとき、ファイルに入っている SQL 操作に関する情報は、プロファイルと呼ばれる、SQLJ が生成した資源ファイルに保管されます。プロファイルは、元の入力ファイル名に接尾部 _SJProfileN (N は整数) が付いた名前によって識別されます。拡張子は .ser です。プロファイル名を指定するとき、.ser 拡張子はあってもなくてもかまいません。

例

```
db2profrc -user=username -password=password -url=JDBC-url
-prepoptions="bindfile using pgmname1.bnd package using pgmname1"
pgmname_SJProfile1.ser
```


使用上の注意

SQLJ 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。

以下も参照

96ページの『db2profp - DB2 SQLJ プロファイル・プリンター』

db2profp - DB2 SQLJ プロファイル・プリンター

db2profp - DB2 SQLJ プロファイル・プリンター

DB2 カスタマイズ・バージョンのプロファイルの内容を平文テキストで印刷します。

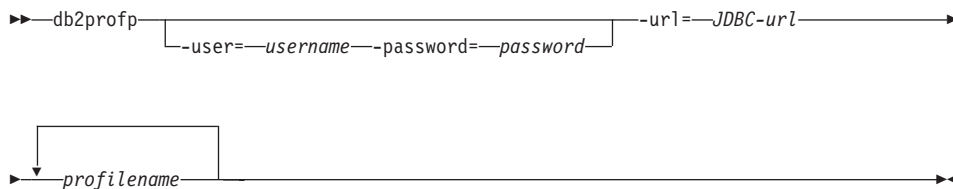
許可

なし

必須接続

このコマンドは、データベース接続を確立します。

コマンド構文



コマンド・パラメーター

-user= username

カスタマイズ済みプロファイルを印刷するためにデータベースに接続するとき使用する名前を指定します。

-password= password

ユーザー名のパスワードを指定します。

-url= JDBC-url

データベース接続の設定に使用する JDBC URL を指定します。

profilename

SQL ステートメントが保管される 1 つまたは複数のプロファイルを指定します。SQLJ ファイルが Java ファイルに変換されるとき、ファイルに入っている SQL 操作に関する情報は、プロファイルと呼ばれる、SQLJ が生成した資源ファイルに保管されます。プロファイルは、元の入力ファイル名に接尾部 `_SJProfileN` (N は整数) が付いた名前によって識別されます。拡張子は `.ser` です。プロファイル名を指定するとき、`.ser` 拡張子はあってもなくてもかまいません。

例

```
db2profp -user=username -password=password -url=JDBC-url  
pgmname_SJProfile1.ser
```

使用上の注意

SQLJ 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。

以下も参照

93ページの『db2profc - DB2 SQLJ プロファイル・カスタマイザー』

db2rbind - すべてのパッケージの再バインド

db2rbind - すべてのパッケージの再バインド

データベース内のパッケージを再バインドします。

許可

以下のどれかが必要です。

- *sysadm*
- *dbadm*

必須接続

なし

コマンド構文

```
▶ db2rbind database -/l logfile [all] [/u userid -/p password]
▶ -/r [conservative] any
```

コマンド・パラメーター

database

再び妥当性検査を行うパッケージが含まれているデータベースの別名を指定します。

/l パッケージの再妥当性検査プロシージャーからのエラーを記録するときに使用するパス (任意指定) とファイル名 (必須) を指定します。

all すべての有効および無効パッケージの再バインドが実行されるように指定します。このオプションを指定しないと、データベース内のすべてのパッケージが検査されますが、アプリケーションの実行時に暗黙的に再バインドされることのないよう、無効のマークが付いたパッケージのみを再バインドします。

/u ユーザー ID。パスワードを指定する場合には、このパラメーターを指定しなければなりません。

/p パスワード。ユーザー ID を指定する場合には、このパラメーターを指定しなければなりません。

/r 解決。パッケージの再バインドの実行に、従来のバインド・セマンティクスを使用するかどうかを指定します。これは、パッケージの静的

DML ステートメントの関数およびタイプの解決時に、新しい関数およびデータ・タイプを対象にするかどうかに影響します。このオプションは DRDA ではサポートされていません。有効な値は以下のとおりです。

conservative

関数およびタイプの解決時に、最後の明示的バインドのタイム・スタンプより前に定義された SQL パスにある関数とタイプだけを対象にします。従来のバインド・セマンティクスを使用します。これが省略時値です。

any 関数およびタイプの解決時に、SQL パスにあるすべての関数とタイプを対象にします。従来のバインド・セマンティクスは使用されません。

使用上の注意

このコマンドは、データベース内のすべてのパッケージの妥当性検査を再び行うために、CLP REBIND コマンドを使用します。必ずしも **db2rbind** を使用しなければならないわけではありません。パッケージを最初に使用するときパッケージの再妥当性検査を暗黙のうちに行えるようにするか、または REBIND か BIND コマンドのどちらかを使用して選択した特定のパッケージにだけ再妥当性検査を行うようにすることもできます。

以下も参照

165ページの『BIND』

496ページの『REBIND』

db2sampl - サンプル・データベースの作成

SAMPLE という名前のサンプル・データベースを作成します。このデータベースの詳細については、*SQL 解説書* を参照してください。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*

コマンド構文

```
db2sampl [-path] [-k]
```

コマンド・パラメーター

path SAMPLE データベースを作成するパスを指定します。OS/2 と Windows の場合、パスは 1 文字のドライブ名になります。

パスを指定しない場合、SAMPLE は省略時のデータベース・パス (データベース・マネージャ構成ファイルの *dftdbpath* パラメーター) に作成されます。UNIX ベースのシステムでは、省略時値はインスタンス所有者の HOME ディレクトリーです。OS/2 または Windows オペレーティング・システムでは、(DB2 がインストールされている) ルート・ディレクトリーになります。

-k 以下の SAMPLE 表について基本キーを作成します。

表	基本キー
-----	-----
DEPARTMENT	DEPTNO
EMPLOYEE	EMPNO
ORG	DEPTNUMB
PROJECT	PROJNO
STAFF	ID
STAFFG	ID (DBCS のみ)

注: パスは、このオプションの前に 指定しなければなりません。

使用上の注意

このコマンドはサーバー・ノードからしか実行できません。SAMPLE は、データベース・クライアントのみのノードに作成することはできません。

db2sampl - サンプル・データベースの作成

SAMPLE データベースは、データベース・マネージャー構成パラメーター *authentication* (292ページの『GET DATABASE MANAGER CONFIGURATION』参照) で指定したインスタンス認証タイプで作成されます。

SAMPLE 内の表の修飾子は、コマンドを発行しているユーザー ID が決めます。

SAMPLE がすでに存在する場合、**db2sampl** はコマンドを発行しているユーザー ID のために表を作成し、適切な特権を付与します。

db2set - DB2 プロファイル・レジストリー・コマンド

DB2 プロファイル変数を表示、設定、または除去します。DB2 プロファイル・レジストリーには、DB2 管理サーバーを経由しての、DB2 の環境変数のローカルおよびリモート管理をサポートする外部環境レジストリー・コマンドが保管されています。

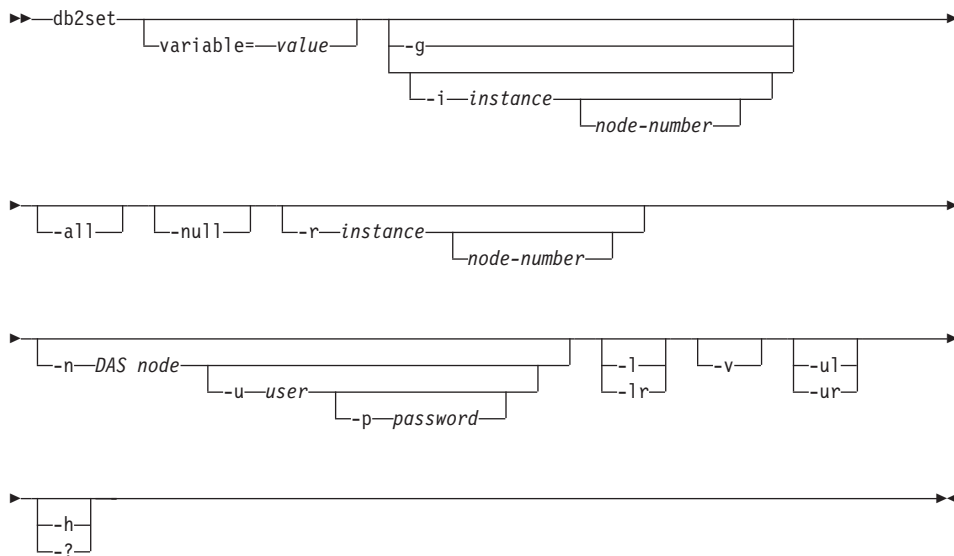
許可

sysadm

必須接続

なし

コマンド構文



コマンド・パラメーター

variable= value

指定値に、指定される変数を設定します。変数を削除する場合は、指定される変数に値を指定しないでください。設定の変更は、インスタンスが再始動された後に有効になります。

-g グローバル・プロファイル変数にアクセスします。

-i インスタンス・プロファイルを指定して、現行値または省略時値の代わりに使用します。

node-number

db2nodes.cfg ファイルでリストされる番号を指定します。

-all 以下の形式で定義されたローカル環境変数のオカレンスをすべて表示します。

- 環境 ([e] で表示)
- ノード・レベル・レジストリー ([n] で表示)
- インスタンス・レベル・レジストリー ([i] で表示)
- グローバル・レベル・レジストリー ([g] で表示)

-null 指定したレジストリー・レベルの変数値をヌル値に設定します。これによって探索順で定義された次のレジストリー・レベルの値を探索せずに済みます。

-r instance

指定したインスタンスのプロファイル・レジストリーをリセットします。

-n DAS node

リモート DB2 管理サーバーのノード名を指定します。

-u user

管理サーバー接続を使用する際のユーザー ID を指定します。

-p password

管理サーバー接続を使用する際のパスワードを指定します。

-l すべてのインスタンス・プロファイルをリストします。

-lr サポートされているレジストリー変数をすべてリストします。

-v 冗長モードを指定します。

-ul ユーザー・プロファイル変数にアクセスします。

注: このパラメーターは Windows オペレーティング・システム上だけでサポートされます。

-ur ユーザー・プロファイル変数を最新表示します。

注: このパラメーターは Windows オペレーティング・システム上だけでサポートされます。

-h/-? ヘルプ情報を表示します。このオプションを指定すると、他のすべてのオプションは無視され、ヘルプ情報だけが表示されます。

例

- 定義済みプロファイル (DB2 インスタンス) をすべて表示するには、次のように入力します。

```
db2set -l
```

- サポートされているレジストリー変数をすべて表示するには、次のように入力します。

```
db2set -lr
```

- 定義済み大域変数をすべて表示するには、次のように入力します。

```
db2set -g
```

- 現行インスタンスの定義済み変数をすべて表示するには、次のように入力します。

```
db2set
```

- 現行インスタンス用に定義された値をすべて表示するには、次のように入力します。

```
db2set -all
```

- 現行インスタンスの DB2COMM 用に定義された値をすべて表示するには、次のように入力します。

```
db2set -all DB2COMM
```

- ノード 3 のインスタンス INST 用に定義された変数をすべてリセットするには、次のように入力します。

```
db2set -r -i INST 3
```

- ユーザー ID に MYID およびパスワードに MYPASSWD を使用し、DAS ノード RMTDAS を介してリモート・インスタンス RMTINST の変数 DB2CHKPTR を設定解除するには、次のように入力します。

```
db2set -i RMTINST -n RMTDAS -u MYID -p MYPASSWD DB2CHKPTR=
```

- 変数 DB2COMM を TCPIP、IPXSPX、NETBIOS に大域的に設定するには、次のように入力します。

```
db2set -g DB2COMM=TCPIP,IPXSPX,NETBIOS
```

- 変数 DB2COMM を、インスタンス MYINST 用の TCPIP のみで設定するには、次のように入力します。

```
db2set -i MYINST DB2COMM=TCPIP
```

- 変数 DB2COMM を指定したインスタンス・レベルでヌル値に設定するには、次のように入力します。

```
db2set -null DB2COMM
```

使用上の注意

変数名を指定しないと、定義済み変数の値がすべて表示されます。変数名を 1 つだけ 指定した場合、その変数の値だけが表示されます。変数の値をすべて表示するには、`variable -all` を指定します。すべてのレジストリーに定義されている変数をすべて表示するには、`-all` を指定します。

変数の値を修正するには、`variable=` の後に新規の値を指定します。変数の値に NULL を指定するには、`variable -null` を指定します。

注: 設定の変更は、インスタンスが再始動された後に有効になります。

変数を削除するには、値を指定せずに `variable=` だけにします。

db2sql92 - SQL92 準拠 SQL ステートメント・プロセッサ

フラット・ファイルまたは標準入力のどちらかから SQL ステートメントを読み取り、ステートメントを動的に記述、準備し、応答セットを戻します。複数のデータベースへの並行接続をサポートします。

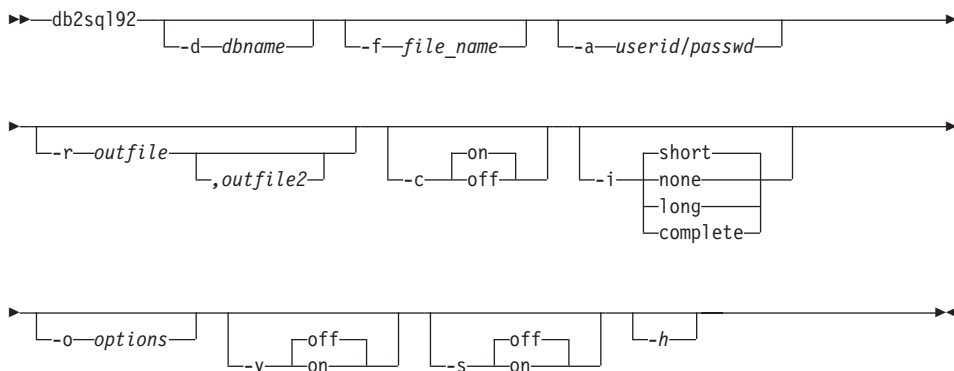
許可

sysadm

必須接続

なし。このコマンドは、データベース接続を確立します。

コマンド構文



コマンド・パラメーター

-d dbname

SQL ステートメントが適用されるデータベースの別名。省略時は、**DB2DBDFT** 環境変数の値です。

-f file_name

SQL ステートメントが入っている入力ファイルの名前。省略時は標準入力です。

注釈テキストは、各行の先頭に 2 つのハイフンを付けて `-- <注釈>` で表します。注釈を出力にも含めるときは、次のように注釈にマークを付けます。 `--#COMMENT <注釈>`。

ブロック は、一まとまりとして処理されるいくつかの SQL ステートメントからなっています。つまり、ステートメントで使用する情報を 1 つずつ収集するのではなく、すべてのステートメントに必要な情報を一

db2sql92 - SQL92 準拠 SQL ステートメント・プロセッサ

度に収集します。照会ブロックの開始は、--#BGBLK で表します。照会ブロックの終了は、--#E0BLK で表します。

1 つまたは複数の制御オプションを指定するには、次のようにします。
--#SET <制御オプション> <値>。有効な制御オプションは、以下のとおりです。

ROWS_FETCH

応答セットから取り出す行数。有効な値は $-1 \sim n$ です。省略時値は -1 (すべての行を取り出す) です。

ROWS_OUT

取り出された行のうち出力へ送られる行数。有効な値は $-1 \sim n$ です。省略時値は -1 (取り出された行をすべて出力へ送る) です。

AUTOCOMMIT

自動コミットのオン、またはオフを指定します。有効な値は、ON または OFF です。省略時値は、ON です。

PAUSE

継続するかどうかの入力を要求するプロンプトをユーザーに出します。

TIMESTAMP

タイム・スタンプを生成します。

-a userid/passwd

データベースへの接続に使用する名前およびパスワード。

-r outfile

照会結果が入る出力ファイル。任意指定の出力ファイル 2 には、結果の要約が入ります。省略時値は標準出力です。

-c 各 SQL ステートメントの実行による変更を自動的にコミットします。

-i 経過時間の間隔 (秒単位)。

none 時間情報を収集しないことを指定します。

short 照会の実行時間。

long 次の照会が開始する経過時間。

complete

準備、実行、および取り出しの時間。別個に表示します。

-o options

制御オプション。有効なオプションは以下のとおりです。

f rows_fetch

応答セットから取り出す行数。有効な値は $-1 \sim n$ です。省略時値は -1 (すべての行を取り出す) です。

r rows_out

取り出された行のうち出力へ送られる行数。有効な値は $-1 \sim n$ です。省略時値は -1 (取り出された行をすべて出力へ送る) です。

- v** 冗長。照会処理中に標準エラーに情報を送信します。省略時値は OFF です。
- s** 要約表。収集した値の算術平均と幾何平均の両方を含む、経過時間と CPU 時間の要約を提供します。
- h** ヘルプ情報を表示します。このオプションを指定すると、他のすべてのオプションは無視され、ヘルプ情報だけが表示されます。

使用上の注意

db2sql92 コマンド・プロンプトで以下を実行できます。

- すべての制御オプション
- SQL ステートメント
- CONNECT ステートメント
- コミット作業
- ヘルプ
- 終了

このツールは、プログラムの単一実行中での異なる複数のデータベース間の切り替えをサポートしています。これを行うために **CONNECT RESET** を発行し、次に **db2sql92** コマンド・プロンプト (stdin) で以下のどちらかを入力します。

```
connect to database
connect to database USER userid USING passwd
```

SQL ステートメントの長さの上限は、65 535 文字です。ステートメントの末尾はセミコロンにしなければなりません。

SQL ステートメントは、反復可能読取り (RR) 分離レベルで実行されます。

以下も参照

15ページの『db2batch - ベンチマーク・ツール』

db2start - DB2 の開始

単一ノードまたはマルチノード環境で定義されているすべてのノードで、現行のデータベース・マネージャー・インスタンス背景処理を開始します。データベースへの接続、アプリケーションの事前コンパイル、またはパッケージのデータベースへのバインドの前に、サーバーで DB2 を始動します。

db2start は、システム・コマンドまたは CLP コマンドとして実行することができます。このコマンドの詳細については、576ページの『START DATABASE MANAGER』を参照してください。

db2stop - DB2 の停止

db2stop - DB2 の停止

現行のデータベース・マネージャー・インスタンスを停止します。

db2stop は、システム・コマンドまたは CLP コマンドとして実行することができます。このコマンドの詳細については、582ページの『STOP DATABASE MANAGER』を参照してください。

db2sync - DB2 シンクロナイザーの開始

サテライトの初期構成および構成の変更が簡単になります。このコマンドは、同期化セッションの進行を停止またはモニターしたり、サテライトの構成情報(たとえば、通信パラメーター)をコントロール・サーバーにアップロードしたりすることにも使用できます。

許可

なし

必須接続

なし

コマンド構文

```
▶▶ db2sync [ -t | -s application_version | -g ] ▶▶
```

コマンド・パラメーター

- t** 管理者がサテライトのアプリケーション・バージョンまたは同期化認証のいずれかを変更できる、グラフィカル・ユーザー・インターフェースを表示します。
- s application_version** サテライトにアプリケーションのバージョンを設定します。
- g** サテライトに現在設定されているアプリケーションのバージョンを表示します。

使用上の注意

サテライト同期の全般的な情報については、[サテライト管理 手引き](#)および[解説書](#)を参照してください。

db2tbst - 表スペース状態の獲得

db2tbst - 表スペース状態の獲得

16 進数の表スペース状態値を受け入れ、その状態を戻します。状態値は 405 ページの『LIST TABLESPACES』からの出力の一部です。

許可

なし

必須接続

なし

コマンド構文

▶—db2tbst—*tablespace-state*————▶

コマンド・パラメーター

tablespace-state

16 進数の表スペース状態値。

例

db2tbst 0x0000 の要求は、次の出力を生成します。
State = Normal

以下も参照

405 ページの『LIST TABLESPACES』

db2trc - トレース

DB2 トレース機能のトレースを活動化します。DB2 は独自のトレース機能を使用して、事象をトレースし、トレース・データをファイルにダンプし、トレース・データを読み取り可能な形式にフォーマットします。DB2 顧客サービスが指示した場合、または技術サポート担当者が指示した場合にのみ、トレース機能を使用します。

許可

UNIX ベースのシステムでは、次のどれかが必要です。

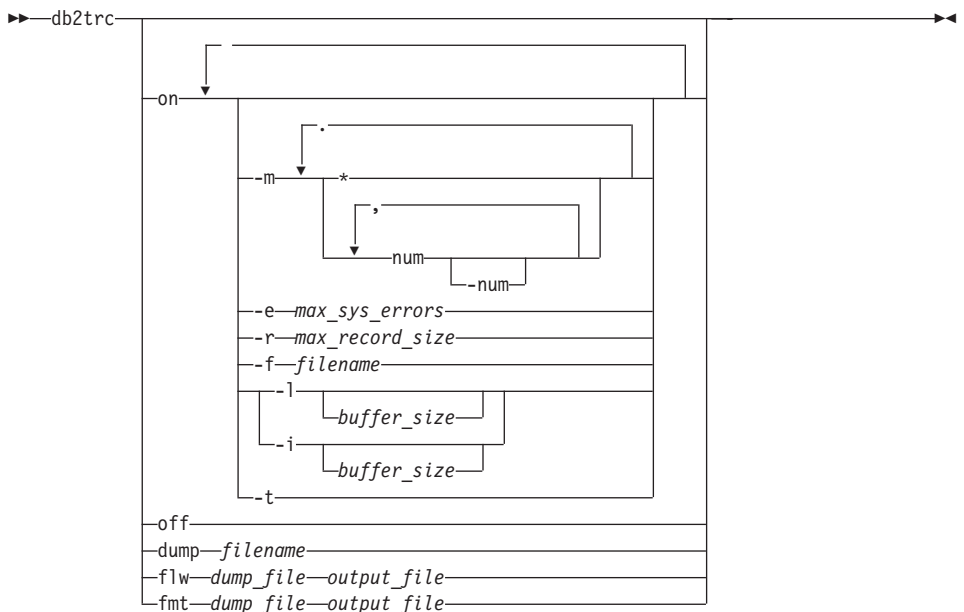
- *sysadm*
- *sysctrl*
- *sysmaint*

OS/2 または Windows オペレーティング・システムでは、許可は必要ありません。

必須接続

なし

コマンド構文



コマンド・パラメーター

- on** DB2 トレース機能を開始するには、このパラメーターを使用します。このパラメーターのオプションについては、次の項を参照してください。
- dump** エラーを複製した後で、トレース情報をファイルにダンプします。次のコマンドは、トレース情報を現行ディレクトリーの `db2trc.dmp` というファイルに入れます。

```
db2trc dump db2trc.dmp
```

このパラメーターでファイル名を指定します。明示的にパスが指定されていない場合、そのファイルは現行ディレクトリーに保管されます。

- off** トレースをファイルにダンプした後、次のように入力してトレースをオフにします。

```
db2trc off
```

flw | fmt

トレースをバイナリー・ファイルにダンプした後、それが ASCII ファイルにフォーマットされたことを確認します。 `flw` オプション (プロセスまたはスレッドで分類する)、または `fmt` オプション (事象が起きた順にリストする) のどちらかを使用します。どちらのオプションの場合でも、ダンプ・ファイル名および生成される出力ファイルの名前を指定します。たとえば、次のようにします。

```
db2trc flw db2trc.dmp db2trc.flw
```

DB2 トレースの開始

トレース機能を使用するには、まず **db2trc on** を使用してトレース機能をオンにします。DB2 顧客サービスからは他の指示がない場合、オプションを指定せずにこのコマンドを使用します。省略時のトレース・オプション値は、以下のとおりです。

- `mask = *` (すべてトレースする)
- `max_sys_errors = -1` (すべて収集する)
- `max_record_size = 16KB`
- `Trace destination = -s` (共用メモリー)
- `Records to retain = -1` (最後のレコード)
- `buffer_size = 64KB` (トレースの宛先が共用メモリーの場合)、または `1MB` (トレースの宛先がファイルの場合)

オプションを指定してトレースを調整するように指示された場合、DB2 顧客サービスの指示通りに以下のオプションを使用します。

-m *mask*

トレース・レコード・タイプを指定して、探索をフォーカスします。
mask 変数は、ピリオドで区切られた 4 つの 1 バイト・マスクからなっています。バイト・マスクは、それ自体の ID に基づいて事象ごとに DB2 が送信するトレース・レコードを受け入れたり、拒否したりするフィルターとして動作します。4 つの 1 バイト・マスクは、製品、事象タイプ、構成要素および関数にそれぞれ対応しています。

-e *max_sys_errors*

トレースが *max_sys_errors* に保持する DB2 内部システム・エラーの数を制限します。

-r *max_record_size*

トレース・レコードのサイズを *max_record_size* バイトに制限します。長いトレース・レコードは切り捨てられます。

-f *filename*

共用メモリーまたはファイル中の追跡バッファのロケーションを指定します。

-l [*buffer_size*] | **-i** [*buffer_size*]

'l' ("lucky" の "l") は、最後のトレース・レコードを保存することを指定します (つまり、バッファが満杯になると最初のレコードが重ね書きされます)。'i' は、最初のトレース・レコードを保存することを指定します (つまり、バッファがいったん満杯になると、レコードの書き込みはそれ以上行われません)。これらのオプションのどちらかを使用して、バッファ・サイズを指定してください。

-t タイム・スタンプを含めます。これは UNIX 環境でのみ適用できます。タイム・スタンプのログを記録するとパフォーマンスに影響を与えます。

例

トレース例を含め、**db2trc** についての追加情報は、*問題判別の手引き* を参照してください。

使用上の注意

db2trc コマンドは、トレースをオンにする時、ダンプ・ファイルを生成する時、ダンプ・ファイルをフォーマットする時、トレースを再びオフにする時など、何回か発行する必要があります。パラメーターのリストは、そのパラメー

db2trc - トレース

ターを使用する順序になっています。

db2uidl - V5 セマンティクスへの固有索引変換の準備

ユーザー自身のスケジュールでの、固有索引の段階的な移行の管理を容易にします。ユーザー表の固有索引のための CREATE UNIQUE INDEX ステートメントを生成します。このツールの使用法の詳細については、概説およびインストール 資料のいずれかを参照してください。

許可

sysadm

必須接続

データベース。このコマンドは指定されたデータベースへの接続を自動的に確立します。

コマンド構文

```

▶▶ db2uidl -d database-name [-u table-schema] [-t table-name]
                                [-o filename] [-h]

```

コマンド・パラメーター

-d database-name

照会するデータベースの名前。

-u table-schema

処理する表のスキーマ (作成者ユーザー ID) を指定します。省略時アクションでは、すべてのユーザー ID によって作成された表を処理します。

-t table-name

処理する表の名前。省略時アクションでは、すべての表を処理します。

-o filename

出力を書き込むファイルの名前。省略時アクションでは、出力は標準出力に書き込まれます。

-h

ヘルプ情報を表示します。このオプションを指定すると、他のすべてのオプションは無視され、ヘルプ情報だけが表示されます。

使用上の注意

このツールは、DB2 バージョン 5 に移行したデータベース上でのみ使用できます。

注: このツールは、特定のタイプの名前を処理するようには設計されていません。特定の表名または表スキーマが、小文字、特殊文字、ブランクを含む区切り識別子である場合、すべての表またはスキーマの処理を要求したほうがよいでしょう。出力結果は編集できます。

db2untag - コンテナ・タグの解放

表スペース・コンテナの DB2 タグを除去します。このタグは、DB2 が複数の表スペースで 1 つのコンテナを再利用できないようにするために使用します。コンテナが関連しているデータベースを識別して、コンテナ・タグについての情報を表示します。削除されたデータベースが最後に使用したコンテナを解放する必要があるときに便利です。タグが残されている場合、DB2 がそれ以後その資源を使用しないようにします。

考慮事項: このツールは、経験のあるシステム管理者だけが使用してください。

許可

ユーザーには、データベースを作成した ID が所有する表スペースのコンテナへの読み取り / 書き込みアクセス権が必要です。

必須接続

なし

コマンド構文

```
▶▶—db2untag—-f—filename—————▶▶
```

コマンド・パラメーター

-f filename

DB2 タグが除去される、表スペース・コンテナの完全修飾名を指定します。

使用上の注意

データベースの作成、表スペースの作成または更新の操作から、SQLCODE -294 (使用中のコンテナでエラー) が戻されることがあります。通常それは、そのコンテナが別の表スペースによってすでに使用中であるために、オペレーティング・システム資源名の指定エラーであることを示します。コンテナは、一度に 1 つの表スペースでしか使用できません。

最後にコンテナを使用したデータベースが削除されていることをシステムまたはデータベース管理者が検出した場合、コンテナのタグが除去されていなければ、**db2untag** ツールを使用できます。そのコンテナを解放する場合は、以下のどちらかを行います。

db2untag - コンテナ・タグの解放

- SMS コンテナの場合、適切な削除コマンドを使用して、ディレクトリーとその内容を除去します。
- DMS ロー・コンテナの場合、ファイルまたは装置を削除するか、または **db2untag** でコンテナ・タグを除去します。ツールは、それ以外の点では DMS コンテナに何の修正も加えません。

以下も参照

235ページの『CREATE DATABASE』

第2章 コマンド行プロセッサ (CLP)

この章では、コマンド行プロセッサを呼び出して使用方法と CLP オプションについて説明します。

コマンド行プロセッサの呼び出し

コマンド行プロセッサの呼び出し

db2 コマンドはコマンド行プロセッサ (CLP) を開始します。CLP は、データベース・ユーティリティー、SQL ステートメントおよびオンライン・ヘルプを実行するために使用します。これにはさまざまなコマンド・オプションがあり、以下のモードで開始することができます。

- 対話式入力モード。特徴は **db2 =>** 入力プロンプトです。
- コマンド・モード。各コマンドの前に **db2** を付ける必要があります。
- バッチ・モード。 **-f** ファイル入力オプションを使用します。

注: Windows NT では、37ページの『db2cmd - DB2 コマンド・ウィンドウのオープン』が CLP 可能 DB2 ウィンドウをオープンし、DB2 コマンド行環境を初期化します。このコマンドを発行することは、「*DB2 コマンド・ウィンドウ (DB2 Command Window)*」アイコンをクリックすることと同じです。

495ページの『QUIT』はコマンド行プロセッサを停止します。586ページの『TERMINATE』もコマンド行プロセッサを停止しますが、関連するバックエンド・プロセスを除去し、使用されていたメモリーをすべて解放します。データベースが停止している場合やデータベース構成パラメーターを変更した場合は、TERMINATE の使用をお勧めします。

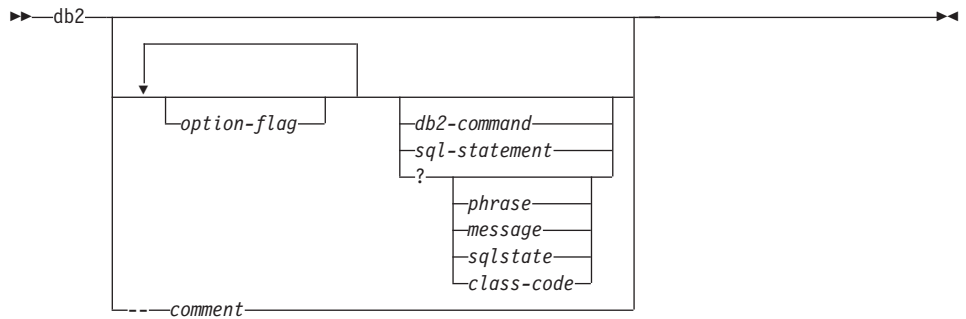
注: CLP を終了する前に、既存の接続をリセットしなければなりません。シェル・コマンド (!) を使うと、UNIX ベースのシステム、OS/2、または Windows オペレーティング・システム上で、オペレーティング・システムのコマンドを対話式またはバッチ・モードで実行できるようになります (たとえば、UNIX の場合 !ls、OS/2 または Windows オペレーティング・システムの場合 !dir)。

注: シェル・コマンド・サポートは Windows 3.1 では利用できません。

許可

なし

コマンド構文



コマンド・パラメーター

option-flag

有効な CLP オプション・フラグの要約については、125ページの表1を参照してください。

db2-command

DB2 コマンドを指定します。DB2 コマンドの説明については、143ページの『第3章 CLP コマンド』を参照してください。

sql-statement

SQL ステートメントを指定します。有効な 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。

? CLP 一般ヘルプを要求します。

? phrase

指定したコマンドまたはトピックに関連のあるヘルプ・テキストを要求します。データベース・マネージャーは、要求した情報を見つけれない場合、一般ヘルプ画面を表示します。

? options は、CLP オプションの説明と、現在の設定値を要求します。? help は、オンライン・ヘルプ構文図を読み取るための情報を要求します。

? message

有効な SQLCODE (たとえば、? sql10007n) により指定されているメッセージのヘルプを要求します。

? sqlstate

有効な SQLSTATE により指定されているメッセージのヘルプを要求します。

コマンド行プロセッサの呼び出し

? class-code

有効なクラス・コードによって指定されているメッセージのヘルプを要求します。

-- comment

コマンド行プロセッサは、注釈文字 `--` で始まる入力を注釈として扱います。

注: 各パラメーターについて、疑問符 (?) の後にスペースを入れて変数名から分ける必要があります。

コマンド行プロセッサのオプション

CLP コマンド・オプションは、コマンド行プロセッサ **DB2OPTIONS** 環境変数 (大文字) を設定するか、またはコマンド行フラグを使用することにより指定できます。

ユーザーは、**DB2OPTIONS** を使用して、セッション全体に適用されるオプションを設定することができます。

オプション・フラグの現在の設定および **DB2OPTIONS** の値を表示するには、367ページの『LIST COMMAND OPTIONS』を使用します。オプションの設定を対話式入力モードまたはコマンド・ファイルから変更するには、604ページの『UPDATE COMMAND OPTIONS』を使用します。

コマンド行プロセッサは、以下の手順でオプションを設定します。

1. 省略時オプションを設定します。
2. **DB2OPTIONS** を読み取って、省略時値を一時変更します。
3. コマンド行を読み取って、**DB2OPTIONS** を一時変更します。
4. UPDATE COMMAND OPTIONS からの入力を、最終的な対話式一時変更として受け入れます。

表1 に CLP オプション・フラグを要約してあります。これらのオプションは、どのような順序や組み合わせでも指定することができます。オプションをオンにするには、対応するオプション文字の前に負符号 (-) を付けます。オプションをオフにする場合、オプション文字の接頭部に負符号を付けオプション文字の後にも別の負符号を付けるか、またはオプション文字の接頭部に正符号 (+) を付けてください。たとえば、-c で auto-commit オプションはオンになり、-c- または +c でオフになります。これらのオプション文字では大文字と小文字は区別されず、-a と -A は同じものとみなされます。

表1. CLP コマンド・オプション

オプション・フラグ	説明	省略時設定
-a	コマンド行プロセッサは SQLCA データを表示します。	OFF
-c	コマンド行プロセッサは SQL ステートメントを自動的にコミットします。	ON
-e{cls}	コマンド行プロセッサは SQLCODE または SQLSTATE を表示します。この 2 つのオプションは相互に排他的です。	OFF

コマンド行プロセッサのオプション

表1. CLP コマンド・オプション (続き)

オプション・フラグ	説明	省略時設定
-filename	コマンド行プロセッサは、コマンド入力を標準入力ではなくファイルから読み取ります。	OFF
-lfilename	コマンド行プロセッサはコマンドを履歴ファイルに記録します。	OFF
-n	単一の区切りトークン内の改行文字を除去します。このオプションを指定しない場合、改行文字はスペースで置換されます。このオプションは、-t オプションと共に使用する必要があります。	OFF
-o	コマンド行プロセッサは、出力データおよびメッセージを標準出力に表示します。	ON
-p	コマンド行プロセッサは、対話型入力モードのときにコマンド行プロセッサ・プロンプトを表示します。	ON
-rfilename	コマンド行プロセッサは、コマンドが生成した報告をファイルに書き込みます。	OFF
-s	コマンド行プロセッサは、バッチ・ファイルまたは対話式モードでコマンドを実行中にエラーが発生した場合、実行を停止します。	OFF
-t	コマンド行プロセッサはセミコロン (;) をステートメント終了文字として使用します。	OFF
-tdx	コマンド行プロセッサは <i>x</i> をステートメント終了文字として定義し、使用します。	OFF
-v	コマンド行プロセッサはコマンド・テキストを標準出力にエコーさせます。	OFF
-w	コマンド行プロセッサは SQL ステートメント警告メッセージを表示します。	ON
-zfilename	コマンド行プロセッサはすべての出力の宛先をファイルに変更します。これは -r オプションと似ていますが、出力の他にメッセージやエラー・コードも含まれます。	OFF

例

AIX コマンド

```
export DB2OPTIONS='+a -c +ec -o -p'
```


を実行すると、このセッションの省略時設定は以下のようになります。

```
Display SQLCA   - off
Auto Commit     - on
Display SQLCODE - off
Display Output  - on
Display Prompt  - on
```

以下に、これらのオプションの詳細を説明します。

SQLCA データ表示オプション (-a):

DB2 コマンドまたは SQL ステートメントを実行した後で、SQLCA データを標準出力に表示します。SQLCA データが、エラーまたは完了メッセージの代わりに表示されます。

このコマンド・オプションの省略時設定は OFF です (+a または -a-)。

-o および -r オプションは -a オプションに影響します。詳細については、オプションの説明を参照してください。

自動コミット・オプション (-c):

このオプションは、各コマンドまたはステートメントを独立して処理するかどうかを指定します。ON (-c) に設定する場合、各コマンドまたはステートメントは自動的にコミットされるかまたはロールバックされます。コマンドまたはステートメントが正常に処理された場合、それ自体、およびそれ以前に自動コミット OFF (+c または -c-) で発行され、正常に実行されたコマンドおよびステートメントがすべてコミットされます。しかし、コマンドまたはステートメントが失敗した場合、それ自体、およびそれ以前に自動コミット OFF で発行され、正常に実行されたコマンドおよびステートメントはすべてロールバックされます。OFF (+c または -c-) に設定した場合、COMMIT または ROLLBACK を明示的に発行しなければなりません。そうしないと、次に自動コミット ON (-c) でコマンドを発行したときに、COMMIT または ROLLBACK のどちらかのアクションが起こります。

このコマンド・オプションの省略時設定は ON です。

自動コミット・オプションは、他のコマンド行プロセッサ・オプションに影響することはありません。

例: 以下のシナリオを考えてみましょう。

1. db2 create database test
2. db2 connect to test
3. db2 +c "create table a (c1 int)"

コマンド行プロセッサのオプション

4. db2 select c2 from a

ステップ 4 の SQL ステートメントは、表 A に C2 という名前の列がないので失敗します。そのステートメントは自動コミット ON (省略時値) で発行されたので、ステップ 4 のステートメントだけでなく、ステップ 3 のステートメントもロールバックされます。ステップ 3 が自動コミット OFF で発行されているためです。コマンド

```
db2 list tables
```

を実行しても、ヌルのリストが戻されます。

SQLCODE/SQLSTATE 表示オプション (-e):

-e{c|s} オプションを指定すると、コマンド行プロセッサは SQLCODE (-ec) または SQLSTATE (-es) を標準出力に表示します。オプション -ec および -es は、CLP 対話式モードでは無効です。

このコマンド・オプションの省略時設定は OFF です (+e または -e-)。

-o および -r オプションは、-e オプションに影響します。詳細については、オプションの説明を参照してください。

SQLCODE/SQLSTATE 表示オプションは他のコマンド行プロセッサ・オプションには影響しません。

例: AIX 上で実行しているコマンド行プロセッサから SQLCODE を検索するには、次のように入力します。

```
sqlcode='db2 -ec +o db2-command'
```

入力ファイルからの読み取りオプション (-f):

-filename オプションを指定すると、コマンド行プロセッサは標準入力ではなく指定したファイルから入力を読み取るようになります。*filename* は、指定ファイルまでのディレクトリー・パスを含む絶対または相対ファイル名です。ディレクトリー・パスを指定していない場合、現行ディレクトリーが使用されます。

他のオプションと一緒にオプション -f を指定する場合には、オプション -f は最後に指定する必要があります。たとえば、次のようにします。

```
db2 -tvf filename
```

注: このオプションは、対話式モード内では変更できません。

このコマンド・オプションの省略時設定は OFF です (+f または -f-)。

コマンド処理は 495ページの『QUIT』または 586ページの『TERMINATE』が出されるまで、またはファイルの終わりまで続行されます。

このオプションとデータベース・コマンドの両方を指定すると、コマンド行プロセッサはコマンドを全く処理せずにエラー・メッセージを戻します。

注釈文字 `--` で始まる入力ファイル行は、コマンド行プロセッサにより注釈として処理されます。行の最初の非空白文字は注釈文字でなければなりません。

`-filename` オプションを指定すると、`-p` オプションは無視されます。

入力ファイルからの読み取りオプションは、他のコマンド行プロセッサ・オプションには影響しません。

履歴ファイルへのコマンドのログ・オプション (-l):

`-lfilename` オプションを指定すると、コマンド行プロセッサは指定したファイルにコマンドのログを記録します。この履歴ファイルには、実行したコマンドとその完了状況が記録されます。`filename` は、指定ファイルまでのディレクトリー・パスを含む絶対または相対ファイル名です。ディレクトリー・パスを指定していない場合、現行ディレクトリーが使用されます。指定したファイルまたは省略時のファイルがすでに存在している場合、そのファイルに新しいログ項目が追加されます。

他のオプションと一緒にオプション `-l` を指定する場合には、オプション `-l` は最後に指定する必要があります。たとえば、次のようにします。

```
db2 -tv1 filename
```

このコマンド・オプションの省略時設定は OFF です (+1 または -1-)。

履歴ファイルへのコマンドのログ・オプションは、他のコマンド行プロセッサ・オプションには影響しません。

改行文字除去オプション (-n):

単一の区切りトークン内の改行文字を除去します。このオプションを指定しない場合、改行文字はスペースで置換されます。

注: このオプションは、対話式モード内では変更できません。

コマンド行プロセッサのオプション

このコマンド・オプションの省略時設定は OFF です (+n または -n-)。

このオプションは、-t オプションと共に使用する必要があります。詳細については、オプションの説明を参照してください。

出力表示オプション (-o):

-o オプションを指定すると、コマンド行プロセッサは出力データおよびメッセージを標準出力に送信します。

このコマンド・オプションの省略時設定は ON です。

対話式モード開始情報は、このオプションによって影響を受けることはありません。出力データは、ユーザー指定コマンドからの実行結果のレポート出力、および SQLCA データ (要求した場合) で構成されます。

以下のオプションは、+o オプションの影響を受けます。

- -rfilename: 対話式開始情報は保管されません。
- -e: +o を指定しても、SQLCODE または SQLSTATE が標準出力に表示されます。
- -a: +o を指定しても、影響を受けません。-a、+o および -rfilename を指定すると、SQLCA 情報がファイルに書き込まれます。

-o と -e オプションの両方を指定すると、データおよび SQLCODE または SQLSTATE のどちらかが画面に表示されます。

-o と -v オプションの両方を指定すると、データが表示され、発行した各コマンドのテキストが画面にエコーされます。

出力表示オプションは、他のコマンド行プロセッサ・オプションには影響しません。

DB2 対話式プロンプト表示オプション (-p):

-p オプションを指定すると、ユーザーが対話式モードになっているときに、コマンド行プロセッサはコマンド行プロセッサ・プロンプトを表示します。

このコマンド・オプションの省略時設定は ON です。

コマンドがコマンド行プロセッサにパイプ接続されているときには、プロンプトをオフにすると便利です。たとえば、CLP コマンドを含むファイルを、次のコマンドを発行することによって実行できます。

```
db2 +p < myfile.clp
```

-p オプションは、-filename オプションを指定すると無視されます。

DB2 対話式プロンプト表示オプションは、他のコマンド行プロセッサ・オプションには影響しません。

報告ファイルへの保管オプション (-r):

`-rfilename` オプションは、コマンドによって生成される出力データをすべて指定したファイルに書き込まれるようにします。これは、キャプチャーしないと画面がスクロールして見えなくなってしまう報告のキャプチャーに便利です。メッセージまたはエラー・コードはファイルに書き込まれません。`filename` は、指定ファイルまでのディレクトリー・パスを含む絶対または相対ファイル名です。ディレクトリー・パスを指定していない場合、現行ディレクトリーが使用されます。新しい報告項目がファイルに追加されます。

このコマンド・オプションの省略時設定は OFF です (+r または -r-)。

`-a` オプションを指定すると、SQLCA データがファイルに書き込まれます。

`-r` オプションは `-e` オプションには影響しません。`-e` オプションを指定すると、SQLCODE または SQLSTATE がファイルではなく標準出力に書き込まれます。

`-rfilename` を **DB2OPTIONS** に設定すると、ユーザーはコマンド行から +r (または -r-) オプションを設定して、特定のコマンド呼び出しの出力データがファイルに書き込まれないようにすることができます。

報告ファイルへの保管オプションは、他のコマンド行プロセッサ・オプションには影響しません。

コマンド・エラー時の実行の停止オプション (-s):

コマンドが対話式モードでまたは入力ファイルから発行され、構文またはコマンド・エラーが起きた場合、`-s` オプションを指定してあると、コマンド行プロセッサが実行を停止して、標準出力にエラー・メッセージが書き込まれます。

このコマンド・オプションの省略時設定は OFF です (+s または -s-)。この設定により、コマンド行プロセッサはエラー・メッセージを表示し、残りのコマンドの実行を続け、システム・エラーが発生したとき (戻りコード 8) に限って実行を停止します。

コマンド行プロセッサのオプション

次の表に、この動作を要約します。

表2. CLP 戻りコードおよびコマンド実行

戻りコード	-s オプション設定	+s オプション設定
0 (成功)	実行の継続	実行の継続
1 (0 行が選択された)	実行の継続	実行の継続
2 (警告)	実行の継続	実行の継続
4 (DB2 または SQL エラー)	実行の停止	実行の継続
8 (システム・エラー)	実行の停止	実行の停止

ステートメント終了文字オプション (-t):

-t オプションを指定すると、コマンド行プロセッサはステートメント終了文字としてセミコロン (;) を使用するようになり、スラッシュ (/) 行継続文字は使用不能になります。

注: このオプションは、対話式モード内では変更できません。

このコマンド・オプションの省略時設定は OFF です (+t または -t-)。

終了文字を定義するには、-td の後に選択した終了文字を指定します。たとえば、-tdx とすると x がステートメント終了文字として設定されます。

終了記号かどうか調べられるのが各入力行の最後の非空白文字だけであるため、コマンド行から複数のステートメントを連結するために終了文字を使用することはできません。

ステートメント終了文字オプションは、他のコマンド行プロセッサ・オプションには影響しません。

冗長出力オプション (-v):

-v オプションを指定すると、コマンド行プロセッサはコマンドからの出力またはメッセージを表示する前に、ユーザーが入力したコマンド・テキストを (標準出力に) エコーします。256ページの『ECHO』にはこのオプションは無効です。

このコマンド・オプションの省略時設定は OFF です (+v または -v-)。

+o (または -o-) が指定された場合、-v オプションは無効になります。

冗長出力オプションは、他のコマンド行プロセッサ・オプションには影響しません。

警告メッセージ表示オプション (-w):

-w オプションを指定すると、コマンド行プロセッサは SQL ステートメント警告メッセージを表示します。

このコマンド・オプションの省略時設定は ON です。

すべての出力のファイルへの保管オプション (-z):

-z*filename* オプションは、コマンドによって生成される出力データをすべて指定したファイルに書き込まれるようにします。これは、キャプチャーしないと画面がスクロールして見えなくなってしまう報告のキャプチャーに便利です。これは、-r オプションに似ていますが、メッセージ、エラー・コード、および情報出力もファイルに書き込まれる点異なります。 *filename* は、指定ファイルまでのディレクトリー・パスを含む絶対または相対ファイル名です。ディレクトリー・パスを指定していない場合、現行ディレクトリーが使用されます。新しい報告項目がファイルに追加されます。

このコマンド・オプションの省略時設定は OFF です (+z または -z-)。

-a オプションを指定すると、SQLCA データがファイルに書き込まれます。

-z オプションは -e オプションには影響しません。 -e オプションを指定すると、SQLCODE または SQLSTATE がファイルではなく標準出力に書き込まれます。

-z*filename* を **DB2OPTIONS** に設定すると、ユーザーはコマンド行から +z (または -z-) オプションを設定して、特定のコマンド呼び出しの出力データがファイルに書き込まれないようにすることができます。

すべての出力のファイルへの保管オプションは、他のコマンド行プロセッサ・オプションには影響しません。

コマンド行プロセッサの戻りコード

コマンド行プロセッサはコマンドの処理または SQL ステートメントの処理を終了すると、終了 (または戻り) コードを戻します。これらのコードは、このコマンド行から CLP 関数を実行しているユーザーには意識されませんが、これらの関数をシェル・スクリプトから実行した場合は、コードを検索することができます。

コマンド行プロセッサの戻りコード

たとえば、次の B シェル・スクリプトは、GET DATABASE MANAGER CONFIGURATION コマンドを実行してから、CLP 戻りコードを検査します。

```
db2 get database manager configuration
if ["$?" = "0"]
then echo "OK!"
fi
```

戻りコードは以下のいずれかになります。

コード 説明

- 0 DB2 コマンドまたは SQL ステートメントが正常に実行されました。
- 1 SELECT または FETCH ステートメントが行を戻しませんでした。
- 2 DB2 コマンドまたは SQL ステートメントからの警告です。
- 4 DB2 コマンドまたは SQL ステートメント・エラーです。
- 8 コマンド行プロセッサのシステム・エラーです。

ユーザーが対話式モードでステートメントを実行しているとき、または入力 (-f オプションを使用して) ファイルから読み込んでいるときは、コマンド行プロセッサは戻りコードを戻しません。

戻りコードが利用可能なのは、ユーザーが対話式モードを終了した後、または入力ファイルの処理が終了したときだけです。この場合、戻りコードは、その時点までに実行された個々のコマンドまたはステートメントから戻された個別のコードの論理和になります。

たとえば、ユーザーが対話式モードでコマンドを発行した結果が戻りコード 0、1、および 2 になった場合、戻りコード 3 はユーザーが対話式モードを終了した後で戻されます。個々のコード 0、1、および 2 は戻されません。戻りコード 3 によって、対話式モード処理中に、1 つまたは複数のコマンドが 1 を戻し、1 つまたは複数のコマンドが 2 を戻したことがユーザーに通知されます。

戻りコード 4 になるのは、DB2 コマンドまたは SQL ステートメントが負の SQLCODE を戻したときです。戻りコード 8 になるのは、コマンド行プロセッサでシステム・エラーが起きたときだけです。

コマンドが入力ファイルからまたは対話式モードで発行され、かつコマンド行プロセッサでシステム・エラーが起きた場合 (戻りコード 8)、コマンドの実行は直ちに停止されます。1 つまたは複数の DB2 コマンドまたは SQL ステートメントがエラーで終了した場合 (戻りコード 4)、-s (コマンド・エラー時

の実行の停止) オプションが設定されていると、コマンドの実行が停止します。このオプションが設定されていない場合は、コマンドの実行は継続します。

コマンド行プロセッサの使用

コマンド行プロセッサは以下の順序で動作します。

- CLP コマンド (大文字または小文字) をコマンド・プロンプトに入力します。
- ENTER キーを押すと、コマンドがコマンド・シェルに送られます。
- 出力の宛先が標準出力装置に自動的に指定されます。
- パイピングとリダイレクトがサポートされます。
- 処理の成功または失敗が、ユーザーに通知されます。
- コマンドの実行の後に、制御がオペレーティング・システム・コマンド・プロンプトに戻され、ユーザーは続けて次のコマンドを入力することができるようになります。

データベースにアクセスする前に、ユーザーは 576ページの『START DATABASE MANAGER』で DB2 を始動するなど、準備作業のタスクを実行しなければなりません。さらに、データベースを照会する前に、ユーザーはそのデータベースに接続していなければなりません。データベースに接続するには、以下のいずれかを行います。

- SQL CONNECT TO *database* ステートメントを発行します。
- 環境変数 **DB2DBDFT** により定義されている省略時データベースに暗黙接続を確立します。

データベース内の表での作業に関する情報については、*SQL 解説書* を参照してください。

コマンドの文字数がコマンド・プロンプトに許可されている文字制限を超える場合、円記号 (¥) を行継続文字として使用することができます。コマンド行プロセッサは行継続文字を検出すると、次の行を読み取って、その行にある文字を前の行の文字に連結します。その代わりに、**-t** オプションを使用して行終了文字を設定することもできます (132ページを参照)。この場合、行継続文字は無効になり、すべてのステートメントとコマンドの最後を必ず行終了文字にしなければなりません。

コマンド行プロセッサは、NULL と呼ばれるストリングをヌル文字列として識別します。以前に何らかの値に設定したフィールドを後でヌル値に設定することができます。たとえば、次のように指定します。

コマンド行プロセッサの使用

```
db2 update database manager configuration using tm_database NULL
```

この場合、`tm_database` フィールドがヌル値に設定されます。この操作は大文字小文字を区別します。小文字の `null` はヌル文字列としては解釈されず、`null` を含む文字列として解釈されます。

コマンド・ファイルでのコマンド行プロセッサの使用

データベース・マネージャーへの CLP 要求をシェル・スクリプトのコマンド・ファイルに組み込むことができます。次の例では、シェル・スクリプト・コマンド・ファイルに `CREATE TABLE` ステートメントを入力する方法を示しています。

```
db2 "create table mytable (name VARCHAR(20), color CHAR(10))"
```

コマンドおよびコマンド・ファイルの詳細については、適切なオペレーティング・システムの解説書を参照してください。

コマンド行プロセッサの設計

コマンド行プロセッサは 2 つのプロセス、つまりユーザー・インターフェースとして機能するフロントエンド・プロセス (DB2 コマンド) と、データベース接続を維持するバック・エンド・プロセス (db2bp) とで構成されています。

データベース接続の保守

db2 が呼び出されるたびに、新しいフロントエンド・プロセスが開始されます。バック・エンド・プロセスは、最初の **db2** 呼び出しのときに開始し、586ページの『TERMINATE』によって明示的に終了します。親が同じであるフロントエンド・プロセスはすべて単一のバック・エンド・プロセスにより保守されているので、単一のデータベース接続を共有していることとなります。

たとえば以下のように、同じオペレーティング・システムのコマンド・プロンプトから複数の **db2** 呼び出しを行うと、1 つのバック・エンド・プロセスを共有する複数のフロントエンド・プロセスが個別に開始します。そしてこのバック・エンド・プロセスがデータベース接続を保持します。

- `db2 'connect to sample'`,
- `db2 'select * from org'`,
- `. foo` (foo は DB2 コマンドを含むシェル・スクリプト)
- `db2 -tf myfile.clp`

以下のように、同じオペレーティング・システム・プロンプトから複数の呼び出しを行うと、それぞれの呼び出しの親プロセスが異なるために個別のバック・エンド・プロセスが必要になり、結果として複数のデータベース接続が個別に開始してしまいます。

- foo
- . foo &
- foo &
- sh foo

フロントエンド・プロセスとバック・エンド・プロセスとの間の通信

フロントエンド・プロセスとバック・エンド・プロセスとの間の通信は、要求待ち行列、入力待ち行列、および出力待ち行列の 3 つのメッセージ待ち行列を介して行います。

環境変数

以下の環境変数によって、2 つのプロセス間での通信が構成されます。

表 3. 環境変数

変数	最小	最大	省略時値
DB2BQTIME	1 秒	5294967295	1 秒
DB2BQTRY	試行 0 回	5294967295	試行 60 回
DB2RQTIME	1 秒	5294967295	5 秒
DB2IQTIME	1 秒	5294967295	5 秒

DB2BQTIME

コマンド行プロセッサを呼び出すと、フロントエンド・プロセスは、バック・エンド・プロセスがすでに活動しているかどうかを調べます。活動している場合、フロントエンド・プロセスは接続を再び確立します。活動していない場合、フロントエンド・プロセスがバック・エンド・プロセスを活動化します。フロントエンド・プロセスは、次に **DB2BQTIME** 変数に指定された期間、活動を停止し、その後もう一度検査します。フロントエンド・プロセスは、 **DB2BQTRY** 変数に指定した回数だけ検査を継続し、それでもバック・エンド・プロセスが活動していない場合は、時間切れとなりエラー・メッセージを戻します。

コマンド行プロセッサの設計

DB2BQTRY

DB2BQTIME 変数と連携しており、バック・エンド・プロセスが活動しているかどうかの判別をフロントエンド・プロセスが試行する回数を指定します。

DB2BQTIME および **DB2BQTRY** の値は、ピーク時に照会時間を最適化するために増やすことができます。

DB2RQTIME

一度バック・エンド・プロセスが開始すると、このプロセスはフロントエンドからの要求があるまで要求待ち行列で待機します。さらに、コマンド・プロンプトから要求が開始されてから次の要求が開始されるまでの間も、要求待ち行列で待機します。

DB2RQTIME 変数は、バック・エンド・プロセスがフロントエンド・プロセスからの要求を待機する長さを指定します。この時間が経過すると、要求待ち行列に要求がない場合、バック・エンド・プロセスはフロントエンド・プロセスの親がまだ存在しているかを調べ、存在していないことが分かるとバック・エンド・プロセスは終了します。親が存在している場合は、要求待ち行列で待機を継続します。

DB2IQTIME

バック・エンド・プロセスがフロントエンド・プロセスから要求を受信すると、バック・エンド・プロセスはフロントエンド・プロセスに肯定応答を送り、入力待ち行列を介して入力を受け取る準備ができていることを知らせます。そして、バック・エンド・プロセスは入力待ち行列で待機します。さらに、バッチ・ファイル (-f オプションで指定) が実行している間、またユーザーが対話式モードの間も、バック・エンド・プロセスは入力待ち行列で待機します。

DB2IQTIME 変数は、フロントエンド・プロセスがコマンドを渡すまで、バック・エンド・プロセスが待機する時間を指定します。指定した時間が経過すると、バック・エンド・プロセスは、フロントエンド・プロセスが活動しているかどうかを調べ、フロントエンド・プロセスがすでに存在していない場合は、要求待ち行列に戻って待機します。フロントエンド・プロセスが存在している場合は、バック・エンド・プロセスはフロントエンド・プロセスからの入力を待機します。

これらの環境変数の値を表示するには、367ページの『LIST COMMAND OPTIONS』を使用します。

バック・エンド環境変数は、バック・エンド・プロセスが開始されるときにフロントエンド・プロセスから値の集合を継承します。ただし、フロントエンド

環境変数に変更された場合は、バック・エンド・プロセスはその変更を継承することはありません。変更した値を継承するには、まずバック・エンド・プロセスを終了してから、再始動 (**db2** コマンドを発行する) しなければなりません。

バック・エンド・プロセスを終了しなければならない場合の例を、次のシナリオで示します。

1. ユーザー A がログオンし、いくつか CLP コマンドを出した後、**TERMINATE** を発行せずにログオフします。
2. ユーザー B が同じウィンドウを使用してログオンします。
3. ユーザー B が特定の CLP コマンドを発行すると、コマンドはメッセージ **DB21016** (システム・エラー) を出して失敗します。

ユーザー B のフロントエンド・プロセス (コマンドが出されたオペレーティング・システム・ウィンドウ) の親が依然として活動状態にあるため、ユーザー A が開始したバック・エンド・プロセスは、ユーザー B が CLP を使用して開始した時点でもまだ活動しています。バック・エンド・プロセスは、ユーザー B が発行した新規コマンドを処理しようとしませんが、ユーザー B のフロントエンド・プロセスには、バック・エンド・プロセスのメッセージ待ち行列を使用するだけの十分な権限がありません。そのためにはバック・エンド・プロセスを作成したユーザー A の権限が必要だからです。CLP セッションを **TERMINATE** コマンドを使用して終了してから、ユーザーは同じオペレーティング・システム・ウィンドウを使用して、新規 CLP セッションを開始しなければなりません。これによって新規ユーザーごとに新しいバック・エンド・プロセスが作成されることになり、権限の問題は回避され、新規ユーザーのバック・エンド・プロセス内で環境変数 (**DB2INSTANCE** など) の値を訂正して設定できます。

CLP 使用上の注意

コマンドはコマンド・プロンプトから大文字または小文字で入力できます。ただし、**DB2** が大文字小文字を区別するパラメーターには、大文字小文字を正確に入力しなければなりません。たとえば、**CHANGE DATABASE COMMENT** の **WITH** 文節にある *comment-string* は、大文字小文字を区別するパラメーターです。

区切り付き識別子は、SQL ステートメント内で使用できます。SQL ステートメント内での区切り識別子の使用方法の詳細については、*SQL 解説書* を参照してください。

CLP 使用上の注意

特殊文字またはメタ文字 (たとえば、\$ & * () ; < > ? ¥ ' " など) は、CLP コマンド内で使用できます。これらの文字が CLP 対話式モード以外または CLP バッチ入力モード以外で使用されている場合、これらの文字はオペレーティング・システム・シェルによって解釈されます。シェルが特殊な処置を行わない場合は、疑問符またはエスケープ文字が必要です。

たとえば、AIX Korn シェル環境でコマンド

```
db2 select * from org where division > 'Eastern'
```

を実行すると、"select <the names of all files> from org where division" と解釈されます。この結果の SQL 構文エラーは、ファイル Eastern にリダイレクトされます。次の構文であれば、正しい出力が生成されます。

```
db2 "select * from org where division > 'Eastern'"
```

特殊文字は、プラットフォームごとに異なります。AIX Korn シェルでは、上記の例はエスケープ文字 (¥)、たとえば ¥*、 ¥>、または ¥' などを使って書き直すことができます。OS/2 シェルでは、¥* または ¥' とすると構文エラーになります。

コマンドを MS-DOS プロンプトからコマンド行モードで入力すると、Windows 95 DOS コマンド・プロセッサは、パーセント記号 (%) で囲んで入力したものは、すべて変数名として解釈するように前処理します。たとえば、次のようにします。

```
db2 "SELECT NAME, CREATOR FROM SYSIBM.SYSPLAN WHERE NAME LIKE 'SQL%' OR NAME LIKE 'SQLC%'"
```

この場合、db2.exe に渡されるのは以下のようになります。

```
"SELECT NAME,CREATOR FROM SYSIBM.SYSPLAN WHERE NAME LIKE 'SQL%' ('%' OR NAME LIKE 'SQLC%' は DOS によって値がヌルの変数として扱われます。
```

この振る舞いを避けるには、以下のように二重パーセント記号 (%%) を使用して、パーセント記号 (変数ではない) をプログラムに渡すように指示します。

```
db2 "SELECT NAME, CREATOR FROM SYSIBM.SYSPLAN WHERE NAME LIKE 'SQL%%%' OR NAME LIKE 'SQLC%%'"
```

これは、db2.exe に以下のように渡されます。

```
"SELECT NAME, CREATOR FROM SYSIBM.SYSPLAN WHERE NAME LIKE 'SQL%%%' OR NAME LIKE 'SQLC%%'"
```

ほとんどのオペレーティング・システム環境では、入力と出力をリダイレクトすることが可能です。たとえば、SAMPLE データベースに接続している場

合、次に示す要求は STAFF 表を照会し、出力を mydata ディレクトリーにある staflist.txt という名前のファイルに送ります。

```
db2 "select * from staff" > mydata/staflist.txt
```

出力のリダイレクトがサポートされていない Windows 3.1 などの環境では、CLP オプションを使用することができます。たとえば、上記の要求は以下のよう書き直すことができます。

```
db2 -r mydata%staflist.txt "select * from staff"
```

```
db2 -z mydata%staflist.txt "select * from staff"
```

Windows 用の CLP オプションの詳細については、インストールおよび構成補足を参照してください。

コマンド行プロセッサはプログラミング言語ではありません。たとえば、ホスト変数およびステートメントはサポートしません。そのため、

```
db2 connect to :HostVar in share mode
```

このステートメント例は、:HostVar が有効なデータベース名ではないため、構文上は正しくありません。

コマンド行プロセッサは、SQL NULL 値をハイフン (-) で表します。列が数値である場合、ハイフンは列の一番右に置きます。列が数値ではない場合、ハイフンは列の一番左に置きます。DB2 コネクトとホスト・データベースを使っている場合のコマンド行プロセッサの使用の詳細については、DB2 コネクト 使用者の手引き を参照してください。

CLP 使用上の注意

第3章 CLP コマンド

この章では、DB2 コマンドをアルファベット順に説明します。これらのコマンドを使用して、システムを対話式に制御できます。

注: ディレクトリー・パス中の斜線 (/) は UNIX ベースのシステムだけに用いられるもので、OS/2 および Windows オペレーティング・システムの円記号 (¥) に相当します。

コマンドの記述および編成の詳細については、2ページの『コマンドの説明の編成方法』を参照してください。

DB2 CLP コマンド

次の表では、CLP コマンドを機能別カテゴリーに分けて示しています。

表 4. DB2 CLP コマンド

CLP セッション制御
367ページの『LIST COMMAND OPTIONS』
604ページの『UPDATE COMMAND OPTIONS』
232ページの『CHANGE ISOLATION LEVEL』
570ページの『SET RUNTIME DEGREE』
586ページの『TERMINATE』
495ページの『QUIT』
データベース・マネージャー制御
576ページの『START DATABASE MANAGER』
582ページの『STOP DATABASE MANAGER』
292ページの『GET DATABASE MANAGER CONFIGURATION』
533ページの『RESET DATABASE MANAGER CONFIGURATION』
608ページの『UPDATE DATABASE MANAGER CONFIGURATION』
269ページの『GET ADMIN CONFIGURATION』
528ページの『RESET ADMIN CONFIGURATION』
599ページの『UPDATE ADMIN CONFIGURATION』
データベース制御
537ページの『RESTART DATABASE』

DB2 CLP コマンド

表4. DB2 CLP コマンド (続き)

235ページの『CREATE DATABASE』
251ページの『DROP DATABASE』
455ページの『MIGRATE DATABASE』
386ページの『LIST INDOUBT TRANSACTIONS』
148ページの『ACTIVATE DATABASE』
243ページの『DEACTIVATE DATABASE』
380ページの『LIST DRDA INDOUBT TRANSACTIONS』
279ページの『GET DATABASE CONFIGURATION』
531ページの『RESET DATABASE CONFIGURATION』
606ページの『UPDATE DATABASE CONFIGURATION』
データベース・ディレクトリー管理
194ページの『CATALOG DATABASE』
587ページの『UNCATALOG DATABASE』
199ページの『CATALOG DCS DATABASE』
589ページの『UNCATALOG DCS DATABASE』
230ページの『CHANGE DATABASE COMMENT』
369ページの『LIST DATABASE DIRECTORY』
378ページの『LIST DCS DIRECTORY』
202ページの『CATALOG GLOBAL DATABASE』
ODBC 管理
224ページの『CATALOG ODBC DATA SOURCE』
398ページの『LIST ODBC DATA SOURCES』
597ページの『UNCATALOG ODBC DATA SOURCE』
274ページの『GET CLI CONFIGURATION』
602ページの『UPDATE CLI CONFIGURATION』
クライアント / サーバー・ディレクトリー管理
215ページの『CATALOG LOCAL NODE』
218ページの『CATALOG NAMED PIPE NODE』
186ページの『CATALOG APPC NODE』
189ページの『CATALOG APPCLU NODE』
191ページの『CATALOG APPN NODE』
204ページの『CATALOG IPX/SPX NODE』
221ページの『CATALOG NETBIOS NODE』

表 4. DB2 CLP コマンド (続き)

226ページの『CATALOG TCP/IP NODE』
595ページの『UNCATALOG NODE』
391ページの『LIST NODE DIRECTORY』
ネットワーク・サポート
511ページの『REGISTER』
245ページの『DEREGISTER』
610ページの『UPDATE LDAP NODE』
208ページの『CATALOG LDAP DATABASE』
591ページの『UNCATALOG LDAP DATABASE』
213ページの『CATALOG LDAP NODE』
593ページの『UNCATALOG LDAP NODE』
509ページの『REFRESH LDAP』
回復
157ページの『BACKUP DATABASE』
500ページの『RECONCILE』
539ページの『RESTORE DATABASE』
550ページの『ROLLFORWARD DATABASE』
382ページの『LIST HISTORY』
487ページの『PRUNE HISTORY/LOGFILE』
615ページの『UPDATE HISTORY FILE』
359ページの『INITIALIZE TAPE』
549ページの『REWIND TAPE』
575ページの『SET TAPE POSITION』
操作ユーティリティー
267ページの『FORCE APPLICATION』
400ページの『LIST PACKAGES/TABLES』
520ページの『REORGCHK』
517ページの『REORGANIZE TABLE』
561ページの『RUNSTATS』
データベース・モニター
310ページの『GET MONITOR SWITCHES』
613ページの『UPDATE MONITOR SWITCHES』
307ページの『GET DATABASE MANAGER MONITOR SWITCHES』

DB2 CLP コマンド

表4. DB2 CLP コマンド (続き)

312ページの『GET SNAPSHOT』
535ページの『RESET MONITOR』
362ページの『LIST ACTIVE DATABASES』
364ページの『LIST APPLICATIONS』
374ページの『LIST DCS APPLICATIONS』
データ・ユーティリティー
257ページの『EXPORT』
332ページの『IMPORT』
411ページの『LOAD』
453ページの『LOAD QUERY』
アプリケーションの準備
457ページの『PRECOMPILE PROGRAM』
165ページの『BIND』
496ページの『REBIND』
リモート・サーバー・ユーティリティー
154ページの『ATTACH』
250ページの『DETACH』
表スペース管理
403ページの『LIST TABLESPACE CONTAINERS』
572ページの『SET TABLESPACE CONTAINERS』
405ページの『LIST TABLESPACES』
491ページの『QUIESCE TABLESPACES FOR TABLE』
ノード管理
152ページの『ADD NODE』
254ページの『DROP NODE VERIFY』
397ページの『LIST NODES』
ノードグループ管理
394ページの『LIST NODEGROUPS』
505ページの『REDISTRIBUTE NODEGROUP』
追加コマンド
150ページの『ADD DATALINKS MANAGER』
184ページの『CALL』
247ページの『DESCRIBE』

表 4. DB2 CLP コマンド (続き)

256ページの『ECHO』
272ページの『GET AUTHORIZATIONS』
277ページの『GET CONNECTION STATE』
309ページの『GET INSTANCE』
330ページの『HELP』
360ページの『INVOKE STORED PROCEDURE』
373ページの『LIST DATALINKS MANAGERS』
489ページの『QUERY CLIENT』
566ページの『SET CLIENT』

ACTIVATE DATABASE

ACTIVATE DATABASE

指定したデータベースを活動化し、必要なデータベース・サービスをすべて開始することにより、どのアプリケーションからでもそのデータベースに接続して使用できるようにします。

効力範囲

このコマンドは、システム内のすべてのノードで、指定したデータベースを活動化します。データベースを活動化中に 1 つまたは複数のノードがエラーを検出すると、警告が戻されます。コマンドが正常に処理されたすべてのノードで、データベースは活動状態を維持します。

許可

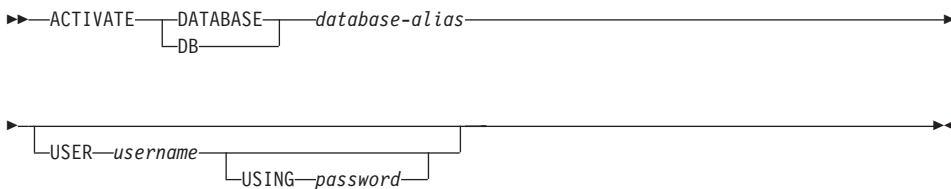
以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

なし

コマンド構文



コマンド・パラメーター

database-alias

開始するデータベースの別名を指定します。

USER username

データベースを開始するユーザーを指定します。

USING password

ユーザー名のパスワードを指定します。

使用上の注意

データベースが開始していないときに、アプリケーションで `CONNECT TO` (または暗黙接続) が発行された場合、アプリケーションは、データベースに対して作業する前に、必要なデータベースをデータベース・マネージャーが開始する間待機しなければなりません。しかし、データベースが開始してしまえば、他のアプリケーションはデータベースの開始を待たずに、ただデータベースに接続するだけでその中のデータを使用することができます。

データベース管理者は、選択したデータベースを開始する際に `ACTIVATE DATABASE` を使用することができます。そうしておけば、データベースの初期化に要するアプリケーション時間を節約することができます。

`ACTIVATE DATABASE` で初期化したデータベースは、243ページの『`DEACTIVATE DATABASE`』 または 582ページの『`STOP DATABASE MANAGER`』 によって遮断することができます。

`CONNECT TO` (または暗黙接続) によって開始したデータベースに続けて `ACTIVATE DATABASE` を発行した場合、そのデータベースを遮断するには `DEACTIVATE DATABASE` を使用しなければなりません。 `ACTIVATE DATABASE` を使用しないでデータベースを始動した場合、最後のアプリケーションが切断した時にデータベースは遮断されます。

`ACTIVATE DATABASE` コマンドは、再始動が必要なデータベース (たとえば、不整合状態にあるデータベース) を処理する場面においては、 `CONNECT TO` (または暗黙接続) と同じように機能します。データベースを再始動してからでないと、 `ACTIVATE DATABASE` でそのデータベースを初期化できません。再始動を実行できるのは、データベースが `AUTORESTART ON` で構成されている場合だけです。

注: `ACTIVATE DATABASE` コマンドを発行するアプリケーションは、どのデータベースへの活動データベース接続も持つことができません。

以下も参照

243ページの『`DEACTIVATE DATABASE`』

582ページの『`STOP DATABASE MANAGER`』

ADD DATALINKS MANAGER

ADD DATALINKS MANAGER

DB2 データ・リンク・マネージャーを、指定されたデータベースに登録される DB2 データ・リンク・マネージャーのリストに追加します。

許可

なし

コマンド構文

```
➤➤➤ ADD DATALINKS MANAGER FOR DATABASE dbname USING DB NODE hostname PORT port-number CELL DFS-cellname DLMINSTANCE instance-name ▶▶▶
```

コマンド・パラメーター

DATABASE *dbname*

データベース名を指定します。

USING NODE *hostname*

DB2 データ・リンク・マネージャー・サーバーの、完全に修飾されたホスト名、または IP アドレス (両方ではない) を指定します。

PORT *port-number*

DB2 サーバーから DB2 データ・リンク・マネージャー・サーバーへの通信用に予約されているポート番号を指定します。

CELL *DFS-cellname*

DFS セルの完全修飾名を指定します。たとえば、`dl1n1.almaden.ibm.com` のようにします。

注: このパラメーターでは、1 データベースに 1 つのセルしか登録できません。

DLMINSTANCE *instance-name*

セルでデータ・リンク・マネージャーを実行するインスタンス名を指定します。

使用上の注意

USING NODE を指定して追加されたデータ・リンク・マネージャーのタイプは「ネイティブ」で、USING CELL を指定して追加されたデータ・リンク・

マネージャーのタイプは "DFS" になります。データベースに登録されるすべてのデータ・リンク・マネージャーは同じタイプでなければならず、タイプ "DFS" のデータ・リンク・マネージャーは 1 データベースに 1 つしか登録できません。

このコマンドを使用して 1 つまたは複数の DB2 データ・リンク・マネージャーをデータベースに登録する場合には、DB2 データ・リンク・マネージャーが 2 度登録されていないことを確認してください。2 度登録されていると、データ・リンク処理中に、エラー SQL20056N と理由コード "99" が共に戻される可能性があります。2 度登録された DB2 データ・リンク・マネージャー・サーバーの db2diag.log ファイルには、このような障害が発生した場合には以下の項目が入ります。

```
dfm_xnstate_cache_insert : Duplicate txn entry.  
dfmBeginTxn : Unable to insert ACTIVE transaction in cache, rc = 41.  
DLFM501E : Transaction management service failed.
```

注: 同じ名前またはアドレスを使用する、重複したデータ・リンク・マネージャーが追加される場合には、コマンド行プロセッサはエラーを検出します。ただし、異なる IP 名またはアドレスを使用して複数回データ・リンク・マネージャーが追加される場合は、重複は検出されません。たとえば、データ・リンク・マネージャーが 2 度追加されており、最初は dln1.almaden.ibm.com という名前、次は dln1 という短い名前を使用しているとすると、上記のような障害が発生する可能性があります。

以下も参照

373ページの『LIST DATALINKS MANAGERS』

ADD NODE

ADD NODE

新規ノードを並列データベース・システムに追加します。このコマンドは、新規ノードの MPP サーバーで現在定義済みのデータベースすべてにデータベース区画を作成します。ユーザーは、データベースとともに作成するシステム一時表スペース用のソース・ノードを指定できます。またはシステム一時表スペースを作成しないように指定することもできます。このコマンドは追加中のノードから発行する必要があり、MPP サーバー上でのみ発行できます。

効力範囲

このコマンドは、それが実行されたノードに対してだけ影響を与えます。

許可

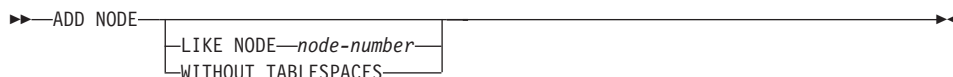
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし

コマンド構文



コマンド・パラメーター

LIKE NODE node-number

システム一時表スペース用のコンテナが、インスタンス内の各データベース用に指定した *node-number* のコンテナと同一になるように指定します。指定するノードは、db2nodes.cfg ファイル中にすでに指定してあるノードでなければなりません。

WITHOUT TABLESPACES

システム一時表スペースのコンテナがどのデータベースに対しても作成されないことを指定します。データベースを使用する前に、ALTER TABLESPACE ステートメントを使用して、システム一時表スペース・コンテナを各データベースに追加しなければなりません。

注: オプションを指定しない場合、システム一時表スペース用のコンテナは各データベースのカatalog・ノード上のコンテナと同じになります。そのCatalog・ノードは、MPP システム内の各データベース用の別のノードである場合があります。

使用上の注意

新規ノードを追加する前に、システム上のすべての既存データベース用にコンテナを作成するだけの十分な記憶域があることを確認してください。

ノード追加操作は、インスタンス中に存在する各データベースの新規ノードに、ヌル・データベース区画を作成します。新規データベース区画の構成パラメーターは、省略時値に設定されます。

データベース区画をローカルで作成中にノード追加操作が失敗すると、この操作は終結処理フェーズに入り、すでに作成されているすべてのデータベースをローカルに消去します。これは、追加中のノード（つまり、ローカル・ノード）からのみデータベース区画が削除されることを意味しています。他のすべてのノードにある既存データベース区画は、影響を受けません。これも失敗すると、終結処理は停止し、エラーが戻されます。

ALTER NODEGROUP ステートメントを使用してノードをノードグループに追加した後まで、新規ノード上のデータベース区画にユーザー・データを入れて使用することはできません。詳細は、*SQL 解説書* を参照してください。

データベース作成操作またはデータベース消去操作が進行中の場合、このコマンドは失敗します。その操作がいったん完了してから、コマンドを再発行できます。

システム一時表スペースにデータベース区画を指定して作成することになっている場合、表スペース定義を検索するために、ADD NODE は MPP システム上の別のノードと通信しなければならないことがあります。他のノードが表スペース定義を応答しなければならない時間を、*start_stop_time* データベース・マネージャー構成パラメーターを使用して分単位で指定します。この時間を超えると、このコマンドは失敗します。その場合、*start_stop_time* の値を増やして、コマンドを再発行してください。

以下も参照

576ページの『START DATABASE MANAGER』

ATTACH

ATTACH

インスタンス・レベルのコマンド (たとえば CREATE DATABASE および FORCE APPLICATION) を実行するインスタンスを指定することを可能にします。このインスタンスは、現在のインスタンス、同じワークステーション上の別のインスタンス、またはリモート・ワークステーションのインスタンスのどれかになります。

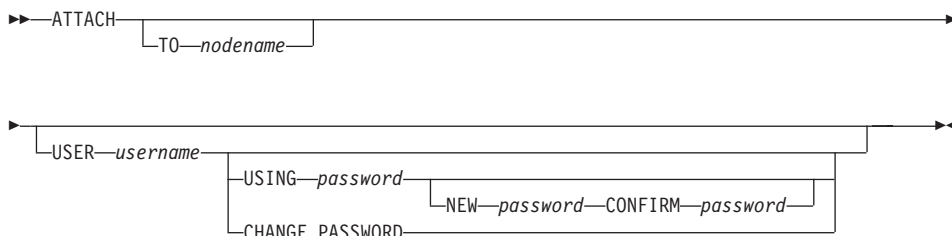
許可

なし

必須接続

なし。このコマンドは、インスタンス接続を確立します。

コマンド構文



コマンド・パラメーター

TO nodename

ユーザーが接続しようとしているインスタンスの別名です。このインスタンスには、ローカル・ノード・ディレクトリーの一致エントリーが必要です。この唯一の例外はローカル・インスタンス (**DB2INSTANCE** 環境変数で指定されている) です。これは、接続のオブジェクトとして指定されることがありますが、ノード・ディレクトリーのノード名として使用することはできません。

USER username

認証識別子を指定します。

USING password

ユーザー名のパスワードを指定します。ユーザー名は指定されているが、パスワードは指定されていない場合、現在のパスワードを要求するプロンプトがユーザーに出されます。入力時にパスワードは表示されません。

NEW password

ユーザー名に割り当てられる新規パスワードを指定します。パスワードの長さは、最大で 18 文字です。パスワードが変更されるシステムは、ユーザー認証がセットアップされた方法によって異なります。

CONFIRM password

新規パスワードと同一のストリング。このパラメーターは、入力エラーを検出するために使用されます。

CHANGE PASSWORD

このオプションが指定されていると、ユーザーにプロンプトが出され、現在のパスワード、新規パスワード、および新規パスワードの確認を要求します。入力時にパスワードは表示されません。

例

2 つのリモート・ノードをカタログにします。

```
db2 catalog tcpip node node1 remote freedom server server1
db2 catalog tcpip node node2 remote flash server server1
```

最初のノードに接続し、すべてのユーザーを強制終了し、その後切り離します。

```
db2 attach to node1
db2 force application all
db2 detach
```

2 番目のノードに接続して、何がオンになっているかを確認します。

```
db2 attach to node2
db2 list applications
```

コマンドがエージェント ID 1、2 および 3 を戻した後で 1 および 3 を強制終了し、その後切り離します。

```
db2 force application (1, 3)
db2 detach
```

現在のインスタンスに接続して (必ずしも必要ではありません。暗黙になります)、すべてのユーザーを強制終了し、その後切り離します (AIX のみ)。

```
db2 attach to $DB2INSTANCE
db2 force application all
db2 detach
```

使用上の注意

コマンドから *nodename* を省略すると、現在の接続状態についての情報が戻されます。

ATTACH

ATTACH が実行されていない場合、インスタンス・レベル・コマンドは、**DB2INSTANCE** 環境変数によって指定した現在のインスタンスで実行されません。

以下も参照

250ページの『DETACH』

BACKUP DATABASE

データベースまたは表スペースのバックアップ・コピーを作成します。

効力範囲

このコマンドは、それが実行されたノードに対してだけ影響を与えます。

許可

以下のどれかが必要です。

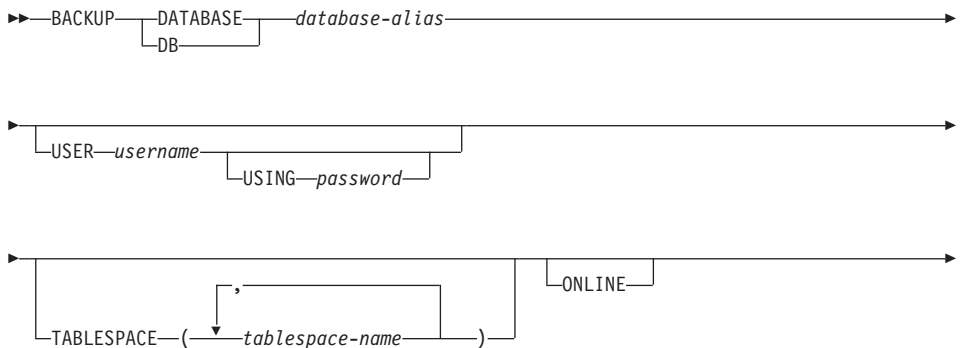
- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

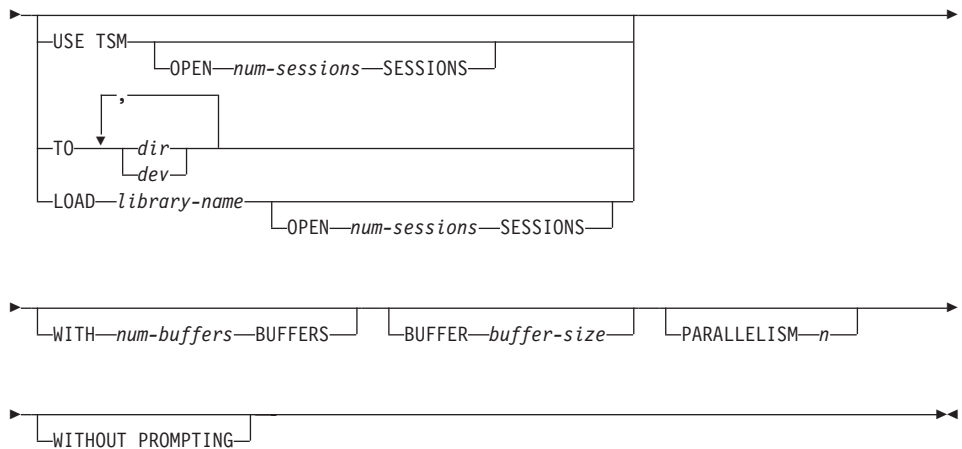
データベース。このコマンドは指定されたデータベースへの接続を自動的に確立します。

注: 指定したデータベースへの接続がすでに存在している場合、それはバックアップ操作に使用されます。接続は、バックアップの完了時に終了します。

コマンド構文



BACKUP DATABASE



コマンド・パラメーター

DATABASE database-alias

バックアップを取るデータベースの別名を指定します。

USER username

データベースのバックアップを取るユーザー名を識別します。

USING password

ユーザー名を承認するために使用するパスワード。パスワードを省略すると、ユーザーに入力を求めるプロンプトが出ます。

TABLESPACE tablespace-name

バックアップを取る表スペースを指定するとき使用する名前のリスト。

ONLINE

オンライン・バックアップを指定します。省略時値はオフライン・バックアップです。オンライン・バックアップは、*logretain* または *userexit* を使用可能にして構成されたデータベースにのみ、使用可能です。

注: オンライン・バックアップ操作は、*sysibm.systables* に IX ロックがある場合、タイムアウトになる可能性があります。それは DB2 バックアップ・ユーティリティーが、LOB を含むオブジェクト上に S ロックを必要とするためです。

USE TSM

バックアップには Tivoli Storage Manager (以前は ADSM と呼ばれる) 出力を使用することを指定します。

OPEN num-sessions SESSIONS

DB2 と TSM または他のバックアップ・ベンダー製品との間で作成される入出力セッションの数。

注: このパラメーターは、テープ、ディスク、または他のローカル装置にバックアップする場合には効果はありません。

TO dir/dev

ディレクトリーまたはテープ装置名のリストです。ディレクトリーが常駐する完全パス名を指定しなければなりません。宛先は、データベース・サーバーに常駐していなければなりません。このパラメーターは、バックアップ・イメージが複数の宛先ディレクトリーや装置にわたる場合に、それらを指定するために繰り返すことができます。宛先が複数指定されている場合 (たとえば、宛先 1、宛先 2、および宛先 3)、宛先 1 が最初にオープンされます。媒体ヘッダーおよび特殊ファイル (構成ファイル、表スペース表、および活動記録ファイルを含む) は、宛先 1 にあります。他の残りの宛先は、オープンされており、これらはバックアップのときに並列で使用されます。

テープ装置やフロッピー・ディスクを使用することにより、メッセージやユーザー入力のプロンプトを生成できます。有効な応答オプションは、次のとおりです。

- c** 続行。警告メッセージを生成した装置の使用を続けます (たとえば、新しいテープをマウントしたときなど)。
- d** 装置の終了。警告メッセージの原因となった装置の使用だけを停止します (たとえば、これ以上テープがない場合など)。
- t** 終了。バックアップまたは復元ユーティリティを打ち切ります。

テープは OS/2 ではサポートされていません。OS/2 の場合、0 または 0: を指定すると、ユーザー出口プログラムが呼び出されて、バックアップ操作が起きます (管理の手引きを参照)。このオプションは、他のプラットフォームでは無効です。

LOAD library-name

使用するバックアップおよび復元 I/O 関数を含む共用ライブラリー (OS/2 または Windows オペレーティング・システムでは DLL) の名前。フルパスで指定することができます。フルパスを指定していない場合、省略時値はユーザー出口プログラムが常駐しているパスになります。

BACKUP DATABASE

WITH num-buffers BUFFERS

使用するバッファの数です。省略時値は 2 です。ただし、バックアップを複数の場所に作成する場合は、パフォーマンスを向上させるために多数のバッファを使用することができます。

BUFFER buffer-size

バックアップ・イメージを構築するときに使用する、数ページに渡るバッファのサイズです。このパラメーターの最小値は 8 ページです。省略時値は 1024 ページです。バッファ・サイズがゼロに指定されている場合、データベース・マネージャー構成パラメーターの値 *backbufsz* が、バッファ割り振りサイズとして使用されます。

テープ装置を使用するには、SCO UnixWare 7 上の DB2 ユーザーは、バッファ・サイズを 16 に指定する必要があります。

PARALLELISM n

バックアップ・ユーティリティーによって同時に読み取り可能な表スペースの数を決定します。省略時値は 1 です。

WITHOUT PROMPTING

バックアップは、管理されることなく実行されるため、通常はユーザーの介入を必要とするアクションでエラー・メッセージが戻されるように指定されます。

例

```
db2 backup database sample use tsm open 2 sessions with 4 buffers
```

```
db2 backup database payroll tablespace syscatspace, userspace1 to /dev/rmt0, /dev/rmt1 with 8 buffers without prompting
```

使用上の注意

データベースおよび表スペースの両方のバックアップで、バックアップ・ユーティリティーは、複数の表スペースから同時に読み取り、複数の場所にある装置に同時に書き込むことができます。

データベース・レベル・バックアップ

注: それぞれのデータベースのバックアップは、通常の基準で行われます。データベースが損傷を受けたり、破壊されたりした場合、バックアップ済み複写状態に戻すことができます (539ページの『RESTORE DATABASE』を参照)。

バックアップ操作のときに、正常に格納されたデータベースでロールフォワード回復を行うことが可能である場合、障害が起こる前の状態にまでデータベースを戻すことができます (550ページの『ROLLFORWARD DATABASE』を参照)。

このバックアップは、ハード・ディスク、ディスクレット、テープ、TSM ユーティリティー、または他の、DB2 で使用可能なベンダー製品に直接接続することができます。

UNIX ベースのシステムでは、バックアップ・ファイル名がピリオドにより分離されているいくつかの情報単位が連結してできています。

dbname.type.instance.nodexxxx.catnxxxx.yyyymmddhhmmss.seq

dbname	1~8 文字のデータベースの別名です。
type	実行するバックアップのタイプです (完全データベースには 0、または表スペース・レベル・バックアップには 3 です)。
instance	1~8 文字のデータベース・インスタンス名です。
nodexxxx	ノードの番号。非区分データベース・システムでは、この値は常にゼロです (NODE0000)。区分データベース・システムでは、この値は NODExxxx です。xxxx は、db2nodes.cfg ファイル中でノードに割り当てられている数です。
catnxxxx	データベース用のカタログ・ノードの数です。非区分データベース・システムでは、この値は常にゼロです (CATN0000)。区分データベース・システムでは、この値は CATNxxxx です。xxxx は、db2nodes.cfg ファイル中でノードに割り当てられている数です。
yyymmdd	日付 (年、月、日) です。
hhmmss	時刻 (時間、分、秒) です。
seq	3 桁の順序番号です。

ハード・ディスク、テープ、TSM および他のベンダーに加えて、バックアップを直接ディスクレットに取ることができます。OS/2 または Windows オペレーティング・システムの場合、汎用テープ装置はサポートされていないので、テープ装置のタイプごとに固有のデバイス・ドライバが必要です。

BACKUP DATABASE

OS/2 および Windows オペレーティング・システムの FAT ファイル・システムにバックアップを取るには、ユーザーは 8.3 命名規則に適合するようにしなければなりません。ファイルは、5 レベル・サブディレクトリー・ツリーにあります。このことを以下に示します。

dbname.type#db2instance.nodexxx#catnxxx#yyyymmdd#hhmmss.seq

dbname	1~8 文字のデータベースの別名です。
type	実行するバックアップのタイプです。完全データベースには 0、表スペース・レベル・バックアップには 3、表ロードからのコピーには 4 です。
db2instance	1~8 文字のデータベース・インスタンス名です。
nodexxx	ノードの番号。非区分データベース・システムでは、この値は常にゼロです (NODE000)。区分データベース・システムでは、この値は NODExxx です。xxx は、db2nodes.cfg ファイル中でノードに割り当てられている数です。
catnxxx	データベース用のカタログ・ノードの数です。非区分データベース・システムでは、この値は常にゼロです (CATN000)。区分データベース・システムでは、この値は CATNxxx です。xxx は、db2nodes.cfg ファイル中でノードに割り当てられている数です。
yyyymmdd	日付 (年、月、日) です。
hhmmss	時刻 (時間、分、秒) です。
seq	3 桁の順序番号です。

ロールフォワード回復が可能である場合は、オンライン・バックアップは許可されます。ユーティリティーがアクセスされており、他のアプリケーションにより修正されている場合は、このユーティリティーはオンライン・バックアップを実行することが可能です。

オフライン・バックアップを実行するには、ユーティリティーは排他モードでデータベースを接続することができなければなりません。呼び出しアプリケーションを含む、アプリケーションのどれかがすでにデータベースに接続されている場合は、BACKUP は失敗します。接続に成功した場合、このバックアップが完了するまでは BACKUP により他のアプリケーションは締め出された状態になります。

現在データベースを必要としない場合は、BACKUP DATABASE をオフラインで実行します。

データベースの状態が一貫していない場合、オフライン・バックアップ操作は失敗します。データベースが矛盾している場合、破損回復によって一貫した状態に戻すためにデータベースを再始動してから、`BACKUP` を実行する必要があります (537ページの『`RESTART DATABASE`』、および管理の手引きの `autorestart` 構成パラメーターの説明を参照)。回復時にシステムが失敗した後で、データベースが部分的に復元された状態にある場合、バックアップを実行する前に復元操作を再び実行して、完了させなければなりません。データベースが復元され、ロールフォワード操作が必要である場合、バックアップを取る前にデータベースを下方にロールして一貫した状態にしなければなりません。

データベースがロールフォワード使用不能状態からロールフォワード使用可能状態に変わった場合、`logretain` または `userexit` データベース構成オプションを使用可能にしてからデータベースのバックアップを取る必要があります (279ページの『`GET DATABASE CONFIGURATION`』を参照)。

表スペース・レベル・バックアップ

表スペース・レベル・バックアップには、データベースの表スペースが 1 つまたは複数含まれています。これはコマンドを実行するときに指定することができます。表スペース・レベル・バックアップは、オンラインでもオフラインでも行えますが、データベースはロールフォワードする回復を使用可能にして構成する必要があります。

表スペース・レベル・バックアップは、特定の表スペースのみに影響する問題を回復するときに使用します。この回復を実行するときは、他のすべての表スペースで処理をすることが可能です。

復元した表スペースが、他のデータベースと同期化されていることを確認するには、ログの最後まで (または表スペースを最後に使用したところまで) 表スペースの下方ロールを行わなければなりません。このため、表スペース・レベルのバックアップおよび復元が行えるのは、ロールフォワード回復が可能な場合のみです。表スペース・レベル・バックアップを実行した後に、ロールフォワード回復が行えない場合、バックアップから復元することができないため、早めに現時点まで表スペースの下方ロールを行ってください。この場合、その時点より前に取った表スペース・レベルのバックアップはすべて復元できなくなります。そのようなバックアップを復元しようとする、復元操作は失敗します。バックアップが無効である (たとえば、新しいログ順序列を作成して、データベースが復元およびロールフォワードされている場合) を判別することができない場合には、その復元は、正常に行われますが、ロールフォワード回復処理中に、壊れた復元セットが検出されることとなります。

BACKUP DATABASE

ユーザーは、データ、索引、ロング・フィールド (LONG)、およびラージ・オブジェクト (LOB) を別々の表スペースに入れることも可能です。同じ表のロング・フィールドおよび LOB データは、同じ表スペースに存在していなければなりません。

表のそれぞれの構成要素は、同じ表の他の構成要素からは独立させて、常駐している表スペースでバックアップを取り、復元させることが可能です。

一時表のために、システムまたはユーザーの表スペースのバックアップを取る必要はありません。バックアップを取る表スペースのリストに、このような表スペースがある場合、BACKUP は失敗します。

表スペース・レベルのバックアップと復元は、同時に実行することはできません。

以下も参照

455ページの『MIGRATE DATABASE』

539ページの『RESTORE DATABASE』

550ページの『ROLLFORWARD DATABASE』

BIND

バインド・ユーティリティーを呼び出すことにより、事前コンパイラーに生成されるバインド・ファイルに SQL ステートメントを保管し、データベースに保管されるパッケージを作成します。

効力範囲

このコマンドは、`db2nodes.cfg` 中のどのノードからでも発行できます。実行すると、カタログ・ノードのデータベース・カタログが更新されます。その影響はすべてのノードから見えます。

許可

以下のどれかが必要です。

- *sysadm* または *dbadm* の権限
- パッケージが存在しない場合は、BINDADD 特権および以下のどちらかが必要です。
 - パッケージのスキーマ名が存在しない場合は、データベースに対する IMPLICIT_SCHEMA 権限
 - パッケージのスキーマ名が存在する場合は、スキーマに対する CREATEIN 特権
- パッケージが存在する場合は、スキーマに対する ALTERIN 特権
- パッケージに対する BIND 特権 (パッケージが存在する場合)

ユーザーには、アプリケーション内の静的 SQL ステートメントをコンパイルするのに必要な特権もすべて必要になります。グループに付与された特権は、静的ステートメントの許可検査では使用されません。ユーザーに *sysadm* 権限があつてバインドを完成させる明示特権がない場合、データベース・マネージャーは、明示的な *dbadm* 権限を自動的に付与します。

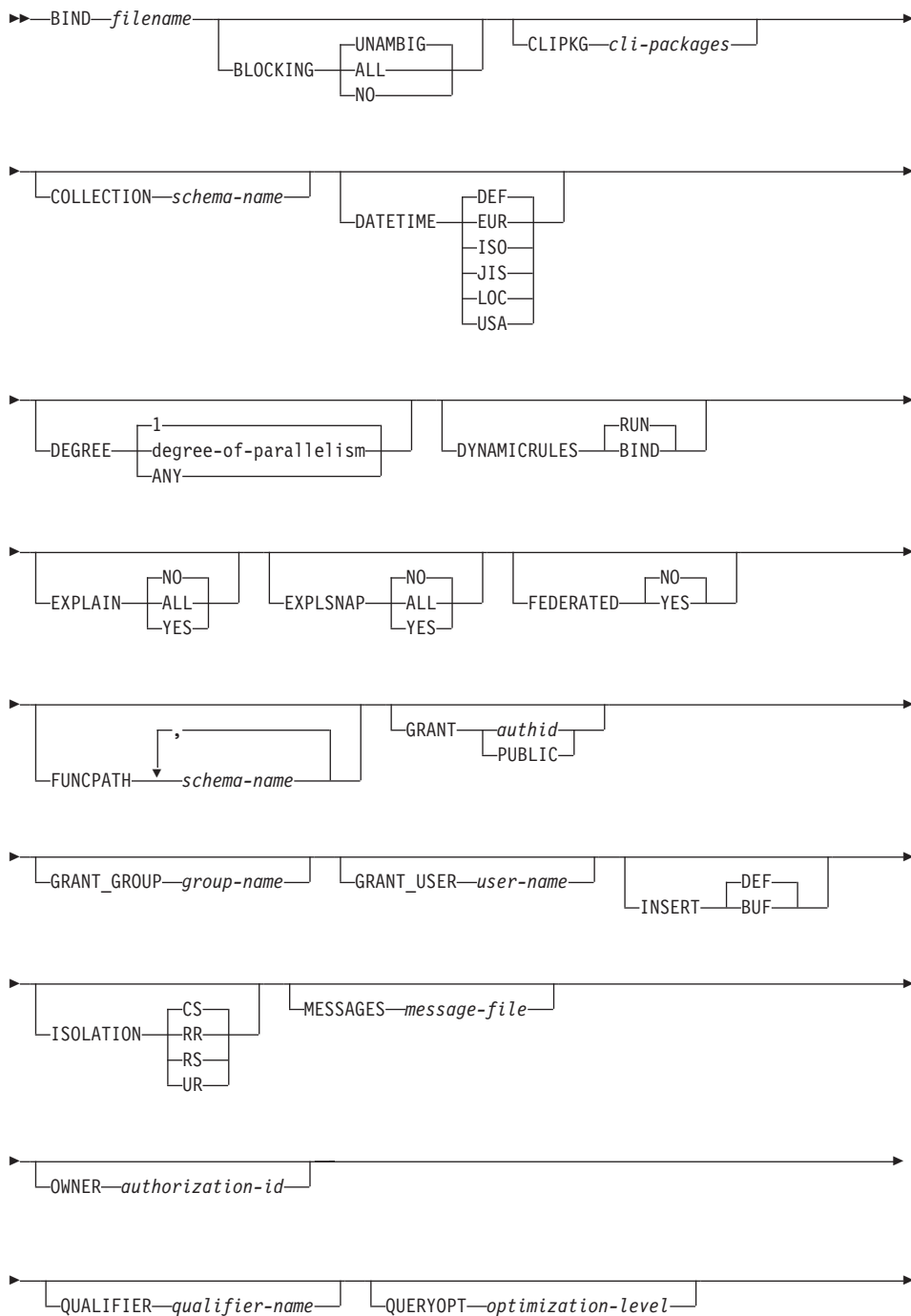
必須接続

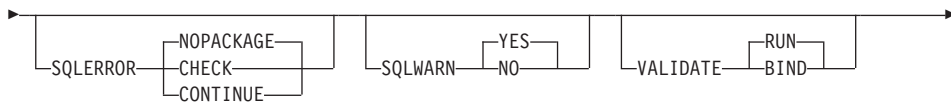
データベース。暗黙接続が可能な場合には、省略時データベースへの接続が確立されます。

コマンド構文

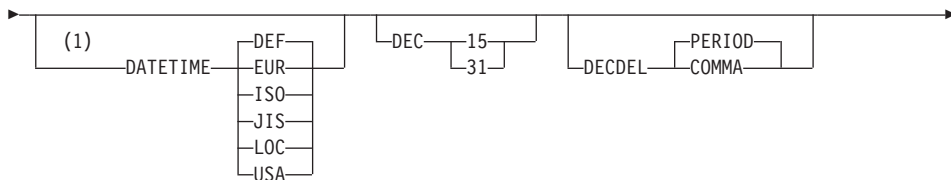
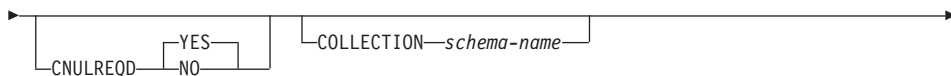
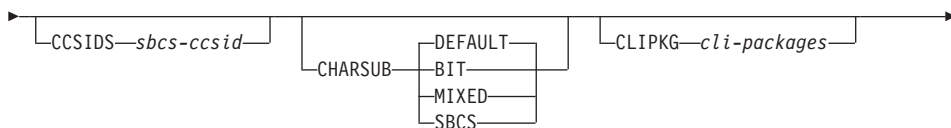
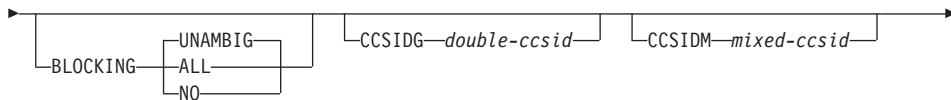
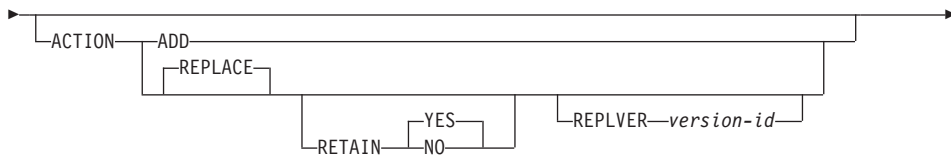
DB2 の場合

BIND

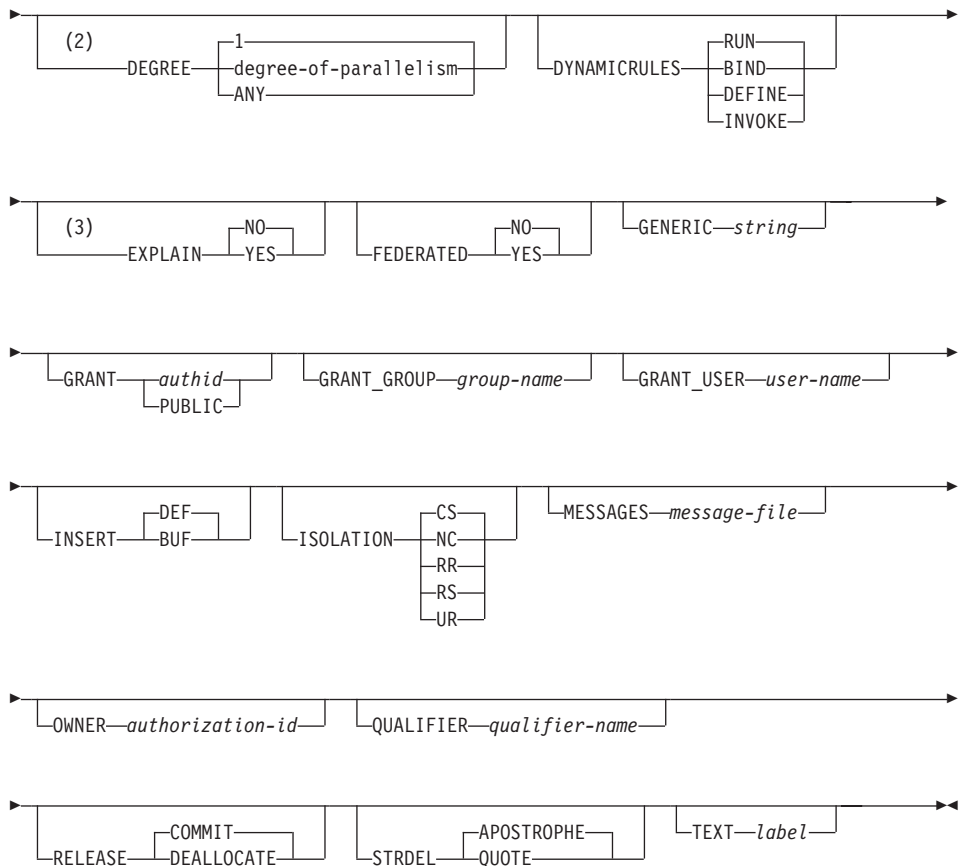




DRDA の場合



BIND



注:

- 1 DATETIME DEF オプションは DRDA ではサポートされておらず、DB2 コネクトを経て ISO にマップされます。
- 2 DEGREE オプションは DRDA レベル 2 のアプリケーション・サーバーでしかサポートされていません。
- 3 DRDA は、EXPLAIN オプションが値 YES または NO を持つように定義します。EXPLAIN オプションを指定しないと、これは DRDA "EXPLAIN NO" にマップされます。

コマンド・パラメーター

filename

アプリケーション・プログラムをプリコンパイルしたときに生成されたバインド・ファイル、または複数のバインド・ファイル名を含むリス

ト・ファイルの名前を指定します。バインド・ファイルの拡張子は .bnd です。また、全パス名も指定できます。

リスト・ファイルを指定した場合、その名前の先頭文字は @ 文字でなければなりません。リスト・ファイルには、数行のバインド・ファイル名を含めることができます。同一行にリストするバインド・ファイルはプラス (+) 文字で区切る必要がありますが、各行の先頭ファイルの前や最後のファイルの後に + を挿入することはできません。たとえば、次のように指定します。

```
/u/smith/sqllib/bnd/@all.lst
```

これは、次のバインド・ファイルを含むリスト・ファイルです。

```
mybind1.bnd+mybind2.bnd+mybind3.bnd+
mybind4.bnd+mybind5.bnd+
mybind6.bnd+
mybind7.bnd
```

ACTION

パッケージを追加または置換できるかどうかを示します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。

ADD 名前付きパッケージが存在せず、新規パッケージを作成するということを指示します。すでにパッケージがある場合は、実行停止状態となり、診断エラー・メッセージが戻されます。

REPLACE

既存のパッケージを、ロケーション、コレクション、およびパッケージ名が同じ新規パッケージと置き換えることを指示します。

RETAIN

パッケージを置き換えたときに EXECUTE 権限が保持されるかどうかを指示します。パッケージの所有権を変更した場合、新規所有者は前のパッケージ所有者に BIND 権限と EXECUTE 権限を付与します。

NO パッケージを置き換えたとき、EXECUTE 権限を保持しません。

YES パッケージを置き換えたとき、EXECUTE 権限を保持します。

REPLVER version-id

特定のバージョンのパッケージを置き換えます。バー

ジョン識別子は、どのバージョンのパッケージを置き換えるのかを指定するものです。最大長は 254 文字です。

BLOCKING

行のブロック化については、*管理の手引き* または *アプリケーション開発の手引き* を参照してください。

ALL 次のカーソルをブロック化することを指定します。

- 読み取り専用カーソル。
- FOR UPDATE OF と指定されていないカーソル。

未確定のカーソルは、読み取り専用として扱われます。

NO どのカーソルもブロック化しないことを指定します。未確定のカーソルは、更新可能として扱われます。

UNAMBIG

次のカーソルをブロック化することを指定します。

- 読み取り専用カーソル。
- FOR UPDATE OF と指定されていないカーソル。

未確定のカーソルは、更新可能として扱われます。

CCSIDG double-ccsid

CREATE および ALTER TABLE SQL ステートメントの文字列定義で、2 バイト文字用のコード化文字セット識別子 (CCSID) (特定の CCSID 文節は使用しない) を指定する整数。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

CCSIDM mixed-ccsid

CREATE および ALTER TABLE SQL ステートメントの文字列定義で、混合バイト文字用のコード化文字セット識別子 (CCSID) (特定の CCSID 文節は使用しない) を指定する整数。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

CCSIDS sbcs-ccsid

CREATE および ALTER TABLE SQL ステートメントの文字列定義で、1 バイト文字用のコード化文字セット識別子 (CCSID) (特定の CCSID 文節は使用しない) を指定する整数。なお、DB2 はこの DRDA

プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

CHARSUB

CREATE および **ALTER TABLE SQL** ステートメントの列定義に使用する、省略時の文字サブタイプを指定します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。

BIT 明示的にサブタイプを指定しなかった場合、すべての新規文字列に **FOR BIT DATA SQL** 文字サブタイプが使用されます。

DEFAULT

明示的にサブタイプを指定しなかった場合、すべての新規文字列にターゲット・システムが定義した省略時サブタイプが使用されます。

MIXED

明示的にサブタイプを指定しなかった場合、すべての新規文字列に **FOR MIXED DATA SQL** 文字サブタイプが使用されます。

SBCS 明示的にサブタイプを指定しなかった場合、すべての新規文字列に **FOR SBCS DATA SQL** 文字サブタイプが使用されます。

CLIPKG cli-packages

3 ~ 30 の整数で、CLI バインド・ファイルをデータベースに対してバインドするとき作成される、CLI ラージ・パッケージの数を指定します。CLI バインド・ファイルとパッケージの詳細については、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。

CNULREQD

このオプションは DRDA でサポートされていない **langlevel** プリコンパイル・オプションと関連します。これは、C または C++ アプリケーションで作成されたバインド・ファイルの場合のみ有効です。この DRDA バインド・オプションは、DB2 ではサポートされていません。

NO C ストリング・ホスト変数中の NULL 終止符に関して、**langlevel SAA1** プリコンパイル・オプションに基づいてアプリケーションがコード化された場合です。

YES C ストリング・ホスト変数中の NULL 終止符に関して、

langlevel MIA プリコンパイル・オプションに基づいてアプリケーションがコード化された場合です。

COLLECTION schema-name

パッケージ用の 30 文字の収集識別子を指定します。これを指定しなかった場合、パッケージを処理する際には、ユーザーの許可識別子が使用されます。

DATETIME

使用する日時形式を指定します。日時形式の詳細については、*SQL 解説書* を参照してください。

DEF データベースの国別コードと対応する日時形式を使用します。

EUR IBM 欧州規格の日時形式を使用します。

ISO 国際標準化機構規格の日時形式を使用します。

JIS 日本工業規格の日時形式を使用します。

LOC データベースの国別コードと対応する地域別日時形式を使用します。

USA IBM 米国規格の日時形式を使用します。

DEC 10 進算術演算に使用する最大精度を指定します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

15 10 進算術演算に 15 桁精度が使用されます。

31 10 進算術演算に 31 桁精度が使用されます。

DECDEL

10 進数および浮動小数点リテラル中で 10 進小数点標識としてピリオド (.) またはコンマ (,) のどちらかを指定します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

COMMA

10 進小数点標識としてコンマ (,) を使用します。

PERIOD

10 進小数点標識としてピリオド (.) を使用します。

DEGREE

SMP システムで静的 SQL ステートメントを実行するための並列処理の程度を指定します。このオプションは、CREATE INDEX 並列処理には影響を与えません。

1 ステートメントの実行に並列処理を使用しません。

degree-of-parallelism

ステートメントを実行する際の並列処理の程度を指定します。値の範囲は 2 ~ 32 767 です。

ANY ステートメントの実行時にデータベース・マネージャーで判別した程度で並列処理を行うよう指定します。

DYNAMICRULES

許可 ID に使用される値の初期設定、および未修飾オブジェクト参照の暗黙的な修飾の、実行時の動的 SQL に適用される規則を定義します。

RUN パッケージを実行するユーザーの許可 ID が使用されるように指定します。これが省略時値です。

BIND 許可および修飾の静的 SQL に適用されるすべての規則が、実行時に使用されるように指定します。つまり、パッケージ所有者の許可 ID が動的 SQL の権限検査に使用され、省略時のパッケージ修飾子が動的 SQL ステートメント内の未修飾オブジェクト参照の暗黙的な修飾に使用されます。

このオプションを指定してパッケージをバインドする場合、パッケージのバインダーはパッケージのユーザーが受け取るべきでない許可を持つことはできません。動的 SQL ステートメントがパッケージ所有者の許可 ID を使用するためです。次の動的な準備済み SQL ステートメントは、このオプションを指定してバインドされているパッケージ内では使用できません。

GRANT、REVOKE、ALTER、CREATE、DROP、COMMENT ON、RENAME、SET CONSTRAINTS、および SET EVENT MONITOR STATE です。

DEFINE

UDF またはストアド・プロシージャ内にある動的 SQL ステートメントの実行に使用する許可識別子が、UDF またはストアド・プロシージャの定義者であることを指示します。このオプションは、DB2 ではサポートされません。

INVOKE

UDF またはストアド・プロシージャ内にある動的 SQL ステートメントの実行に使用する許可識別子が、UDF またはストアド・プロシージャの呼び出し元であるということを指示します。このオプションは、DB2 ではサポートされません。

EXPLAIN

各 SQL ステートメント用に選択したアクセス・プランに関する Explain 表の情報を、パッケージに保管します。DRDA では、このオプションの ALL 値がサポートされていません。

NO Explain 情報はキャプチャーされません。

YES Explain 表には、静的ステートメントの場合は prep/bind 時間で、増分バインド・ステートメントの場合はランタイムで、選択されたアクセス・プランについての情報が取り込まれます。

ALL 適格な静的 SQL ステートメントの Explain 情報が、prep/bind 時間で各 Explain 表に入れます。適格な増分バインド SQL ステートメントの Explain 情報が、ランタイムで各 Explain 表に入れます。さらに、CURRENT EXPLAIN SNAPSHOT レジスターが NO に設定されていても、Explain 情報は実行時に適格な動的 SQL ステートメント用に集められます。特殊レジスターの詳細については、SQL 解説書を参照してください。

注: DRDA では、EXPLAIN のこの値 (ALL) はサポートされていません。

EXPLSNAP

Explain 表に Explain スナップショットを保管します。この DB2 プリコンパイル / バインド・オプションは、DRDA ではサポートされていません。

NO Explain スナップショットはキャプチャーされません。

YES Explain 表には、静的ステートメントの場合は prep/bind 時間で、増分バインド・ステートメントの場合はランタイムで、適格な各静的 SQL ステートメントの Explain スナップショットが、Explain 表内に入れます。

ALL 適格な各静的 SQL ステートメントの Explain スナップショットが、prep/bind 時間で Explain 表内に入れます。適格な増分バインド SQL ステートメントの Explain スナップショット情報が、ランタイムで各 Explain 表に入れます。さら

に、CURRENT EXPLAIN SNAPSHOT レジスターが NO に設定されていても、 Explain スナップショット情報は実行時に適格な動的 SQL ステートメント用に集められます。特殊レジスターの詳細については、SQL 解説書 を参照してください。

FEDERATED

パッケージ内の静的 SQL ステートメントが、通称または統合視点を参照するかどうかを指定します。このオプションを指定しており、通称または統合視点が参照されない場合、エラーが戻され、パッケージは作成されません。このオプションが指定されず、通称または統合視点がパッケージ内の静的ステートメントによって参照される場合、エラーが戻され、パッケージは作成されません。

NO ユーティリティは DB2 ユニバーサル・データベースに接続します。これが省略時値です。

YES ユーティリティは DB2 統合システムにアクセスします。

FUNCPATH

静的 SQL で、ユーザー定義の個別タイプおよび機能を解析する際に使用する機能パスを指定します。このオプションを指定しなかった場合、省略時の機能パスは "SYSIBM"、"SYSFUN"、または USER になります。この DB2 プリコンパイル / バインド・オプションは、DRDA ではサポートされていません。

schema-name

SQL 識別子 (通常または区切り)。これは、アプリケーション・サーバーに存在するスキーマを識別します。スキーマが存在する場合、プリコンパイル時やバインド時に妥当性検査は行われません。同一スキーマは、機能パス内に一度しか存在できません。指定できるスキーマ数は、処理結果の機能パスの長さによって限定され、254 バイトを超えることはできません。スキーマ SYSIBM は、明示的に指定する必要がありません。機能パス内に含まれていなければ、最初のスキーマに暗黙的に想定されます。詳細については、SQL 解説書 を参照してください。

GENERIC string

宛先 DRDA データベースに新規バインド・オプションを渡す手段を提供します。宛先データベースで定義されていても、ローカル・コマンドで認識されない新規バインド・オプションをサポートします。165ページの『BIND』または 457ページの『PRECOMPILE PROGRAM』で定義されているバインド・オプションを渡すようにするには、このオプ

BIND

ションを使用しないでください。このオプションは、動的 SQL のパフォーマンスをかなり向上させることができます。構文は次のとおりです。

```
generic "option1 value1 option2 value2 ..."
```

各オプションと値は、1 つまたは複数のブランク・スペースで区切らなければなりません。たとえば、宛先 DRDA データベースが DB2 MVS バージョン 5 の場合、次のようにします。

```
generic "keepdynamic yes"
```

こうして、新規 **keepdynamic YES** オプションは PRECOMPILE PROGRAM または BIND コマンドでローカルに定義されてはいますが、バインドできます。

ストリングの最大長は 1023 バイトです。この DRDA バインド・オプションは、現在 DB2 MVS バージョン 5 のみでサポートされています。DB2 ではサポートされていません。

GRANT

authid 指定したユーザー名またはグループ ID に EXECUTE 特権と BIND 特権を付与します。

PUBLIC

PUBLIC に EXECUTE 特権と BIND 特権を付与します。

GRANT_GROUP group-name

指定したグループ ID に EXECUTE 特権と BIND 特権を付与します。

GRANT_USER user-name

指定したユーザー名に EXECUTE 特権と BIND 特権を付与します。

INSERT

DB2 エンタープライズ拡張エディション・サーバーヘブリコンパイルまたはバインドされているプログラムが、パフォーマンス向上のために挿入データをバッファリングすることを要求できるようにします。

BUF アプリケーションからの挿入データをバッファリングすることを指定します。

DEF アプリケーションからの挿入データをバッファリングしないことを指定します。

ISOLATION

このパッケージにバインドされるプログラムを、他の実行プログラムの

影響からどの程度分離できるかを指定します。分離レベルの詳細については、*SQL 解説書* を参照してください。

- CS** 分離レベルとしてカーソル固定を指定します。
- NC** コミットしません。コミット制御が使用されないということを指定します。なお、DB2 はこの分離レベルをサポートしません。
- RR** 分離レベルとして反復可能読み取りを指定します。
- RS** 分離レベルとして読み取り固定を指定します。読み取り固定は、パッケージ内での SQL ステートメントの実行を、他のアプリケーションが読み取りおよび変更を行った行に対する処理から分離させます。
- UR** 分離レベルとして非コミット読み取りを指定します。

MESSAGES message-file

警告メッセージ、エラー・メッセージ、および完了状況メッセージの宛先を指定します。メッセージ・ファイルは、バインドが正常であるかどうかによって作成されます。メッセージ・ファイル名を指定しなかった場合、メッセージは標準出力に書き込まれます。ファイルへの完全パスを指定しなかった場合は、現行ディレクトリーが使用されます。なお、既存ファイルの名前を指定すると、そのファイルの内容は上書きされます。

OWNER authorization-id

パッケージ所有者の 30 文字の許可識別子を指定します。その所有者には、パッケージに含まれる SQL ステートメントを実行するための特権が必要です。SYSADM または DBADM 許可を持つユーザーのみが、ユーザー ID 以外の許可識別子を指定できます。省略時値は、プリコンパイル / バインド処理の 1 次許可 ID です。SYSIBM、SYSCAT、および SYSSTAT はこのオプションには無効な値です。

QUALIFIER qualifier-name

パッケージに含まれる未修飾オブジェクトの 30 文字の暗黙修飾子を指定します。owner が明示的に指定されていれば、その所有者の許可 ID が省略時 ID になります。

QUERYOPT optimization-level

パッケージに含まれるすべての静的 SQL ステートメントに必要な最適化レベルを指示します。省略時値は 5 です。使用可能な最適化レベルの範囲の詳細については、*SQL 解説書* の SET CURRENT QUERY

OPTIMIZATION ステートメントを参照してください。この DB2 プリコンパイル / バインド・オプションは、DRDA ではサポートされていません。

RELEASE

リソースを、各 COMMIT ポイントで解放するか、アプリケーションの終了時に解放するかどうかを指示します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。

COMMIT

各コミット点でリソースを解放します。これは、動的 SQL ステートメントに使用されます。

DEALLOCATE

アプリケーションの終了時にだけリソースを解放します。

SQLERROR

エラーを検出した場合に、パッケージまたはバインド・ファイルを作成するかどうかを指示します。

CHECK

ターゲット・システムがすべての構文を実行し、バインドしている SQL ステートメントの意味を調べるということを指定します。この処理の一部としてパッケージが作成されることはありません。バインド中に、名前とバージョンが同じ既存パッケージを検出した場合、その既存パッケージはドロップも置換 (**ACTION REPLACE** を指定した場合) もされません。

CONTINUE

SQL ステートメントのバインド時にエラーが発生しても、パッケージを作成します。許可または存在などの理由でバインドに失敗したこれらのステートメントは、VALIDATE RUN も指定されている場合は、実行時に増分でバインドすることができます。ランタイムでこれらのステートメントを実行しようとすると、エラー (SQLCODE -525, SQLSTATE 51015) が生成されます。

NOPACKAGE

エラーを検出した場合、パッケージもバインド・ファイルも作成しません。

SQLWARN

動的 SQL ステートメントの完了時 (PREPARE または EXECUTE IMMEDIATE を通して)、または記述処理時 (PREPARE...INTO または

DESCRIBE を通して) に警告を戻すかどうかを指示します。この DB2 プリコンパイル / バインド・オプションは、DRDA ではサポートされていません。

NO SQL コンパイラから警告を戻しません。

YES SQL コンパイラから警告を戻します。

注: SQLCODE +238 は例外です。これは、**SQLWARN** オプションの値が何であろうと戻されます。

STRDEL

SQL ステートメントで使用するストリング区切り文字として、アポストロフィ (') または二重引用符 (") のどちらを使用するか指定します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

APOSTROPHE

ストリング区切り文字として、アポストロフィ (') を使用します。

QUOTE

ストリング区切り文字として、二重引用符 (") を使用します。

TEXT label

パッケージの記述。最大長は 255 文字です。また、省略時値はブランクです。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。

TRANSFORM GROUP

静的 SQL ステートメントが、ユーザー定義の構造型の値をホスト・プログラムと交換するために使用する、変換グループ名を指定します。この変換グループは、動的 SQL ステートメントには使用されません。また、パラメーターの交換や外部関数またはメソッドの結果にも使用されません。

groupname

SQL 識別子。長さは最大で 18 文字です。グループ名には、修飾子接頭部を含めることはできません。また、接頭部 SYS はデータベースで使用するために予約されているので、その接頭部は使用できません。ホスト変数とやりとりする静的 SQL ステートメントでは、構造型の値の交換に使用する変換グループの名前は以下ようになります。

- TRANSFORM GROUP バインド・オプション内のグループ名 (もしあれば)
- TRANSFORM GROUP 準備オプション内のグループ名。最初の事前コンパイル時に指定したとおりのもの (もしあれば)
- DB2_PROGRAM グループ。グループ名が DB2_PROGRAM の、特定のタイプに対する変換がある場合。
- 上記のいずれの条件もない場合には、変換グループは使用されません。

静的 SQL ステートメントのバインド時には、以下のエラーが発生する可能性があります。

- SQLCODE yyy, SQLSTATE xxxxx: 変換が必要ですが、静的変換グループが選択されていません。
- SQLCODE yyy, SQLSTATE xxxxx: 選択された変換グループには、交換するデータ・タイプに必要な変換が含まれていません (入力変数用の TO SQL、出力変数用の FROM SQL)。
- SQLCODE yyy, SQLSTATE xxxxx: FROM SQL 変換の結果タイプは、出力変数のタイプと互換性がありません。または、TO SQL 変換のパラメーター・タイプは、入力変数のタイプと互換性がありません。

VALIDATE

データベース・マネージャーが、許可エラーとエラー未検出のオブジェクトをいつ検査するかを判別します。この妥当性検査には、パッケージ所有者の許可 ID が使用されます。

BIND プリコンパイル / バインド時に妥当性検査が実行されます。オブジェクトが 1 つもない場合、または権限がまったく保持されていない場合、エラー・メッセージが作成されます。 **SQLERROR CONTINUE** を指定した場合、エラー・メッセージにかかわらずパッケージ / バインド・ファイルは作成されますが、エラーのあるステートメントは実行できません。

RUN バインド時に妥当性検査が行われます。すべてのオブジェクトが存在し、全権限が保持されていれば、それ以上実行しても検査は行われません。

プリコンパイル / バインド時にオブジェクトが 1 つもない場合、または権限がまったく保持されていない場合、 **SQLERROR CONTINUE** オプションの設定とは

無関係に警告メッセージが作成されて、パッケージは正常にバインドされます。ただし、プリコンパイル / バインド処理時に SQL ステートメントの権限検査と存在検査に障害が生じた場合、実行時に再実行される可能性があります。

例

以下は、myapp.bnd (myapp.sqc プログラムのプリコンパイル時に生成されるバインド・ファイル) と接続が確立しているデータベースをバインドする例です。

```
db2 bind myapp.bnd
```

バインド処理で生じたすべてのメッセージは、標準出力に送信されます。

使用上の注意

バインドは、アプリケーション・プログラム・ソース・ファイルのプリコンパイル処理の一部として、または後の分離ステップとして実行することができます。分離処理としてバインドを実行するときは、BIND を使用してください。

パッケージの作成に使用する名前は、それを生成したソース・ファイル名を基にして指定され (既存パスや拡張子は廃棄されます)、バインド・ファイルに保管されます。たとえば、myapp.sql というプリコンパイル・ソース・ファイルは、myapp.bnd という省略時バインド・ファイルと省略時パッケージ名の MYAPP を生成します。ただし、**BINDFILE** および **PACKAGE** オプションを使用すれば、プリコンパイル時にバインド・ファイル名とパッケージ名を変更することができます。

すでに存在しないスキーマ名にパッケージをバインドすると、そのスキーマを暗黙に作成することになります。スキーマの所有者は SYSIBM になります。スキーマに対する CREATEIN 特権が PUBLIC に付与されます。

BIND は、開始されたトランザクションのもとで実行されます。バインドの実行後、BIND は COMMIT か ROLLBACK を出して、現行トランザクションを終了させ、別のトランザクションを開始します。

致命的エラーまたは 100 を超えるエラーが生じた場合、バインドは停止します。致命的エラーが生じた場合、ユーティリティーは処理を停止させ、全ファイルのクローズを試み、そのパッケージを廃棄します。

パッケージが **dynamicrules** バインドでバインドされる場合、バインド・オプション **owner** の暗黙または明示的な値は、動的 SQL ステートメントの許可

検査に使用され、バインド・オプション **qualifier** の暗黙または明示的な値は、動的 SQL ステートメント内の未修飾オブジェクトの暗黙修飾子として使用されます。複数のパッケージが単一の接続中に参照される場合、特定のパッケージにより準備された動的 SQL は、そのパッケージへのバインド・オプションにしたがって動作します。特殊レジスター **CURRENT SCHEMA** の値は、**dynamicrules** バインドを持つパッケージ結合にある修飾子には影響しません。

クライアント / サーバー環境で、**BIND** コマンドがバージョン 6 のサーバーに接続されたバージョン 6 より前のクライアントに発行される場合、エラーを示す **SQL0020W** がクライアントに戻される場合があります。この状態は、コマンドが **DYNAMICRULES** または **OWNER** パラメーターの一方または両方を使用する場合に起きます。エラー・メッセージが表示されても、それらのパラメーターは引き続き有効であり、使用されます。**SQL0020W** メッセージに表示される他のパラメーターは正しくエラーと示され、メッセージが示すとおりは無視されます。これらの他のパラメーターには、以下のものが含まれません。**ACTION**、**CCSIDG**、**CCSIDM**、**CCSIDS**、**CHARSUB**、**CNULREQD**、**DEC**、**DECDEL**、**GENERIC**、**RELEASE**、**REPLVER**、**RETAIN**、**STRDEL**、**TEXT**、**VALIDATE**、**VERSION**、および値 **CONTINUE** が指定された **SQLERROR**。

同じように、バージョン 6 以前のクライアントがバージョン 7 のサーバーに接続されると、**DYNAMICRULES**、**OWNER**、または **VALIDATE** パラメーターのいずれかを組み合わせて使用すると、**SQL0020W** エラーが示されます。エラー・メッセージが表示されても、それらのパラメーターは引き続き有効であり、使用されます。**SQL0020W** メッセージに表示される他のパラメーターは正しくエラーと示され、メッセージが示すとおりは無視されます。これらの他のパラメーターには、以下のものが含まれます。**ACTION**、**CCSIDG**、**CCSIDM**、**CCSIDS**、**CHARSUB**、**CNULREQD**、**DEC**、**DECDEL**、**GENERIC**、**RELEASE**、**REPLVER**、**RETAIN**、**STRDEL**、**TEXT**、**VERSION**、および値 **CONTINUE** が指定された **SQLERROR**。

注: バージョン 7 またはそれ以降のサーバーに接続されており、**VALIDATE RUN** が指定されており、エラーを生成する原因となるがこの **BIND** オプションによってバイパスされるステートメントがある間は、バージョン 6 以前のクライアントはバインド・ファイルを生成しません。ただし、パッケージは作成されます。

アプリケーション・プログラムのバインドには、この資料では紹介していない前提条件と制限事項が含まれています。アプリケーション・プログラムのデータベースへのバインドについての詳細は、 [アプリケーション開発の手引き](#) を参照してください。

SQL ステートメントがエラーであることが検出され、 `BIND` オプション `SQLERROR CONTINUE` が指定されている場合、このステートメントは無効とマークされます。この SQL ステートメントの状態を変えるためには、さらに別の `BIND` を発行する必要があります。暗黙的および明示的な再バインドでは、無効なステートメントの状態は変わりません。 `VALIDATE RUN` でバインドされたパッケージでは、ステートメントは、再バインド時にオブジェクトが存在するかまたは権限の問題があるかどうかに応じて、暗黙的と明示的な再バインドとの両方で、静的バインドから増分バインドに変更したり、増分バインドを静的バインドに変更することができます。

以下も参照

457ページの『[PRECOMPILE PROGRAM](#)』

CALL

カタログで定義されたストアード・プロシージャを呼び出します。

許可

許可規則は、プロシージャが保管されているサーバーごとに異なります。

DB2 ユニバーサル・データベースの場合

実行時の CALL ステートメントの許可 ID が保持する特権には、最低でも以下のいずれかが含まれていなければなりません。

- *Execute* 特権。ストアード・プロシージャと関連するパッケージに対するもの
- *Control* 特権。ストアード・プロシージャと関連するパッケージに対するもの
- *sysadm* または *dbadm* 権限

DB2 ユニバーサル・データベース (OS/390 版) の場合

バインド実行時の CALL ステートメントの許可 ID が保持する特権には、最低でも以下のいずれかが含まれていなければなりません。

- *Execute* 特権。ストアード・プロシージャと関連するパッケージに対するもの
- スタード・プロシージャと関連するパッケージの所有権
- *Packadm* 権限。パッケージのコレクション *Sysadm* 権限に対するもの

DB2 (AS/400 版)

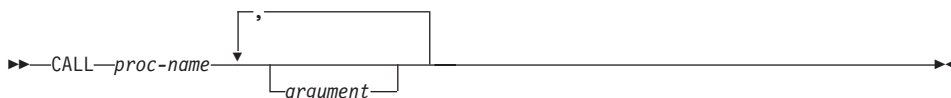
バインド実行時の CALL ステートメントの許可 ID が保持する特権には、最低でも以下のいずれかが含まれていなければなりません。

- スタード・プロシージャが REXX で作成されている場合:
 - プロシージャに関連するソース・ファイルに対するシステム権限 *OBJOPR および *READ
 - ソース・ファイルを含むライブラリーに対するシステム権限 *EXECUTE と、CL コマンドへのシステム権限 *USE
- スタード・プロシージャが REXX で作成されていない場合:
 - プロシージャに関連するプログラムと、そのプログラムの管理権限を含むライブラリーの両方に対するシステム権限 *EXECUTE

必須接続

データベース

コマンド構文



コマンド・パラメーター

proc-name

呼び出されるストアード・プロシージャを識別します。これは、実行するプロシージャ名または完全修飾のスキーマ名にすることができます。たとえば、ユーザー "newton" が "getMedian" という名前のストアード・プロシージャを作成し、コマンド CALL getMedian を発行します。次に、ユーザー "dbuser1" は "getMedian" とも呼ばれるストアード・プロシージャを作成します。"newton" が CALL getMedian を再発行すると、複数のスキーマ (NEWTON および DBUSER1) にこのプロシージャ名が存在するので、このコマンドは失敗します。コマンドを動作させるには、"newton" は CALL NEWTON.getMedian を発行する必要があります。

argument

ストアード・プロシージャに 1 つ以上の引き数を指定します。

CATALOG APPC NODE

CATALOG APPC NODE

ノード・ディレクトリーに APPC ノード項目を追加します。リモート・ノードにアクセスするには、拡張プログラム間通信機能プロトコルを使用します。

許可

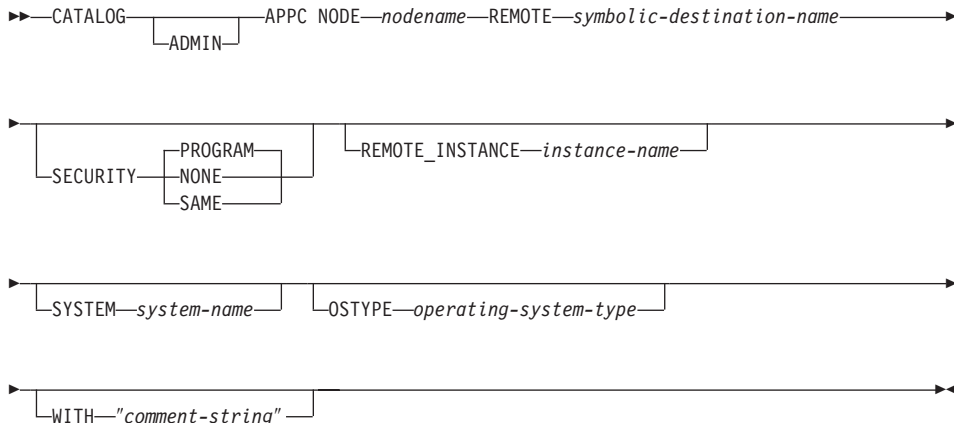
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし

コマンド構文



コマンド・パラメーター

ADMIN

管理サーバー・ノードを指定します。

NODEnodename

カタログ化するノードのローカル別名。これは、ユーザーのワークステーション上の任意名で、ノードを識別する際に使用されます。覚えやすい名前を指定するとよいでしょう。名前は、データベース・マネージャの命名規則に適合していることが必要です (631ページの『付録B. 命名規則』を参照)。

REMOTE symbolic-destination-name

リモート・パートナー・ノードの記号宛先名を指定します。この名前は、クライアントがサーバーへの APPC 接続をセットアップする際に必要な情報を含む、CPI 通信のサイド情報表の項目に対応します (パートナー LU 名、モード名、パートナー TP 名)。最大長は 8 文字です。

SECURITY

パートナー LU に送信する割り振り要求において、安全保護情報を使用する度合いを指定します。有効な値は以下のとおりです。

PROGRAM

パートナー LU に送信する割り振り要求に、ユーザー名とパスワードの両方が含まれるということを指定します。これが省略時値です。

NONE パートナー LU に送信する割り振り要求に、安全保護情報が含まれないということを指定します。

SAME パートナー LU に送信する割り振り要求に、ユーザー名が含まれないということを指定します。これは、ユーザー名が「すでに検査済み」という標識で指定されます。パートナーは、「すでに検査済み」という保証を受け入れられるように構成されていなければなりません。

注: APPC を使用して DB2 (Windows NT 版) バージョン 7.1 (またはそれ以降) サーバーに接続する場合、8 バイトより長いユーザー ID は、SECURITY が NONE に指定されている場合に限りサポートされません。

REMOTE_INSTANCE instance-name

リモート・サーバー・マシン上で接続を確立するインスタンスの実名を指定します。

SYSTEM system-name

サーバー・マシンを識別するために使用する名前を指定します。

OSTYPE operating-system-type

サーバー・マシンのオペレーティング・システムのタイプを指定します。有効な値は次のとおりです。OS2、AIX、WIN95、NT、HPUX、SUN、MVS、OS400、VM、VSE、SNI、SCO、SGI、LINUX、および DYNIX/ptx。

WITH "comment-string"

ノード・ディレクトリー内のノード項目について記述します。ノードに

CATALOG APPC NODE

ついでに記述を補足する任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは、単一引用符または二重引用符で囲む必要があります。

例

```
db2 catalog appc node db2appc1 remote db2inst1 security program
with "A remote APPC node"
```

使用上の注意

データベース・マネージャは、最初のノードがカタログ化されたとき（つまり、CATALOG...NODE コマンドが最初に発行されたとき）にノード・ディレクトリを作成します。OS/2 クライアントまたは Windows クライアントでは、そのクライアントをインストールしたインスタンス・サブディレクトリに、ノード・ディレクトリを保管して維持します。また、AIX クライアントでは、DB2 インストール・ディレクトリにノード・ディレクトリを作成します。

ローカル・ノード・ディレクトリの内容をリストする場合は、391ページの『LIST NODE DIRECTORY』を使用してください。

注: ディレクトリをキャッシュできる場合（292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照）、データベース、ノード、および DCS のディレクトリ・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリ・キャッシュは、最初のディレクトリ参照の間に作成されます。キャッシュはアプリケーションがディレクトリ・ファイルのどれかを修正したときのみ最新にされるため、他のアプリケーションが行ったディレクトリの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリ・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベース・マネージャを停止させてから (**db2stop**)、再始動させます (**db2start**)。別のアプリケーション用のディレクトリ・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

CATALOG APPCLU NODE

通信プロトコルとして APPC を使用するリモート・ワークステーションに関する情報を、ノード・ディレクトリーに書き込みます。DB2 はこの情報を使用して、アプリケーションとこのノードでカタログ作成したリモート・データベース間の接続を確立します。

このコマンドは OS/2 のみで使用可能です。

許可

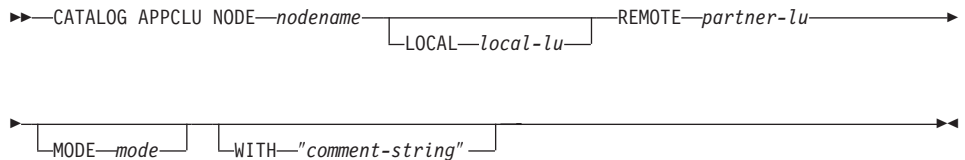
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし

コマンド構文



コマンド・パラメーター

NODE *nodename*

カタログ作成するリモート・ワークステーションの名前を指定します。これは、ワークステーションに常駐しているデータベースのカタログ作成をする場合に、ノード名パラメーターに入力したのと同じの名前です。名前は、DB2 の命名規則に適合していることが必要です (631ページの『付録B. 命名規則』を参照)。

LOCAL *local-lu*

接続に使用した SNA ローカル論理装置の別名を指定します。これは 1 ~ 8 文字の非空白文字からなるストリングでなければなりません。別名は、(コミュニケーション・マネージャー構成の) 対応する SNA 定義に現れるとおりに正確に入力する必要があります (大文字小文字を混合して使用)。

CATALOG APPCLU NODE

REMOTE partner-lu

接続に使用した SNA リモート論理装置の別名を指定します。これは 1 ～ 8 文字の非空白文字からなるストリングでなければなりません。別名は、(コミュニケーション・マネージャー構成の) 対応する SNA 定義に現れるとおりに正確に入力する必要があります (大文字小文字を混合して使用)。

MODE mode

接続で使用した SNA 伝送モードを指定します。名前は、SNA 命名規則に従っていることが必要です。

この値を入力しないと、DB2 は、モード・タイプとして 8 文字の空白を文字ストリングに保管します。

WITH "comment-string"

ノード・ディレクトリー内のノード項目について記述します。ノードについての記述を補足する任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは、単一引用符または二重引用符で囲む必要があります。

例

次の例は、データベースを含むリモート・ワークステーション REMNODE をカタログ作成する方法を示しています。

```
db2 catalog appclu node remnode local LU1 remote LU2
mode SQLMOD01 with "Remote node comment"
```

使用上の注意

このコマンドは、バージョン 1 CATALOG APPC NODE コマンドと同一です。しかし、バージョン 2 CATALOG APPC NODE コマンドとは異なります。

注: CATALOG APPC NODE コマンドを含むバージョン 1 スクリプト・ファイルは、修正する必要があります。キーワード APPC を APPCLU に変更して、バージョン 1 と同様にこのコマンドを実行できるようにしてください。

x'07F6C4C2' 以外の TPNAME を持つサーバーと接続するために、APPCLU ノードで 154ページの『ATTACH』を使用することはできません。

APPCLU ノードはセキュリティー PROGRAM のみをサポートします (セキュリティー SAME および NONE はサポートされていません)。

CATALOG APPN NODE

通信プロトコルとして APPN を使用するリモート・ワークステーションに関する情報を、ノード・ディレクトリーに書き込みます。DB2 はこの情報を使用して、アプリケーションとこのノードでカタログ作成したリモート・データベース間の接続を確立します。

このコマンドは、OS/2、Windows NT、Windows 98、Windows 95、AIX、および Solaris でのみ使用可能です。

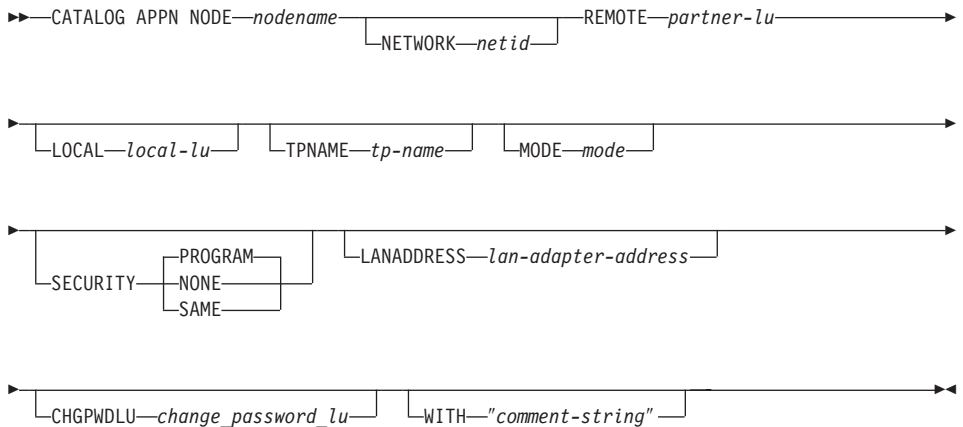
許可

sysadm

必須接続

なし

コマンド構文



コマンド・パラメーター

NODEnodename

カタログ作成するリモート・ワークステーションの名前を指定します。これは、(194ページの『CATALOG DATABASE』を使用して) カタログ化されるワークステーションにデータベースが常駐する場合に、ノード名パラメーターに入力したのと同じ名前です。名前は、DB2 命名規則に適合している必要があります (631ページの『付録B. 命名規則』を参照)。

NETWORK netid

リモート LU が常駐する SNA ネットワークの ID を指定します。このネットワーク ID は、SNA の命名規則に従った 1 ～ 8 文字のストリングです。

REMOTE partner-lu

接続で使用した SNA パートナー論理装置を指定します。リモート・ノードの LU 名を入力してください。名前は、(コミュニケーション・マネージャー構成の) 対応する SNA 定義に現れるとおりに正確に入力する必要があります (大文字小文字を混合して使用)。名前は SNA 命名規則に従っている必要があります。

LOCAL local-lu

接続に使用した SNA ローカル論理装置の別名を指定します。これは 1 ～ 8 文字の非空白文字からなるストリングでなければなりません。別名は、(コミュニケーション・マネージャー構成の) 対応する SNA 定義に現れるとおりに正確に入力する必要があります (大文字小文字を混合して使用)。

TPNAME tp-name

データベース・サーバーの APPC トランザクション・プログラム名を指定します。省略時値は DB2DRDA です。

MODE mode

接続で使用した SNA 伝送モードを指定します。名前は、SNA 命名規則に従っていることが必要です。

この値を入力しないと、DB2 は、モード・タイプとして 8 文字の空白を文字ストリングに保管します。

SECURITY

パートナー LU に送信する割り振り要求において、安全保護情報を使用する度合いを指定します。有効な値は以下のとおりです。

PROGRAM

パートナー LU に送信する割り振り要求に、ユーザー名とパスワードの両方が含まれるということを指定します。これが省略時値です。

NONE パートナー LU に送信する割り振り要求に、安全保護情報が含まれないということを指定します。

SAME パートナー LU に送信する割り振り要求に、ユーザー名が含まれないということを指定します。これは、ユーザー名が「すで

に検査済み」という標識で指定されます。パートナーは、「すでに検査済み」という保証を受け入れられるように構成されていなければなりません。

注: APPN を使用して DB2 (Windows NT 版) バージョン 7.1 (またはそれ以降) サーバーに接続する場合、8 バイトより長いユーザー ID は、SECURITY が NONE に指定されている場合に限りサポートされます。

LANADDRESS lan-adapter-address

DB2 サーバーの LAN アダプター・アドレス。

CHGPWDLU change_password_lu

ホスト・データベース・サーバーのパスワード変更時に使用される、パートナー LU の名前を指定します。

WITH "comment-string"

ノード・ディレクトリー内のノード項目について記述します。ノードについての記述を補足する任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは、単一引用符または二重引用符で囲む必要があります。

例

次の例は、APPN ノードをカタログ化します。

```
db2 catalog appn node remnode remote rlu with "Catalog APPN NODE"
```

CATALOG DATABASE

システム・データベース・ディレクトリーに、データベースのロケーション情報を保管します。データベースは、ローカル・ワークステーションかリモート・ノードのどちらかに位置付けることができます。

効力範囲

区分データベース環境では、ローカル・データベースをシステム・データベース・ディレクトリーにカタログ化するとき、データベースが常駐するサーバー上のノードでこのコマンドを発行する必要があります。

許可

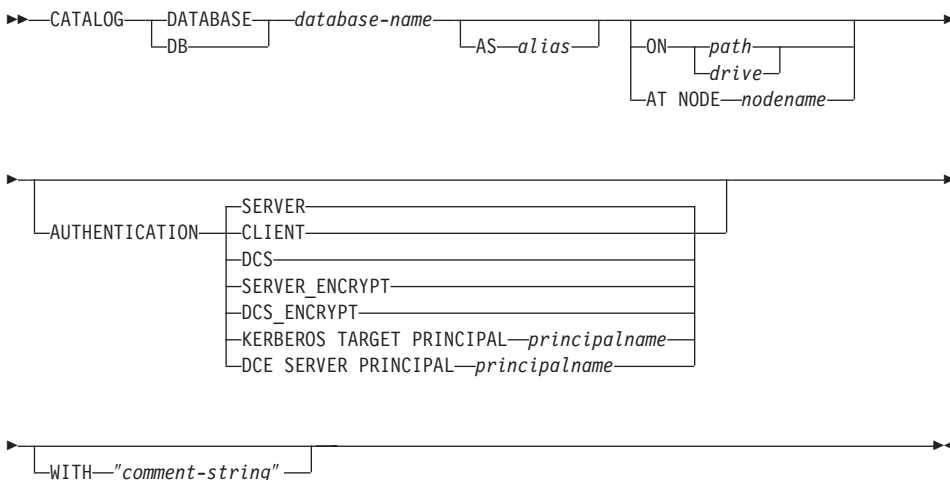
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし。ディレクトリー操作は、ローカル・ディレクトリーだけに影響します。

コマンド構文



コマンド・パラメーター

DATABASE database-name

カタログ化するデータベースの名前を指定します。

AS alias

カタログ化するデータベースの代替名として、別名を指定します。これを指定しなかった場合、データベース・マネージャーは *database-name* を別名として使用します。

ON path/drive

UNIX ベースのシステムでは、カタログ化するデータベースが常駐するパスを指定します。OS/2 または Windows オペレーティング・システムでは、カタログ化するデータベースが常駐するドライブ名を指定します。

AT NODE nodename

カタログ化するデータベースが常駐するノードの名前を指定します。この名前は、ノード・ディレクトリー内の項目名と一致させてください。指定したノード名がノード・ディレクトリーに存在しない場合、警告が戻されますが、データベースはシステム・データベース・ディレクトリーにカタログ化されます。カタログ化したデータベースへの接続が必要なときは、ノード名をノード・ディレクトリーにカタログ化しなければなりません。

AUTHENTICATION

注: DB2 バージョン 2 以降のクライアントが DB2 バージョン 1 のサーバーと通信する場合のみ必要になります。

リモート・データベースの認証値は保管されますが (369ページの『LIST DATABASE DIRECTORY』からの出力に現れる)、ローカル・データベースの認証値は保管されません。

認証タイプを指定すると、パフォーマンスの向上に役立ちます。認証タイプの詳細については、*管理の手引き* を参照してください。

SERVER

認証が、宛先データベースを含むノードで行われるということを指定します。

CLIENT

認証が、アプリケーションを呼び出すノードで行われるということを指定します。

DCS 認証が、宛先データベースを含むノードで行われるということを指定します。ただし、DB2 コネクトの使用時と、認証を DRDA アプリケーション・サーバー (AS) で行うように指定したときを除きます。

SERVER_ENCRYPT

認証が、宛先データベースを含むノードで行われるということ、およびパスワードが送信元で暗号化されることを指定します。送信元でカタログ化される認証タイプによって指定されるとおり、パスワードは宛先で暗号化解除されます。

DCS_ENCRYPT

認証が、宛先データベースを含むノードで行われるということ、を指定します。ただし、DB2 コネクトの使用時は除きます。この場合、認証は DRDA アプリケーション・サーバー (AS) で行われます。送信元でカタログ化される認証タイプによって指定されるとおり、パスワードは送信元で暗号化され、宛先で暗号化解除されます。

KERBEROS

Kerberos セキュリティー・メカニズムを使用して行われる認証を指定します。認証が Kerberos の場合、アクセスには APPC 接続が使用され、SECURITY=NONE のみサポートされます。

TARGET PRINCIPAL principalname

ターゲット・サーバー用の完全修飾 Kerberos プリンシパル名。つまり、DB2 サーバー・サービスのログオン・アカウントで、*userid@xxx.xxx.xxx.com* または *domain#userid* の形式です。

注: このパラメーターは Windows 2000 クライアント上でのみ有効です。

DCE DCE セキュリティー・サービスを使用して行われる認証を指定します。認証が DCE の場合、アクセスには APPC 接続が使用され、SECURITY=NONE のみサポートされます。

SERVER PRINCIPAL principalname

ターゲット・サーバーの完全修飾 DCE 基本名です。この値は、ターゲット・サーバーのキータブ・ファイルにも記録されます。

WITH "comment-string"

システム・データベース・ディレクトリー内のデータベースまたはデータベース項目について記述します。注釈列の最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは必ず二重引用符で囲んでください。

例

```
db2 catalog database sample on /databases/sample  
with "Sample Database"
```

使用上の注意

ローカル・ノードやリモート・ノードにあるデータベースをカタログ化する場合、以前にアンカタログ化したデータベースを再カタログ化する場合、または1つのデータベースに対して複数の別名を保持する場合（データベースのロケーションにかかわらず）、CATALOG DATABASE を使用してください。

データベースを作成したとき、DB2 は自動的にそれらをカタログ化します。ローカル・データベース・ディレクトリーにデータベースの項目、システム・データベース・ディレクトリーに別の項目をカタログ化します。リモート・クライアント（または、同じマシンの別のインスタンスから実行しているクライアント）からデータベースを作成した場合、クライアント・インスタンスでは、システム・データベース・ディレクトリーにも項目が作成されます。

パス名もノード名も指定しなかった場合、データベースはローカルに、データベースのロケーションはデータベース・マネージャー構成パラメーターの *dftdbpath* に指定したものに想定されます。

データベース・マネージャー・インスタンスと同じノードのデータベースは、間接項目としてカタログ化されます。その他のノードのデータベースは、リモート入力としてカタログ化されます。

システム・データベース・ディレクトリーが存在しない場合、CATALOG DATABASE は自動的にそれを作成します。システム・データベース・ディレクトリーは、使用しているデータベース・マネージャー・インスタンスを含むパスに保管され、データベース外部で保守されます。

システム・データベース・ディレクトリーの内容をリストする場合は、369ページの『LIST DATABASE DIRECTORY』を使用してください。

注: ディレクトリーをキャッシュできる場合（292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照）、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときの

CATALOG DATABASE

み最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベース・マネージャーを停止させてから (**db2stop**)、再始動させます (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

以下も参照

587ページの『UNCATALOG DATABASE』

CATALOG DCS DATABASE

リモート・データベースについての情報を、データベース接続サービス (DCS) ディレクトリーに保管します。これらのデータベースは、DB2 コネクトなどのアプリケーション・リクエスター (AR) をとおしてアクセスされます。システム・データベース・ディレクトリー内のデータベース名と一致する名前が DCS ディレクトリー項目にある場合、指定した AR を呼び出して、データベースが常駐するリモート・サーバーに SQL 要求を転送します。DB2 コネクトおよび DCS ディレクトリー項目の詳細は、DB2 コネクト 使用者の手引きを参照してください。

許可

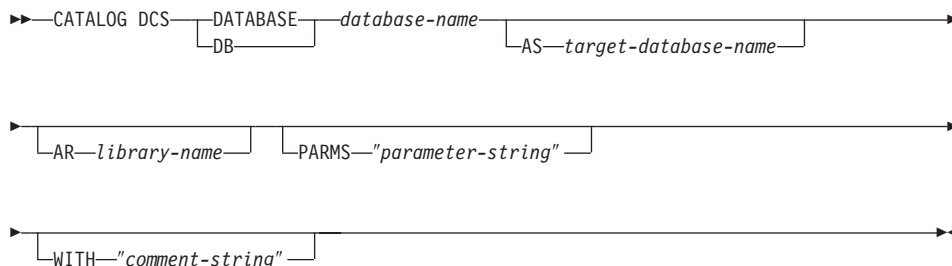
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし

コマンド構文



コマンド・パラメーター

DATABASE database-name

カタログ化する宛先データベースの別名を指定します。この別名は、システム・データベース・ディレクトリーにこのデータベースをリモート・カタログ化した時に入力したデータベース名と一致していなければなりません。

AS target-database-name

カタログ化するターゲット・ホスト・データベースの名前を指定します。

CATALOG DCS DATABASE

AR library-name

DCS ディレクトリーにリストされたりリモート・データベースにアクセスするとき、ロードおよび使用されるアプリケーション・リクエスター・ライブラリーの名前を指定します。

注: DB2 コネクト AR を使用する場合は、ライブラリー名を指定しないでください。省略時で DB2 コネクトを呼び出します。

DB2 コネクトを使用しない場合は、AR のライブラリー名を指定します。そして、そのライブラリーをデータベース・マネージャー・ライブラリーと同じパスに置いてください。OS/2 または Windows オペレーティング・システムでは、そのパスは drive:¥sql1lib¥d11 です。UNIX ベースのシステムでは、パスはインスタンス所有者の \$HOME/sql1lib/lib になります。

PARMS "parameter-string"

呼び出したときに AR にパスされるパラメーター・STRINGを指定します。パラメーター・STRINGは、二重引用符で囲んでください。

WITH "comment-string"

DCS ディレクトリー項目について記述します。このディレクトリーにカタログ化するデータベースについて、任意の注釈を入力できます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは必ず二重引用符で囲んでください。

例

次は、DB1 データベース (DB2 (MVS 版) ホスト・データベース) に関する情報を DCS ディレクトリーにカタログ化する例です。

```
db2 catalog dcs database db1 as dsn_db_1
with "DB2/MVS location name DSN_DB_1"
```

使用上の注意

DB2 コネクト・プログラムは、次のような DRDA アプリケーション・サーバーへの接続を提供します。

- システム/370 およびシステム/390 アーキテクチャー・ホスト・コンピューター上の DB2 (OS/390 版)
- システム/370 およびシステム/390 アーキテクチャー・ホスト・コンピューター上の DB2 (VM および VSE 版)
- AS/400 ホスト・コンピューター上の OS/400 データベース

データベース接続サービスが存在しない場合、データベース・マネージャーは自動的にそれを作成します。このディレクトリーは、使用しているデータベース・マネージャー・インスタンスを含むパスに保管されます。また、データベースの外側で保持されます。

データベースは、システム・データベース・ディレクトリーにリモート・データベースとしてもカタログ化しなければなりません。

DCS ディレクトリーの内容をリストする場合は、378ページの『LIST DCS DIRECTORY』を使用してください。

注: ディレクトリーをキャッシュできる場合 (292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照)、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときのみ最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベース・マネージャーを停止させてから (**db2stop**)、再始動させます (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

以下も参照

589ページの『UNCATALOG DCS DATABASE』

CATALOG GLOBAL DATABASE

DCE タイプのシステム・データベース・ディレクトリー項目を作成します。この項目は、宛先データベースの DCE ディレクトリーにある完全修飾オブジェクト名のローカル別名を定義する際に使用されます。このデータベースについての情報は、集中的に DCE ディレクトリーに保管されます。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし

コマンド構文

```
▶—CATALOG GLOBAL—DATABASE—database-global-name—▶  
                                  └───┬───┘  
                                  DB
```

```
▶—AS—alias—USING DIRECTORY—DCE—WITH—"comment-string"—▶
```

コマンド・パラメーター

DATABASE *database-global-name*

DCE 名スペースのデータベースを固有に識別する完全修飾名を指定します。

AS *alias*

カタログ化するデータベースの代替名を指定します。

USING DIRECTORY DCE

使用するグローバル・ディレクトリー・サービスを指定します。

WITH "comment-string"

システム・データベース・ディレクトリー内の DCE タイプ項目について記述します。このディレクトリーにカタログ化するデータベースについて、任意の注釈を入力できます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは必ず二重引用符で囲ってください。

例

```
db2 catalog global database /.../cell11/subsys/database/DB3
as dbtest using directory dce
```

使用上の注意

database-global-name の最大長は 255 バイトです。この名前は、/.../ または /.:/ で始まらなければなりません。

注: ディレクトリーをキャッシュできる場合 (292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照)、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときのみ最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベース・マネージャーを停止させてから (**db2stop**)、再始動させます (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

CATALOG IPX/SPX NODE

CATALOG IPX/SPX NODE

ノード・ディレクトリーに、インターネットワーク・パケット交換機能 / 順次パケット交換機能 (IPX/SPX) ノード項目を追加します。リモート・ノードにアクセスするには Novell NetWare IPX/SPX 通信プロトコルを使用します。

このコマンドは、OS/2、Windows NT、および Windows 95 でのみ使用可能です。

効力範囲

IPX/SPX ファイル・サーバーのアドレス指定は、マルチノード環境ではサポートされていません。

許可

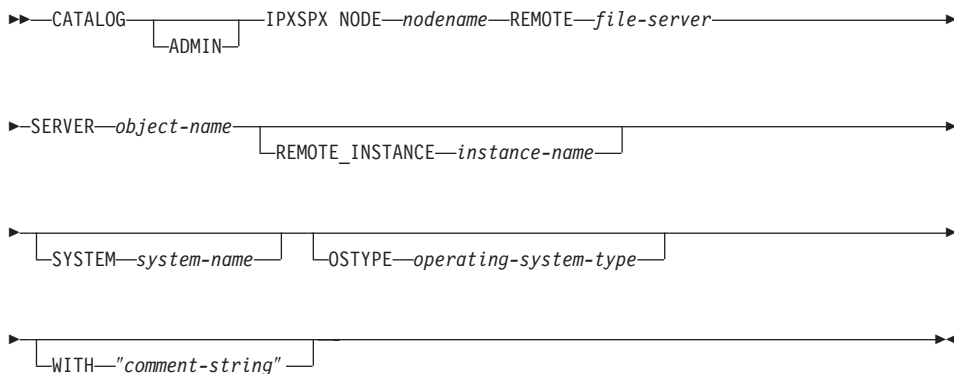
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし

コマンド構文



コマンド・パラメーター

ADMIN

カタログ化される IPX/SPX 管理サーバー・ノードを指定します。

NODEnodename

カタログ化するノードのローカル別名。これは、ユーザーのワークステーション上の任意名で、ノードを識別する際に使用されます。覚えやすい名前を指定するとよいでしょう。名前は、データベース・マネージャーの命名規則に適合していることが必要です (631ページの『付録B. 命名規則』を参照)。

REMOTE file-server

サーバー・データベース・マネージャー・インスタンスのインターネットワーク・アドレスを登録する、NetWare ファイル・サーバーの名前。インターネットワーク・アドレスは、NetWare ファイル・サーバーのバインドリヤーに保管され、*object-name* を使用してアクセスされます。

注: 次の文字は無効です。 / ¥ ; ; , * ?

SERVER object-name

NetWare ファイル・サーバーのバインダーに保管される、データベース・マネージャー・インスタンスの名前。1つのNetWare ファイル・サーバーに登録済みの各サーバー・データベース・マネージャー・インスタンスは、固有の *object-name* によって表示されている必要があります。ネットワーク内の各データベース・マネージャー・インスタンスに、固有の *object-name* を指定することを推奨します。

注: 次の文字は無効です。 / ¥ ; ; , * ?

ファイル・サーバー・アドレス指定を使用して、IPX/SPX クライアントのカタログ作成を行っている場合、ファイル・サーバーおよびオブジェクト名を前述の定義のように指定してください。直接アドレス指定を使用して、IPX/SPX クライアントのカタログ作成を行っている場合、*file-server* には * を指定し、*object-name* パラメーターにはサーバーの IPX/SPX インターネットワーク・アドレスを指定してください。サーバーの IPX/SPX インターネットワーク・アドレスを検索するには、64ページの『db2ipxad - IPX/SPX インターネットワーク・アドレスの獲得』を使用してください。アドレス指定方式の詳細については、概説およびインストール 資料のいずれかを参照してください。

REMOTE_INSTANCE instance-name

接続を確立するサーバー・インスタンスの名前を指定します。

SYSTEM system-name

サーバー・マシンを識別するために使用する DB2 システム名を指定します。

CATALOG IPX/SPX NODE

OSTYPE operating-system-type

サーバー・マシンのオペレーティング・システムのタイプを指定します。有効な値は次のとおりです。OS2、AIX、WIN95、NT、HPUX、SUN、MVS、OS400、VM、VSE、SNI、SCO、SGI、LINUX、および DYNIX。

WITH "comment-string"

ノード・ディレクトリー内のノード項目について記述します。ノードについての記述を補足する任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは、単一引用符または二重引用符で囲む必要があります。

例

```
db2 catalog ipxspx node db2ipx1 remote netwsrv server db2inst1
with "A remote IPX/SPX node"
```

```
db2 catalog ipxspx node db2ipx2 remote * server 09212700.400011527745.879E
with "IPX/SPX node using direct addr"
```

使用上の注意

データベース・マネージャーは、最初のノードがカタログ化されたとき（つまり、CATALOG...NODE コマンドが最初に発行されたとき）にノード・ディレクトリーを作成します。OS/2 クライアントまたは Windows クライアントでは、そのクライアントをインストールしたインスタンス・サブディレクトリーに、ノード・ディレクトリーを保管して維持します。また、AIX クライアントでは、DB2 インストール・ディレクトリーにノード・ディレクトリーを作成します。

ローカル・ノード・ディレクトリーの内容をリストする場合は、391ページの『LIST NODE DIRECTORY』を使用してください。

注: ディレクトリーをキャッシュできる場合（292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター `dir_cache` を参照）、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときのみ最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベース・マネージャーを停止させてから (**db2stop**)、再始動させます (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

CATALOG LDAP DATABASE

CATALOG LDAP DATABASE

Lightweight Directory Access Protocol (LDAP) でデータベースを登録するのに使用します。

このコマンドは、Windows NT、Windows 98、Windows 95、AIX、および Solaris でのみ使用可能です。

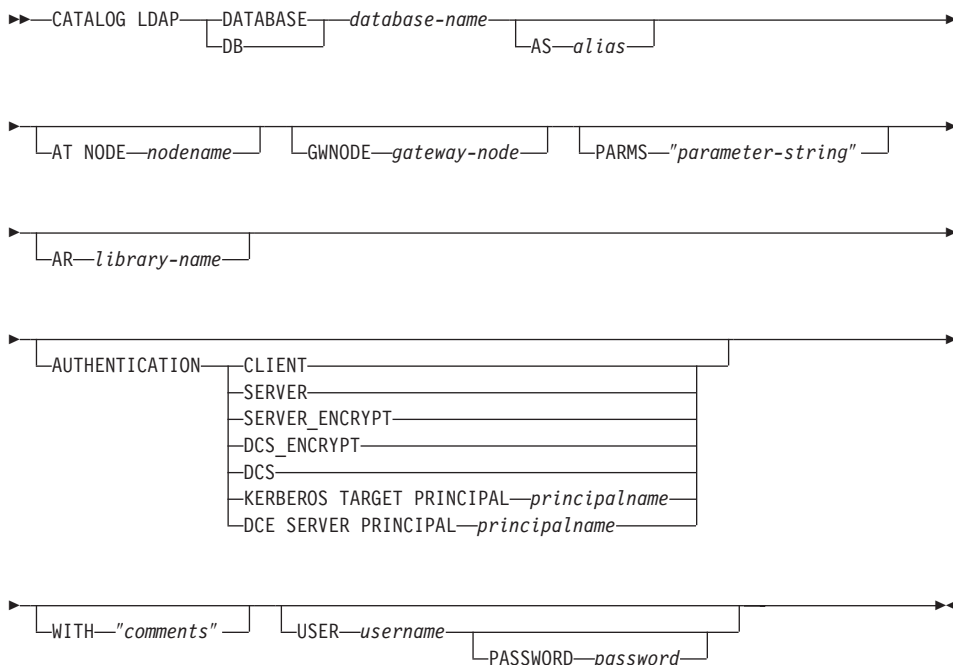
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

DATABASE database-name

カタログ化するデータベースの名前を指定します。

AS alias

カタログ化するデータベースの代替名として、別名を指定します。別名が指定されないと、データベース名が別名として使用されます。

AT NODE nodename

データベースが常駐するデータベース・サーバーに LDAP ノード名を指定します。このパラメーターは、リモート・サーバーでデータベースを登録する場合に指定する必要があります。

GWNODE gateway-node

ゲートウェイ・サーバーに LDAP ノード名を指定します。

PARMS "parameter-string"

DCS データベースへのアクセス時にアプリケーション・リクエスト (AR) に渡される、パラメーター・ストリングを指定します。DDCS が必要とするストリングのフォーマットについての詳細は、DB2 コネクト 使用者の手引き を参照してください。

注: 変更パスワード *sym_dest_name* をパラメーター・ストリングで指定しないでください。LDAP で DB2 サーバーを登録する場合、キーワード **CHGPWDLU** を使用して変更パスワード **LU** 名を指定してください。詳細については、511ページの『REGISTER』を参照してください。

AR library-name

DCS ディレクトリーにリストされたりリモート・データベースにアクセスするとき、ロードおよび使用されるアプリケーション・リクエスト・ライブラリーの名前を指定します。

注: DB2 コネクト **AR** を使用する場合は、ライブラリー名を指定しないでください。省略時値で **DB2** コネクトを呼び出します。

DB2 コネクトを使用しない場合は、**AR** のライブラリー名を指定します。そして、そのライブラリーをデータベース・マネージャー・ライブラリーと同じパスに置いてください。OS/2 または Windows オペレーティング・システムでは、そのパスは `drive:sqllib%dll` です。UNIX ベースのシステムでは、パスはインスタンス所有者の `$HOME/sqllib/lib` になります。

AUTHENTICATION

認証レベルを指定します。パフォーマンスとの関連を含む、認証タイプの詳細については、管理の手引き を参照してください。有効な値は以下のとおりです。

CLIENT

認証が、アプリケーションの呼び出し元であるノードで行われるということを指定します。

SERVER

認証が、宛先データベースを含むノードで行われるということを指定します。

SERVER_ENCRYPT

認証が、宛先データベースを含むノードで行われるということ、およびパスワードが送信元で暗号化されることを指定します。送信元でカタログ化される認証タイプによって指定されるとおり、パスワードは宛先で暗号化解除されます。

DCS_ENCRYPT

認証が、宛先データベースを含むノードで行われるということを指定します。ただし、DB2 コネクトの使用時は除きます。この場合、認証は DRDA アプリケーション・サーバー (AS) で行われます。送信元でカタログ化される認証タイプによって指定されるとおり、パスワードは送信元で暗号化され、宛先で暗号化解除されます。

DCS 認証が、宛先データベースを含むノードで行われるということを指定します。ただし、DB2 コネクトの使用時は除きます。この場合、認証は DRDA アプリケーション・サーバー (AS) で行われます。

KERBEROS

Kerberos セキュリティー・メカニズムを使用して行われる認証を指定します。認証が Kerberos の場合、アクセスには APPC 接続が使用され、SECURITY=NONE のみサポートされます。

TARGET PRINCIPAL *principalname*

ターゲット・サーバー用の完全修飾 Kerberos プリンシパル名。つまり、DB2 サーバー・サービスのログオン・アカウントで、*userid@xxx.xxx.xxx.com* または *domain#userid* の形式です。

注: このパラメーターは Windows 2000 クライアント上でのみ有効です。

DCE DCE セキュリティー・サービスを使用して行われる認証を指定します。認証が DCE の場合、アクセスには APPC 接続が使用され、SECURITY=NONE のみサポートされます。

SERVER PRINCIPAL principalname

ターゲット・サーバーの完全修飾 DCE 基本名です。
この値は、ターゲット・サーバーのキータブ・ファイルにも記録されます。

WITH "comments"

DB2 サーバーを記述します。ネットワーク・ディレクトリーで登録されるサーバーについての記述を補足する、任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは必ず二重引用符で囲んでください。

USER username

ユーザーの LDAP 識別名 (DN) を指定します。LDAP ユーザー DN には、LDAP ディレクトリーでオブジェクトを作成するための十分な権限が必要です。ユーザーの LDAP DN が指定されない場合、現行ログオン・ユーザーの認証が使用されます。

注: ユーザーの LDAP DN およびパスワードが **db2ldcfg** を使用して指定されている場合、ユーザー名とパスワードをここで指定する必要はありません。このコマンドの詳細については、68ページの『db2ldcfg - LDAP 環境の構成』を参照してください。

PASSWORD password

アカウント・パスワード。

注: ユーザーの LDAP DN およびパスワードが **db2ldcfg** を使用して指定されている場合、ユーザー名とパスワードをここで指定する必要はありません。このコマンドの詳細については、68ページの『db2ldcfg - LDAP 環境の構成』を参照してください。

使用上の注意

ノード名が指定されないと、DB2 は現行のマシン上で DB2 サーバーを表す、LDAP の最初のノードを使用します。

次の場合、LDAP でデータベースを手動で登録 (カタログ) する必要があるかもしれません。

- データベース・サーバーが LDAP をサポートしない場合。管理者は、LDAP をサポートするクライアントが、各クライアント・マシン上でローカルにデータベースをカタログ化しなくてもデータベースにアクセスできるように、LDAP で各データベース手動で登録する必要があります。

CATALOG LDAP DATABASE

- アプリケーションが、データベースに接続するために異なる名前を使用する必要がある場合。この場合、管理者は異なる別名を使用してデータベースにカタログできます。
- データベースがホスト・データベース・サーバー (たとえば、DB2/390、DB2/400 など) にある場合。この場合、管理者は LDAP でデータベースを登録し、GWNODE パラメーターを介してゲートウェイ・ノードを指定することができます。
- CREATE DATABASE IN LDAP の実行中、データベース名がすでに LDAP に存在する場合。この場合でもデータベースはローカル・マシン上に作成されます (かつローカル・アプリケーションによってアクセスできる) が、LDAP に存在するエントリは新しいデータベースを反映して変更されることはありません。この場合、管理者は次のことを行えます。
 - LDAP の既存のデータベース・エントリを除去し、手動で新しいデータベースを LDAP に登録する。
 - 異なる別名を使って LDAP で新しいデータベースを登録する。

以下も参照

213ページの『CATALOG LDAP NODE』

591ページの『UNCATALOG LDAP DATABASE』

593ページの『UNCATALOG LDAP NODE』

CATALOG LDAP NODE

Lightweight Directory Access Protocol (LDAP) で新しいノード・エントリーをカタログ化します。

このコマンドは、Windows NT、Windows 98、Windows 95、AIX、および Solaris でのみ使用可能です。

許可

なし

必須接続

なし

コマンド構文

```

▶▶ CATALOG LDAP NODE nodename AS nodealias

```

```

┌── USER username ───┐
│                       │
│ ┌── PASSWORD password ───┐
│ │                           │
│ │                           │
│ └──────────────────────────┘

```

コマンド・パラメーター

NODE *nodename*

DB2 サーバーの LDAP ノード名を指定します。

AS *nodealias*

LDAP ノード・エントリーに新しい別名を指定します。

USER *username*

ユーザーの LDAP 識別名 (DN) を指定します。LDAP ユーザー DN には、LDAP ディレクトリーでオブジェクトを作成するための十分な権限が必要です。ユーザーの LDAP DN が指定されない場合、現行ログオン・ユーザーの認証が使用されます。

PASSWORD *password*

アカウント・パスワード。

使用上の注意

CATALOG LDAP NODE コマンドは、DB2 サーバーを表すノードに、異なる別名を指定するのに使用されます。

CATALOG LDAP NODE

以下も参照

208ページの『CATALOG LDAP DATABASE』

591ページの『UNCATALOG LDAP DATABASE』

593ページの『UNCATALOG LDAP NODE』

CATALOG LOCAL NODE

同一のマシンに常駐するインスタンスのローカル別名を作成します。同じワークステーション上にユーザーのクライアントからアクセスされる複数のインスタンスがある場合、ローカル・ノードをカタログ化する必要があります。ローカル・ノードにアクセスするために、プロセス間通信 (IPC) が使用されます。

許可

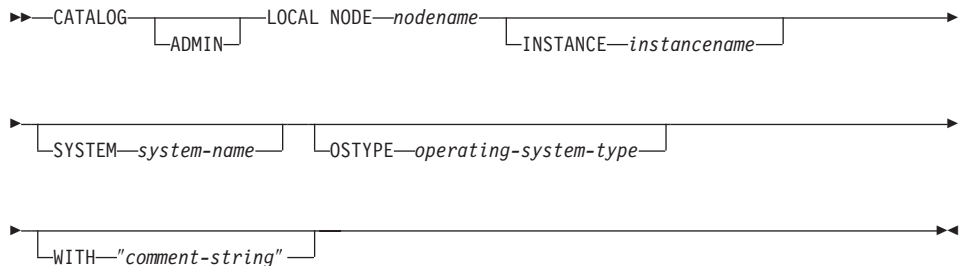
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし

コマンド構文



コマンド・パラメーター

ADMIN

カタログ化するローカル管理サーバー・ノードを指定します。

NODEnodename

カタログ化するノードのローカル別名。これは、ユーザーのワークステーション上の任意名で、ノードを識別する際に使用されます。覚えやすい名前を指定するとよいでしょう。名前は、データベース・マネージャーの命名規則に適合していることが必要です (631ページの『付録B. 命名規則』を参照)。

INSTANCE instancename

アクセスするローカル・インスタンスの名前。

CATALOG LOCAL NODE

SYSTEM *system-name*

サーバー・マシンを識別するために使用する DB2 システム名を指定します。

OSTYPE *operating-system-type*

サーバー・マシンのオペレーティング・システムのタイプを指定します。有効な値は次のとおりです。OS2、AIX、WIN95、NT、HPUX、SUN、MVS、OS400、VM、VSE、SNI、SCO、SGI、LINUX、および DYNIX。

WITH "*comment-string*"

ノード・ディレクトリー内のノード項目について記述します。ノードについての記述を補足する任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは、単一引用符または二重引用符で囲む必要があります。

例

ワークステーション A には、inst1 および inst2 の 2 つのサーバー・インスタンスがあります。単一 CLP セッションから両方のインスタンスにデータベースを作成するには、次のコマンドを出してください (**DB2INSTANCE** 環境変数が inst1 に設定されているものと想定)。

1. inst1 にローカル・データベースを作成します。

```
db2 create database mydb1
```

2. このワークステーションに別のサーバー・インスタンスをカタログ化します。

```
db2 catalog local node mynode2 instance inst2
```

3. mynode2 にデータベースを作成します。

```
db2 attach to mynode2  
db2 create database mydb2
```

使用上の注意

注: ディレクトリーをキャッシュできる場合 (292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照)、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときの

み最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベース・マネージャーを停止させてから (**db2stop**)、再始動させます (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

CATALOG NAMED PIPE NODE

CATALOG NAMED PIPE NODE

ノード・ディレクトリーに名前付きパイプ・ノード項目を追加します。リモート・ノードにアクセスするときに、この名前付きパイプを使用します。

このコマンドは Windows NT でのみ使用可能です。

許可

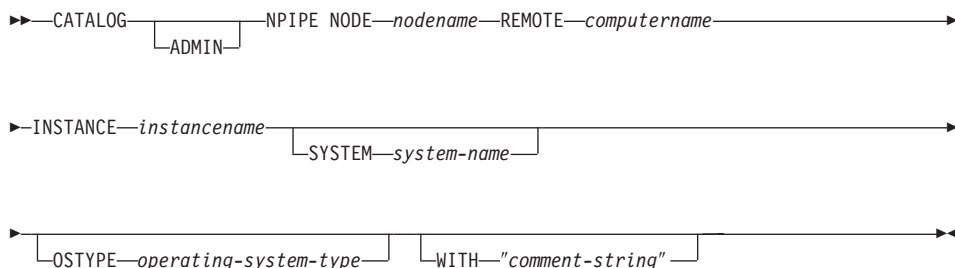
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし

コマンド構文



コマンド・パラメーター

ADMIN

NPIPE 管理サーバー・ノードをカタログ化することを指定します。

NODEnodename

カタログ化するノードのローカル別名。これは、ユーザーのワークステーション上の任意名で、ノードを識別する際に使用されます。覚えやすい名前を指定するとよいでしょう。名前は、データベース・マネージャーの命名規則に適合していることが必要です (631ページの『付録B. 命名規則』を参照)。

REMOTecomputername

宛先データベースが常駐するノードのコンピューター名です。最大長は 15 文字です。

INSTANCE instancename

宛先データベースが常駐するサーバー・インスタンスの名前です。リモート・ノードと通信する際に使用する、リモート名前付きパイプの名前と同じにしてください。

SYSTEM system-name

サーバー・マシンを識別するために使用する DB2 システム名を指定します。

OSTYPE operating-system-type

サーバー・マシンのオペレーティング・システムのタイプを指定します。有効な値は次のとおりです。OS2、AIX、WIN95、NT、HPUX、SUN、MVS、OS400、VM、VSE、SNI、SCO、SGI、LINUX、および DYNIX。

WITH "comment-string"

ノード・ディレクトリー内のノード項目について記述します。ノードについての記述を補足する任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは、単一引用符または二重引用符で囲む必要があります。

例

```
db2 catalog npipe node db2np1 remote nphost instance db2inst1
with "A remote named pipe node."
```

使用上の注意

データベース・マネージャーは、最初のノードがカタログ化されたとき（つまり、CATALOG...NODE コマンドが最初に発行されたとき）にノード・ディレクトリーを作成します。OS/2 クライアントまたは Windows クライアントでは、そのクライアントをインストールしたインスタンス・サブディレクトリーに、ノード・ディレクトリーを保管して維持します。また、AIX クライアントでは、DB2 インストール・ディレクトリーにノード・ディレクトリーを作成します。

ローカル・ノード・ディレクトリーの内容をリストする場合は、391ページの『LIST NODE DIRECTORY』を使用してください。

注: ディレクトリーをキャッシュできる場合（292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照）、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはア

CATALOG NAMED PIPE NODE

アプリケーションがディレクトリー・ファイルのどれかを修正したときのみ最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベース・マネージャーを停止させてから (**db2stop**)、再始動させます (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

CATALOG NETBIOS NODE

ノード・ディレクトリーに NetBIOS ノード項目を追加します。リモート・ノードにアクセスするときは、NetBIOS 通信プロトコルを使用します。

このコマンドは、OS/2、Windows NT、および Windows 95 でのみ使用可能です。

許可

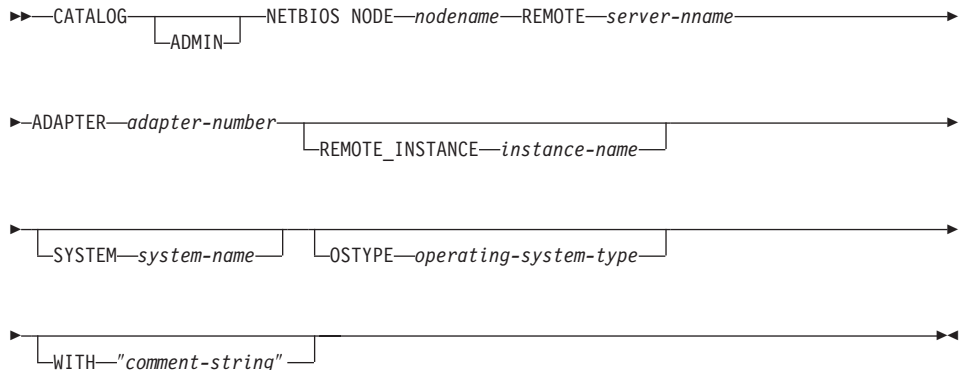
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし。ディレクトリー操作は、ローカル・ディレクトリーだけに影響します。

コマンド構文



コマンド・パラメーター

ADMIN

管理サーバー・ノードを指定します。

NODEnodename

カタログ化するノードのローカル別名。これは、ユーザーのワークステーション上の任意名で、ノードを識別する際に使用されます。覚えやすい名前を指定するとよいでしょう。名前は、データベース・マネージャーの命名規則に適合している必要があります (631ページの『付録B. 命名規則』を参照)。

CATALOG NETBIOS NODE

REMOTE server-nname

宛先データベースが常駐するリモート・ワークステーションの名前。この名前は、データベース・マネージャーの命名規則に適合しなければなりません。これは、サーバー・ワークステーションのデータベース・マネージャー構成ファイルにあるワークステーション名 (*nname*) です。

ADAPTER adapter-number

ローカル、論理、出力 LAN アダプター番号を指定します。省略時値はゼロです。

REMOTE_INSTANCE instance-name

リモート・サーバー・マシン上で接続を確立するインスタンスの実名を指定します。

SYSTEM system-name

サーバー・マシンを識別するために使用する名前を指定します。

OSTYPE operating-system-type

サーバー・マシンのオペレーティング・システムのタイプを指定します。有効な値は次のとおりです。OS2、AIX、WIN95、NT、HPUX、SUN、MVS、OS400、VM、VSE、SNI、SCO、SGI、LINUX、および DYNIX。

WITH "comment-string"

ノード・ディレクトリー内のノード項目について記述します。ノードについての記述を補足する任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは、単一引用符または二重引用符で囲む必要があります。

例

```
db2 catalog netbios node db2netb1 remote db2inst1 adapter 0
with "A remote NetBIOS node"
```

使用上の注意

データベース・マネージャーは、最初のノードがカタログ化されたとき (つまり、CATALOG...NODE コマンドが最初に発行されたとき) にノード・ディレクトリーを作成します。OS/2 クライアントまたは Windows クライアントでは、そのクライアントをインストールしたインスタンス・サブディレクトリーに、ノード・ディレクトリーを保管して維持します。また、AIX クライアントでは、DB2 インストール・ディレクトリーにノード・ディレクトリーを作成します。

ローカル・ノード・ディレクトリーの内容をリストする場合は、391ページの『LIST NODE DIRECTORY』を使用してください。

注: ディレクトリーをキャッシュできる場合 (292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照)、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときのみ最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベース・マネージャーを停止させてから (**db2stop**)、再始動させます (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

CATALOG ODBC DATA SOURCE

CATALOG ODBC DATA SOURCE

ユーザーまたはシステム ODBC データ・ソースをカタログ化します。

ODBC (オープン・データベース・コネクティビティー) でのデータ・ソースという語は、指定したデータベースまたはファイル・システムのユーザー定義名のことです。この名前は、ODBC API を介してデータベースまたはファイル・システムにアクセスするときに使用されます。Windows NT および Windows 95 では、ユーザー・データ・ソースまたはシステム・データ・ソースのどちらであってもカタログ化できます。ユーザー・データ・ソースはそれをカタログ化したユーザーにのみ可視になりますが、システム・データ・ソースは他のすべてのユーザーから可視であり使用可能です。

このコマンドは、Windows NT、Windows 95、および Windows 3.1 でのみ使用可能です。

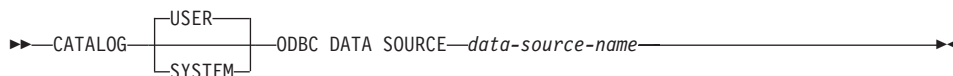
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

USER ユーザー・データ・ソースをカタログ化します。キーワードを指定しない場合、これが省略時値です。

SYSTEM

システム・データ・ソースをカタログ化します。

ODBC DATA SOURCE data-source-name

カタログ化するデータ・ソースの名前を指定します。最大長は 32 文字です。

以下も参照

398ページの『LIST ODBC DATA SOURCES』

597ページの『UNCATALOG ODBC DATA SOURCE』

CATALOG TCP/IP NODE

CATALOG TCP/IP NODE

ノード・ディレクトリーに、伝送制御プロトコル / インターネット・プロトコル (TCP/IP) ノード項目を追加します。リモート・ノードにアクセスするときは、TCP/IP 通信プロトコルを使用します。

許可

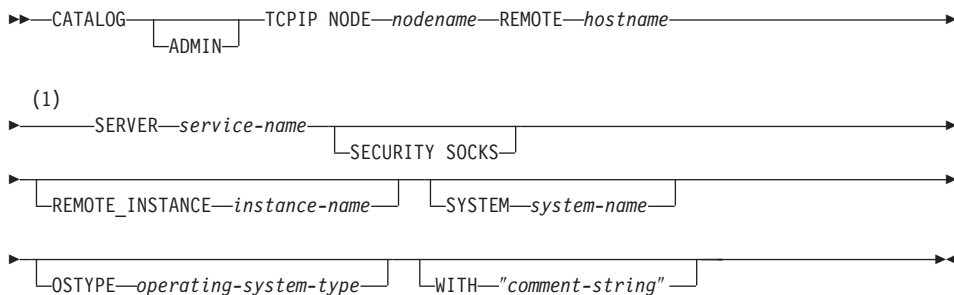
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし。ディレクトリー操作は、ローカル・ディレクトリーだけに影響します。

コマンド構文



注:

- 1 ADMIN ノードには SERVER を指定してはならない。ADMIN ノード以外には必須。

コマンド・パラメーター

ADMIN

カタログ化する TCP/IP 管理サーバー・ノードを指定します。

NODEnodename

カタログ化するノードのローカル別名。これは、ユーザーのワークステーション上の任意名で、ノードを識別する際に使用されます。覚えやす

い名前を指定するとよいでしょう。名前は、データベース・マネージャの命名規則に適合していることが必要です (631ページの『付録B. 命名規則』を参照)。

REMOTE hostname

宛先データベースが常駐するノードのホスト名。このホスト名とは、TCP/IP ネットワークに認識されているノードの名前のことです。最大長は 255 文字です。

SERVER service-name

サーバー・データベース・マネージャー・インスタンスのサービス名またはポート番号を指定します。

CATALOG TCPIP NODE コマンドは、クライアントで実行されます。

- サービス名を指定する場合、クライアントのサービス・ファイルは、そのサービス名をポート番号にマップするのに使用されます。サービス名は、データベース・マネージャー構成ファイルで指定します。サーバーのサービス・ファイルは、このサービス名をポート番号にマップするのに使用されます。クライアントとサーバーのポート番号は一致していなければなりません。

注: サーバーのデータベース・マネージャー構成ファイルでサービス名の代わりにポート番号を指定することもできますが、お勧めできません。

- ポート番号を指定する場合、それはサーバーのデータベース・マネージャー構成ファイルで指定したサービス名に関連付けられているポート番号と一致していなければなりません。なお、ローカル TCP/IP サービス・ファイルに、サービス名を指定する必要はありません。

service-name の値は、対応するポート番号のローカル・サービス・ファイルを探索するキーとして使用されます。一致する項目を検出できず、かつ *service-name* の値が数値の場合、その値はポート番号と解釈されます。

最大長は 14 文字です。このパラメーターには、大文字小文字の区別があります。

注: ADMIN ノードでは、このパラメーターを指定してはなりません。ADMIN ノードの値は、常に 523 です。

SECURITY SOCKS

ノードを SOCKS 使用可能に指定します。

CATALOG TCP/IP NODE

以下の環境変数は必須で、使用可能 SOCKS に設定しなければなりません。

SOCKS_NS

SOCKS サーバーのホスト・アドレスを解決するための定義域名サーバーです。これは IP アドレスでなければなりません。

SOCKS_SERVER

完全修飾ホスト名または SOCKS サーバーの IP アドレスです。完全修飾ホスト名を解決するために SOCKS 化した DB2 クライアントを使用できない場合、IP アドレスがすでに入力されたと想定されます。

以下の条件の少なくとも 1 つが真である必要があります。

- SOCKS サーバーは、定義域名サーバー経由で到達可能でなければならない。
- hosts ファイル中にリストされていなければならない。このファイルのロケーションは、TCP/IP ドキュメンテーションに説明されています。
- IP アドレス形式でなければならない。

db2start を発行した後にこれらの環境変数を設定した場合、**TERMINATE** コマンドを発行することが必要になります。

REMOTE_INSTANCE instance-name

接続を確立するサーバー・インスタンスの名前を指定します。

SYSTEM system-name

サーバー・マシンを識別するために使用する DB2 システム名を指定します。

OSTYPE operating-system-type

サーバー・マシンのオペレーティング・システムのタイプを指定します。有効な値は次のとおりです。OS2、AIX、WIN95、NT、HPUX、SUN、MVS、OS400、VM、VSE、SNI、SCO、SGI、LINUX、および DYNIX。

WITH "comment-string"

ノード・ディレクトリー内のノード項目について記述します。ノードについての記述を補足する任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは、単一引用符または二重引用符で囲む必要があります。

例

```
db2 catalog tcpip node db2tcp1 remote tcpghost server db2inst1
with "A remote TCP/IP node"
```

```
db2 catalog tcpip node db2tcp2 remote 9.21.15.235 server db2inst2
with "TCP/IP node using IP address"
```

使用上の注意

データベース・マネージャーは、最初のノードがカタログ化されたとき（つまり、CATALOG...NODE コマンドが最初に発行されたとき）にノード・ディレクトリーを作成します。OS/2 クライアントまたは Windows クライアントでは、そのクライアントをインストールしたインスタンス・サブディレクトリーに、ノード・ディレクトリーを保管して維持します。また、AIX クライアントでは、DB2 インストール・ディレクトリーにノード・ディレクトリーを作成します。

ローカル・ノード・ディレクトリーの内容をリストする場合は、391ページの『LIST NODE DIRECTORY』を使用してください。

注: ディレクトリーをキャッシュできる場合（292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照）、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときのみ最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベース・マネージャーを停止させてから (**db2stop**)、再始動させます (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

CHANGE DATABASE COMMENT

CHANGE DATABASE COMMENT

システム・データベース・ディレクトリーまたはローカル・データベース・ディレクトリー内の、データベースの注釈を変更します。現行の注釈関連テキストは、新規の注釈テキストと置き換えることができます。

効力範囲

このコマンドは、それが実行されたノードに対してだけ影響を与えます。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし

コマンド構文



コマンド・パラメーター

DATABASE database-alias

注釈を変更するデータベースの別名を指定します。システム・データベース・ディレクトリー内の注釈を変更するには、そのデータベースの別名を指定します。また、ローカル・データベース・ディレクトリー内の注釈を変更するには、そのデータベースが常駐するパスを指定し (*path* パラメーターで)、データベース名 (別名ではない) を入力します。

ON path/drive

UNIX ベースのシステムでは、データベースが常駐するパスを指定して、ローカル・データベース・ディレクトリー内の注釈を変更します。パスを指定しなかった場合、システム・データベース・ディレクトリー内の項目のデータベース注釈が変更されます。OS/2 または Windows オペレーティング・システムでは、データベースが常駐するドライブ名を指定します。

WITH "comment-string"

システム・データベース・ディレクトリーまたはローカル・データベース・ディレクトリー内の項目について記述します。カタログ化したデータベースについての記述を補足する、任意の注釈を入力することができます。注釈列の最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは必ず二重引用符で囲んでください。

例

以下は、SAMPLE データベースのシステム・データベース・ディレクトリー注釈テキストを、"Test 2 - Holding" から "Test 2 - Add employee inf rows" に変更する例です。

```
db2 change database sample comment
with "Test 2 - Add employee inf rows"
```

使用上の注意

既存の注釈テキストは、新規のテキストに置き換えられます。情報を追加する場合、既存の注釈テキストに続けて新規テキストを入力してください。

データベース別名と関連する項目の注釈だけが修正されます。データベース名が同じでも、別名が異なるその他の項目には影響しません。

パスを指定する場合、データベース別名をローカル・データベース・ディレクトリーにカタログ化してください。また、パスを指定しない場合は、データベース別名をシステム・データベース・ディレクトリーにカタログ化してください。

以下も参照

235ページの『CREATE DATABASE』

CHANGE ISOLATION LEVEL

CHANGE ISOLATION LEVEL

データベースのアクセス中に、DB2 が他の処理からデータを分離する方法を変更します。

許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

TO

- CS** カーソル固定を分離レベルとして指定します。
- NC** コミットを分離レベルとして指定しません。 DB2 ではサポートされていません。
- RR** 反復可能読み取りを分離レベルとして指定します。
- RS** 読み取り固定を分離レベルとして指定します。
- UR** コミットされていない読み取りを分離レベルとして指定します。

使用上の注意

DB2 は、分離レベルを使用して、データベース中でデータの保全性を維持します。分離レベルは、並行して実行される他のアプリケーション処理によって加えられる変更から、アプリケーション処理が分離（シールド）される程度を定義します。

選択された分離レベルがデータベースでサポートされていない場合、接続時に、サポートされているレベルまで自動的に調整されます。

タイプ 1 の接続でデータベースへ接続中に、分離レベルを変更することは許可されていません (566ページの『SET CLIENT』を参照)。バックエンド処理は、分離レベルを変更する前に以下のようにして終了しなければなりません。

```
db2 terminate
db2 change isolation to ur
db2 connect to sample
```

タイプ 2 の接続を使用した変更は許可されますが、変更は同じコマンド行プロセッサのバックエンド処理からのすべての接続に適用されるので、注意が必要です。ユーザーは、どの分離レベルが、接続されたどのデータベースに適用するか、記憶している責任があるということを前提とします。

次の例では、SAMPLE データベースの作成に続いて、ユーザーが DB2 対話式モードに入っています。

```
update command options using c off
catalog db sample as sample2

set client connect 2

connect to sample
connect to sample2

change isolation to cs
set connection sample
declare c1 cursor for select * from org
open c1
fetch c1 for 3 rows

change isolation to rr
fetch c1 for 2 rows
```

c1 がこの分離レベルに対応した準備状態になっていないため、SQL0514N エラーが発生します。

```
change isolation to cs
set connection sample2
fetch c1 for 2 rows
```

c1 がこのデータベースに対応した準備状態になっていないため、SQL0514N エラーが発生します。

```
declare c1 cursor for select division from org
```

カーソル c1 がすでに宣言されて開いているため、DB21029E エラーが発生します。

```
set connection sample
fetch c1 for 2 rows
```

CHANGE ISOLATION LEVEL

この場合は、元のデータベース (SAMPLE) が元の分離レベル (CS) で使用されたので、うまくいきます。

分離レベルの詳細については、*SQL 解説書* を参照してください。

以下も参照

489ページの『QUERY CLIENT』

CREATE DATABASE

新しいデータベースを任意選択のユーザー定義照合順序で初期化し、3つの初期表スペースを作成し、システム表を作成し、回復ログを割り振ります。

このコマンドはクライアントでは無効です。

効力範囲

マルチノード環境では、このコマンドは `db2nodes.cfg` ファイルにリストされているノードすべてに影響を与えます。

このコマンドを発行したノードは、新規データベースのカタログ・ノードになります。

許可

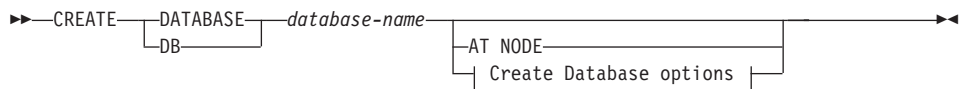
以下のどれかが必要です。

- `sysadm`
- `sysctrl`

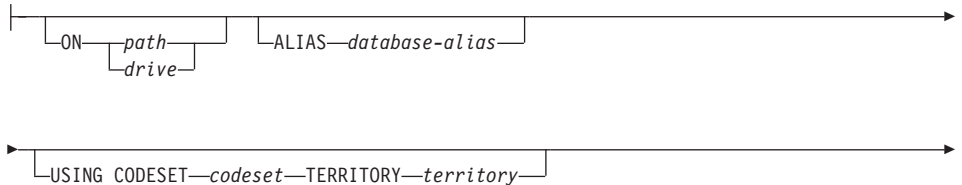
必須接続

インスタンス。別の (リモート) ノードでデータベースを作成するには、まずそのノードに接続する必要があります。このコマンドの処理中、データベース接続が一時的に確立します。

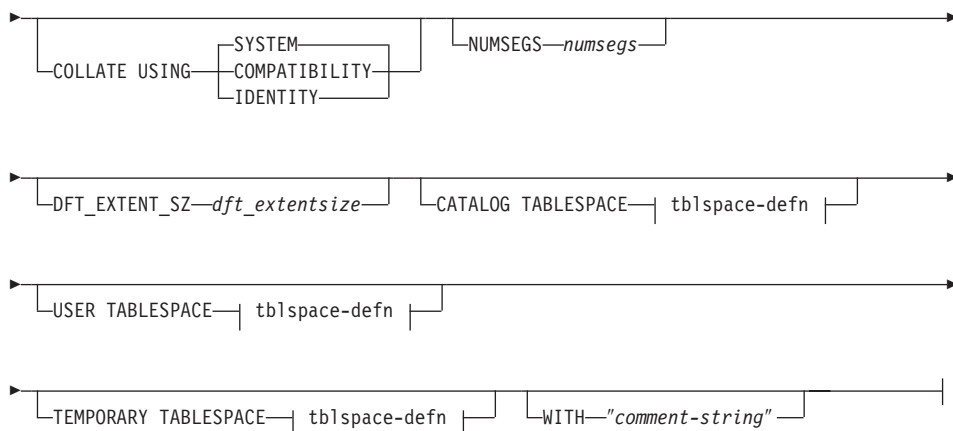
コマンド構文



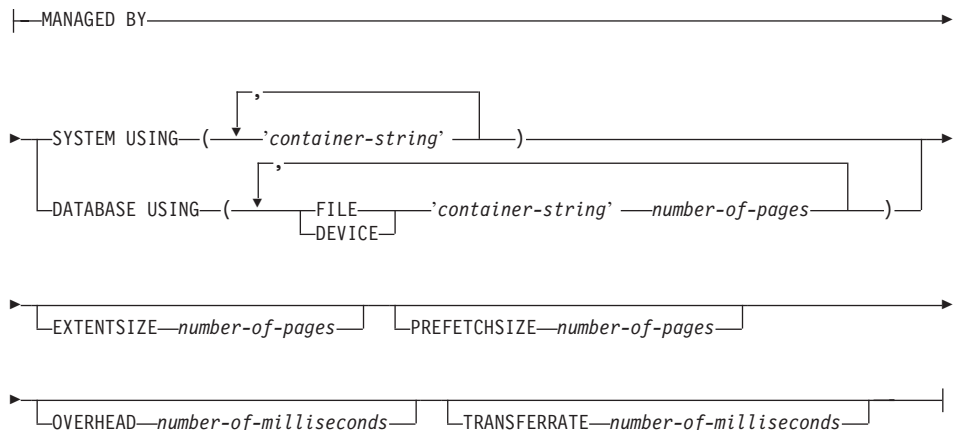
Create Database options:



CREATE DATABASE



tblspace-defn:



注:

1. 指定されるコード設定およびテリトリ値は有効な組み合わせでなければなりません。有効な組み合わせのリストについては、概説およびインストール資料のいずれかを参照してください。
2. **tblspace-defn** パラメーターの詳細については、*SQL 解説書* の CREATE TABLESPACE ステートメントを参照してください。CREATE DATABASE で指定した表スペース定義は、データベースを作成しているすべてのノードに適用されます。定義をノードごとに個別に指定することはできません。表スペース定義を特定のノードごとに異なるものにして作成する場合、CREATE TABLESPACE ステートメントを使用しなければなりません。

表スペースにコンテナを定義する場合、\$N を使用できます。\$N は、コンテナが実際に作成されるときにノード番号で置き換えられます。これは、ユーザーが複数の論理ノード・データベースでコンテナを指定する場合に必要です。

コマンド・パラメーター

DATABASE database-name

新しいデータベースに割り当てられる名前。ローカル・データベース・ディレクトリーまたはシステム・データベース・ディレクトリーの他のデータベースから、そのデータベースを区別する、固有の名前でなければなりません。名前は、データベースの命名規則に適合していることが必要です。

AT NODE

データベースが、コマンドを発行したノードでのみ作成されることを指定します。このパラメーターは、通常の使用を意図したものではありません。たとえば、あるノードでデータベース区画が損傷を受けたため再作成する必要がある場合、このパラメーターを 539ページの『RESTORE DATABASE』とともに使用します。このパラメーターの使用が不適切であると、システム内に不整合が生じることもありえますので、ご使用の際には十分注意してください。

注: (損傷を受けたために) 消去されたデータベース区画の再作成にこのパラメーターを使用する場合、このノードのデータベースは復元の保留状態になります。データベース区画を再作成した後で、データベースはただちにこのノードで復元されます。

ON path/drive

UNIX ベースのシステムでは、データベースを作成するパスを指定します。パスを指定しないと、データベースはデータベース・マネージャー構成ファイル (*dftdbpath* パラメーター) に指定されている省略時値のデータベース・パスに作成されます。最大長は 205 文字です。OS/2 または Windows オペレーティング・システムでは、データベースを作成するドライブの文字を指定します。

注: MPP システムでは、データベースを NFS マウント・ディレクトリーに作成しないようにしてください。パスを指定しない場合、*dftdbpath* データベース・マネージャー構成パラメーターが NFS マウント・パスに設定されていないことを確認してください (たとえば、UNIX ベースのシステムの場合は、パラメーターがインス

CREATE DATABASE

タンス所有者の \$HOME ディレクトリーを指定しないようにします)。MPP システムでは、このコマンドに相対パスを指定することはできません。

ALIAS database-alias

システム・データベース・ディレクトリーのデータベースに付けられる別名。別名が付けられないと、指定されたデータベース名が使用されません。

USING CODESET codeset

このデータベースに入るデータに使用するコード設定を指定します。

TERRITORY territory

このデータベースに入るデータに使用するテリトリーを指定します。

COLLATE USING

データベースに使用する照合順序のタイプを識別します。一度データベースが作成してしまうと、照合順序を変更することはできません。

COMPATIBILITY

DB2 バージョン 2 の照合順序です。一部の照合表が拡張されています。このオプションは、それらの表の直前のバージョンを使用することを指定します。

IDENTITY

ストリングがバイト単位で比較される、照合順序を認識しません。

SYSTEM

現行のテリトリーに基づいた照合順序。

データベース照合順序が使用される方法の詳細については、*SQL 解説書* を参照してください。

NUMSEGS numsegs

省略時 SMS 表スペース用に DAT、IDX、LF、LB、および LBA ファイルを保管する際に作成および使用するセグメント・ディレクトリーの数を指定します。このパラメーターは、DMS 表スペース、作成特性が明示的に指定された SMS 表スペース (データベース作成時に作成される)、またはデータベース作成後に明示的に作成された SMS 表スペースには影響を与えません。

DFT_EXTENT_SZ dft_extentsize

データベース内の表スペースの省略時エクステンツ・サイズを指定します。

CATALOG TABLESPACE tblspace-defn

カタログ表 SYSCATSPACE を保持する、表スペースの定義を指定します。指定しないと、SYSCATSPACE はディレクトリーの *numsegs* 数をコンテナーとして使用し、また *dft_extentsize* のエクステント・サイズを使用して、システム管理スペース (SMS) 表スペースとして作成されます。たとえば、*numsegs* に 5 を指定した場合、以下のコンテナーが作成されます。

```
/u/smith/smith/NODE0000/SQL00001/SQLT0000.0
/u/smith/smith/NODE0000/SQL00001/SQLT0000.1
/u/smith/smith/NODE0000/SQL00001/SQLT0000.2
/u/smith/smith/NODE0000/SQL00001/SQLT0000.3
/u/smith/smith/NODE0000/SQL00001/SQLT0000.4
```

MPP システムの場合、カタログ表スペースはカタログ・ノード (CREATE DATABASE を発行するノード) でのみ作成されます。

USER TABLESPACE tblspace-defn

初期ユーザー表スペース USERSPACE1 の定義を指定します。指定しないと、USERSPACE1 はコンテナーとしてディレクトリーの *numsegs* 数を使用し、また *dft_extentsize* のエクステント・サイズを使用して、SMS 表スペースとして作成されます。たとえば、*numsegs* に 5 を指定した場合、以下のコンテナーが作成されます。

```
/u/smith/smith/NODE0000/SQL00001/SQLT0001.0
/u/smith/smith/NODE0000/SQL00001/SQLT0001.1
/u/smith/smith/NODE0000/SQL00001/SQLT0001.2
/u/smith/smith/NODE0000/SQL00001/SQLT0001.3
/u/smith/smith/NODE0000/SQL00001/SQLT0001.4
```

TEMPORARY TABLESPACE tblspace-defn

初期システム一時表スペース、TEMPSPACE1 の定義を指定します。指定しないと、TEMPSPACE1 はディレクトリーの *numsegs* 数をコンテナーとして使用し、また *dft_extentsize* のエクステント・サイズを使用して、SMS 表スペースとして作成されます。たとえば、*numsegs* に 5 を指定した場合、以下のコンテナーが作成されます。

```
/u/smith/smith/NODE0000/SQL00001/SQLT0002.0
/u/smith/smith/NODE0000/SQL00001/SQLT0002.1
/u/smith/smith/NODE0000/SQL00001/SQLT0002.2
/u/smith/smith/NODE0000/SQL00001/SQLT0002.3
/u/smith/smith/NODE0000/SQL00001/SQLT0002.4
```

WITH "comment-string"

データベース・ディレクトリー内のデータベース項目について記述します。そのデータベースについての記述を補足する、任意の注釈を入力す

CREATE DATABASE

ることができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは、単一引用符または二重引用符で囲む必要があります。

使用上の注意

CREATE DATABASE には以下の特徴があります。

- 指定したサブディレクトリーにデータベースを作成します。MPP システムでは、`db2nodes.cfg` にリストされたすべてのノードでデータベースを作成し、各ノードの指定されたサブディレクトリーの下に、`$DB2INSTANCE/NODExxxx` ディレクトリーを作成してください。非 MPP システムでは、指定されたサブディレクトリーの下に `$DB2INSTANCE/NODE0000` ディレクトリーを作成してください。
- システム・カタログ表および回復ログを作成します。
- 次のデータベース・ディレクトリーでデータベースをカタログ化します。

- *path* によって示されるパスにサーバーのローカル・データベース・ディレクトリー、または、パスが指定されていない場合には、データベース管理システム構成ファイルで定義される省略時値のデータベース・パス。ローカル・データベース・ディレクトリーは、データベースが入っている各ファイル・システムに常駐しています。
- 接続したインスタンスのサーバーのシステム・データベース・ディレクトリー。結果のディレクトリー項目には、データベース名とデータベース別名が入ることになります。

コマンドがリモート・クライアントから発行された場合、クライアントのシステム・データベース・ディレクトリーもデータベース名と別名で更新されます。

システムまたはローカル・データベース・ディレクトリーがどちらも存在しない場合に作成します。指定されていれば、注釈およびコード・セットは両方のディレクトリーに入れられます。

- 指定されたコード設定、領域、および照合順序を保管します。照合順序が固有の重みで構成される場合、またはそれが識別順序である場合、データベース構成ファイルにフラグが設定されます。
- SYSIBM を所有者として、SYSCAT、SYSFUN、SYSIBM、および SYSSTAT というスキーマを作成します。このコマンドを発行したサーバー・ノードが、新規データベースのカタログ・ノードになります。IBMDEFAULTGROUP および IBMCATGROUP の 2 つのノードグループが自動的に作成されます。詳細については、*SQL 解説書* を参照してください。

- 以前に定義されたデータベース管理バインド・ファイルをデータベースにバインドします (このリストは、ユーティリティのバインド・ファイル・リスト `db2ubind.lst` にあります)。これらのファイルの 1 つ以上が正常にバインドされない場合、`CREATE DATABASE` は `SQLCA` に警告を返し、失敗したバインドについての情報を提供します。バインドが失敗した場合、ユーザーは訂正の処置をとり、手動で失敗したファイルをバインドできます。データベースはどのような場合にでも作成されます。`CREATEIN` 特権が `PUBLIC` に付与されたバインドの実行時に、`NULLID` と呼ばれるスキーマが暗黙的に作成されます。

注: ユーティリティ・バインド・ファイル・リストには、下位レベルのサーバーに対してバインドできない 2 つのファイルが含まれています。

- `db2ugtpi.bnd` は、DB2 バージョン 2 サーバーに対してバインドすることができません。
- `db2dropv.bnd` は、DB2 パラレル・エディション バージョン 1 サーバーに対してバインドすることができません。

`db2ubind.lst` が下位レベルのサーバーに対してバインドされていると、これらの 2 つのファイルに関係した警告が戻されますが、これは無視することができます。

- `SYSCATSPACE`、`TEMPSPACE1`、および `USERSPACE1` 表スペースを作成します。`SYSCATSPACE` 表スペースはカタログ・ノードでのみ作成されません。
- 以下の権限や許可を付与します。
 - `DBADM` 権限、および `CONNECT`、`CREATETAB`、`BINDADD`、`CREATE_NOT_FENCED`、`IMPLICIT_SCHEMA`、および `LOAD` 特権をデータベース作成者に。
 - `CONNECT`、`CREATETAB`、`BINDADD`、および `IMPLICIT_SCHEMA` 特権を `PUBLIC` に。
 - `USERSPACE1` 表スペースの `USE` 特権を `PUBLIC` に。
 - 各システム・カタログ上の `SELECT` 特権を `PUBLIC` に。
 - 正常にバインドされたユーティリティに対する `BIND` および `EXECUTE` 特権を `PUBLIC` に。

`dbadm` 権限を使用すると、これらの権限を他のユーザーまたは `PUBLIC` に付与 (または取り消し) することができます。データベース上の `sysadm` 権限、または `dbadm` 権限を持つ別の管理者がそれらの特権を取り消す場合でも、データベース作成者はそれらを保存します。

CREATE DATABASE

MPP 環境では、データベース・マネージャーが、すべてのノードの指定したパスまたは省略時パスの下にサブディレクトリーの `$DB2INSTANCE/NODExxxx` を作成します。 `xxxx` は `db2nodes.cfg` ファイルで定義されたノード番号です (つまり、ノード 0 が `NODE0000` になる)。サブディレクトリー `SQL00001` ~ `SQLnnnnn` は、このパスに常駐します。これにより、異なるノードに関連したデータベース・オブジェクトが異なるディレクトリーに (指定したパスまたは省略時パスの下にサブディレクトリー `$DB2INSTANCE` が、すべてのノードで共通だとしても) 保管されることが保証されます。

LDAP (Lightweight Directory Access Protocol) サポートが現行のマシン上で使用可能である場合、データベースは自動的に LDAP ディレクトリーに登録されます。同じ名前のデータベース・オブジェクトがすでに LDAP ディレクトリーに存在している場合でも、データベースはローカル・マシンに作成されますが、名前の対立があることを示す警告メッセージが戻されます。この場合、ユーザーは 208ページの『CATALOG LDAP DATABASE』を使用して、LDAP データベース・エントリーを手動でカタログ化することができます。

CREATE DATABASE は、アプリケーションがすでにデータベースに接続されている場合、失敗します。

CATALOG DATABASE を使用して、新しいデータベースに異なる別名を定義してください。

以下も参照

165ページの『BIND』

194ページの『CATALOG DATABASE』

251ページの『DROP DATABASE』

DEACTIVATE DATABASE

指定したデータベースを停止します。

効力範囲

MPP システムの場合、このコマンドはシステム内のすべてのノードで、指定したデータベースを非活動化します。1 つまたは複数のノードでエラーが検出されると、警告が戻されます。データベースが正常に非活動化されるノードもありますが、エラーが検出されたノードではデータベースはそのまま活動状態を継続することがあります。

許可

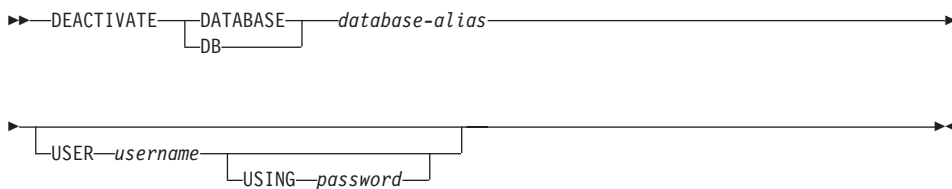
以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

なし

コマンド構文



コマンド・パラメーター

DATABASE database-alias

停止するデータベースの別名を指定します。

USER username

データベースを停止するユーザーを指定します。

USING password

ユーザー ID のパスワードを指定します。

DEACTIVATE DATABASE

使用上の注意

ACTIVATE DATABASE で初期化したデータベースは、DEACTIVATE DATABASE または **db2stop** によって遮断することができます。ACTIVATE DATABASE でデータベースを初期化した場合、そのデータベースから最後のアプリケーションが切断されてもデータベースは遮断されないため、DEACTIVATE DATABASE を使用する必要があります。(この場合、**db2stop** を使用してデータベースを遮断することもできます。)

注: DEACTIVATE DATABASE コマンドを発行するアプリケーションは、どのデータベースへも活動データベース接続を持つことができません。

以下も参照

148ページの『ACTIVATE DATABASE』

582ページの『STOP DATABASE MANAGER』

DEREGISTER

DB2 サーバーの登録をネットワーク・ディレクトリー・サーバーから取り消します。

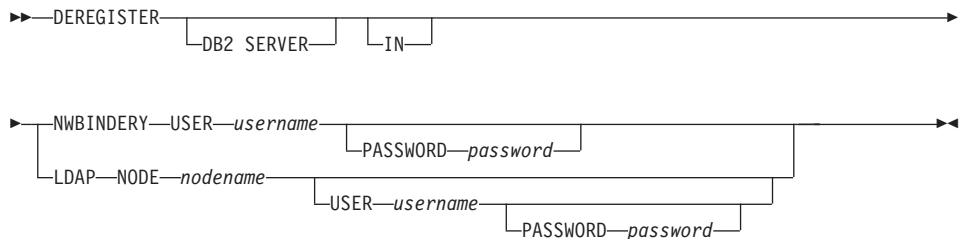
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

IN DB2 サーバーを登録解除するネットワーク・ディレクトリー・サーバーを指定します。有効な値は次のとおりです。 NetWare バインダリーの場合 NWBINDERY、LDAP (Lightweight Directory Access Protocol) ディレクトリー・サーバーの場合 LDAP。

USER username

NWBINDERY の場合、これはネットワーク・サーバーにログインするときに使用するユーザー ID です。ユーザー ID には、SUPERVISOR またはワークグループ管理者と同等のセキュリティーが必要です。ユーザー名は、NetWare ディレクトリー・サーバーから登録を解除するときに提供する必要があり、NETWARE サーバーにログインするためのユーザー ID です。LDAP の場合、ユーザーの LDAP 識別名 (DN) です。LDAP ユーザー DN には、LDAP ディレクトリーからオブジェクトを削除するための十分な権限が必要です。ユーザー名は、LDAP での登録解除時には任意指定です。ユーザーの LDAP DN が指定されない場合、現行ログオン・ユーザーの認証が使用されます。

PASSWORD password

アカウント・パスワード。

DEREGISTER

NODEnodename

ノード名は、DB2 サーバーが LDAP で登録されるときに指定される値です。

使用上の注意

このコマンドは、LDAP 環境にあるリモート・マシンにしか発行できません。リモート・マシンに発行される場合、リモート・サーバーのノード名を指定する必要があります。

DB2 サーバーは、インスタンスが除去されるときに自動的に登録解除されます。

以下も参照

511ページの『REGISTER』

610ページの『UPDATE LDAP NODE』

DESCRIBE

このコマンドは以下の事柄を行います。

- SELECT ステートメントに関する SQLDA 情報の表示
- 表または視点の列の表示
- 表または視点の索引の表示

許可

SELECT ステートメントに関する SQLDA 情報を表示するには、SELECT ステートメント中で参照された表または視点ごとに、以下にリストされた特権または権限の 1 つが必要になります。

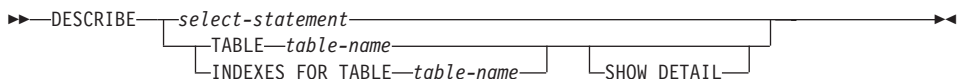
表または視点の列または索引を表示するには、システム・カタログ SYSCAT.COLUMNS (DESCRIBE TABLE) および SYSCAT.INDEXES (DESCRIBE INDEXES FOR TABLE) に対して、以下にリストされた特権または権限の 1 つが必要になります。

- SELECT 特権
- CONTROL 特権
- *sysadm* または *dbadm* の権限

必須接続

データベース。暗黙接続が可能な場合には、省略時データベースへの接続が確立されます。

コマンド構文



コマンド・パラメーター

select-statement

情報が必要なステートメントを識別します。SELECT ステートメントは CLP によって自動的に準備されます。

TABLE table-name

記述する表または視点を指定します。完全修飾の名前または別名の書式は、*schema.table-name* を使用しなければなりません。*schema* はユーザー名で、その下に表または視点が作成されます。

DESCRIBE

DESCRIBE TABLE コマンドは、各列に関する以下の情報をリストします。

- 列名
- タイプ・スキーマ
- タイプ名
- 長さ
- 位取り
- ヌル値 (yes/no)

注: このオプションは、宣言一時表に対してはサポートされていません。

INDEXES FOR TABLE *table-name*

索引を記述する必要がある表または視点を指定します。完全修飾の名前または別名の書式は、*schema.table-name* を使用しなければなりません。*schema* はユーザー名で、その下に表または視点が作成されます。

DESCRIBE INDEXES FOR TABLE コマンドは、その表または視点の各索引に関する以下の情報をリストします。

- 索引スキーマ
- 索引名
- 固有の規則
- 列カウント

SHOW DETAIL

DESCRIBE TABLE コマンドの場合、出力に以下の追加情報を含めることを指定します。

- CHARACTER、VARCHAR または LONG VARCHAR 列のどれかが FOR BIT DATA として定義されたかどうか
- 列番号
- 区分化キー・シーケンス
- コード・ページ
- 省略時値

DESCRIBE INDEXES FOR TABLE コマンドは、出力に以下の追加情報を含めることを指定します。

- 列名

例

SELECT ステートメントの記述

次に示すのは、SELECT ステートメントを記述する方法の一例です。

```
db2 "describe select * from staff"
```

SQLDA Information

```
sqldaid : SQLDA      sqldabc: 896  sqln: 20  sqld: 7
```

Column Information

sqltype	sqllen	sqlname.data	sqlname.length
500 SMALLINT	2	ID	2
449 VARCHAR	9	NAME	4
501 SMALLINT	2	DEPT	4
453 CHARACTER	5	JOB	3
501 SMALLINT	2	YEARS	5
485 DECIMAL	7, 2	SALARY	6
485 DECIMAL	7, 2	COMM	4

表の記述

次に示すのは、表を記述する方法の一例です。

```
db2 describe table user1.department
```

Table: USER1.DEPARTMENT

Column name	Type schema	Type name	Length	Scale	Nulls
AREA	SYSIBM	SMALLINT	2	0	No
DEPT	SYSIBM	CHARACTER	3	0	No
DEPTNAME	SYSIBM	CHARACTER	20	0	Yes

表索引の記述

次に示すのは、表索引を記述する方法の一例です。

```
db2 describe indexes for table user1.department
```

DETACH

Table: USER1.DEPARTMENT

Index schema	Index name	Unique rule	Number of columns
USER1	IDX1	U	2

DETACH

論理 DBMS インスタンス接続を除去し、この層を使用した論理接続がほかがない場合、物理通信接続を終了します。

許可

なし

必須接続

なし。既存のインスタンス接続を除去します。

コマンド構文

▶▶—DETACH—◀◀

コマンド・パラメーター

なし

以下も参照

154ページの『ATTACH』

DROP DATABASE

データベースの内容とそのすべてのログ・ファイルを削除し、データベースをアンカタログし、さらにデータベースのサブディレクトリーを削除します。

効力範囲

省略時では、このコマンドは `db2nodes.cfg` ファイル内にリストされているノードすべてに影響を与えます。

許可

以下のどれかが必要です。

- `sysadm`
- `sysctrl`

必須接続

インスタンス。明示的な接続は必要ありません。データベースがリモートとして示されている場合、リモート・ノードへのインスタンス接続はコマンドの持続期間の間、ずっと確立されたままになります。

コマンド構文



コマンド・パラメーター

DATABASE database-alias

ドロップするデータベースの別名を指定します。データベースはシステム・データベース・ディレクトリー内にカタログ化されている必要があります。

AT NODE

DROP DATABASE コマンドを発行したノードだけで、データベースが削除されることを指定します。このパラメーターは 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に付属のユーティリティーが使用するもので、汎用ではありません。このパラメーターの使用が不適切であると、システム内に不整合が生じることもありえますので、ご使用の際には十分注意してください。

DROP DATABASE

例

次の例は、データベース別名 `SAMPLE` で参照されるデータベースを削除します。

```
db2 drop database sample
```

使用上の注意

`DROP DATABASE` はすべてのユーザー・データとログ・ファイルを削除します。復元操作後のロールフォワード回復にログ・ファイルが必要である場合、このコマンドを発行する前にファイルを保管しておく必要があります。

データベースは使用中であってはなりません。データベースをドロップできるようになるには、すべてのユーザーがデータベースから切断していなければなりません。

ドロップするためには、データベースがシステム・データベース・ディレクトリーにカタログ化されている必要があります。指定されたデータベース別名だけがシステム・データベース・ディレクトリーから除去されます。同じデータベースに対して他の別名が存在する場合、その項目はそのままです。ドロップしようとするデータベースがローカル・データベース・ディレクトリーの最後の項目である場合、ローカル・データベース・ディレクトリーは自動的に削除されます。

`DROP DATABASE` がリモート・クライアント（または同一マシンの別のインスタンス）から出される場合、指定された別名はクライアントのシステム・データベース・ディレクトリーから除去されます。それに対応するデータベース名は、サーバーのシステム・データベース・ディレクトリーから除去されません。

このコマンドは、`DATALINK` 列を介してリンクされているすべてのファイルをリンク解除します。リンク解除操作は `DB2 データ・リンク・マネージャー` で非同期に実行されるので、効果が `DB2 データ・リンク・マネージャー` で即時に見られたり、リンク解除されたファイルが即時に他の操作で使用できるようになったりするとは限りません。このコマンドを発行するときは、データベースに構成されたすべての `DB2 データ・リンク・マネージャー` が使用可能でなければなりません。そうでない場合、データベースのドロップ操作は失敗します。

以下も参照

194ページの『CATALOG DATABASE』

235ページの『CREATE DATABASE』

587ページの『UNCATALOG DATABASE』

DROP NODE VERIFY

任意のデータベースのノードグループ中にノードが存在するかどうか、およびそのノードで事象モニターが定義されているかどうかを検査します。このコマンドは、ノードを MPP システムから消去する前に使用します。

効力範囲

このコマンドは、それが発行されたノードに対してだけ影響を与えます。

許可

sysadm

コマンド構文

▶—DROP NODE VERIFY—▶

コマンド・パラメーター

なし

使用上の注意

ノードが使用中ではないことを示すメッセージが戻された場合、582ページの『STOP DATABASE MANAGER』に DROP NODENUM を指定して使用し、db2nodes.cfg ファイルからそのノードの項目を除去してください。これでそのノードはデータベース・システムから除去されます。

そのノードが使用中であることを示すメッセージが戻された場合、以下のアクションをとる必要があります。

1. そのノードにデータがある場合、505ページの『REDISTRIBUTE NODEGROUP』を使用してデータを再分配し、ノードからデータを除去します。DROP NODE オプションを指定した REDISTRIBUTE NODEGROUP コマンドまたは ALTER NODEGROUP ステートメントのどちらかを使用して、データベースのすべてのノードグループからそのノードを除去します。このアクションを、ノードグループ中にノードを含むデータベースごとに行う必要があります。詳細については、SQL 解説書を参照してください。
2. ノードで定義されている事象モニターをすべて消去します。
3. DROP NODE VERIFY を実行し、データベースが使用中ではなくなったことを確認してください。

以下も参照

582ページの『STOP DATABASE MANAGER』

ECHO

ECHO

ユーザーが文字ストリングを標準出力に書き込めるようにします。

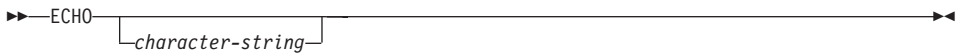
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

character-string

任意の文字列。

使用上の注意

標準入力として入力ファイルが使用される場合、または注釈がコマンド・シェルの解釈を受けずに出力される場合、ECHO コマンドは文字ストリングを標準出力に直接出力します。

ECHO を出す度に 1 行が出力されます。

ECHO コマンドは `verbose (-v)` オプションの影響を受けません (125ページの『コマンド行プロセッサのオプション』を参照)。

EXPORT

データベースから、いくつかある外部ファイル形式のどれかにデータをエクスポートします。ユーザーは、SQL SELECT ステートメントを提供するか、タイプ表の階層情報を提供して、エクスポートするデータを指定します。

許可

以下のどれかが必要です。

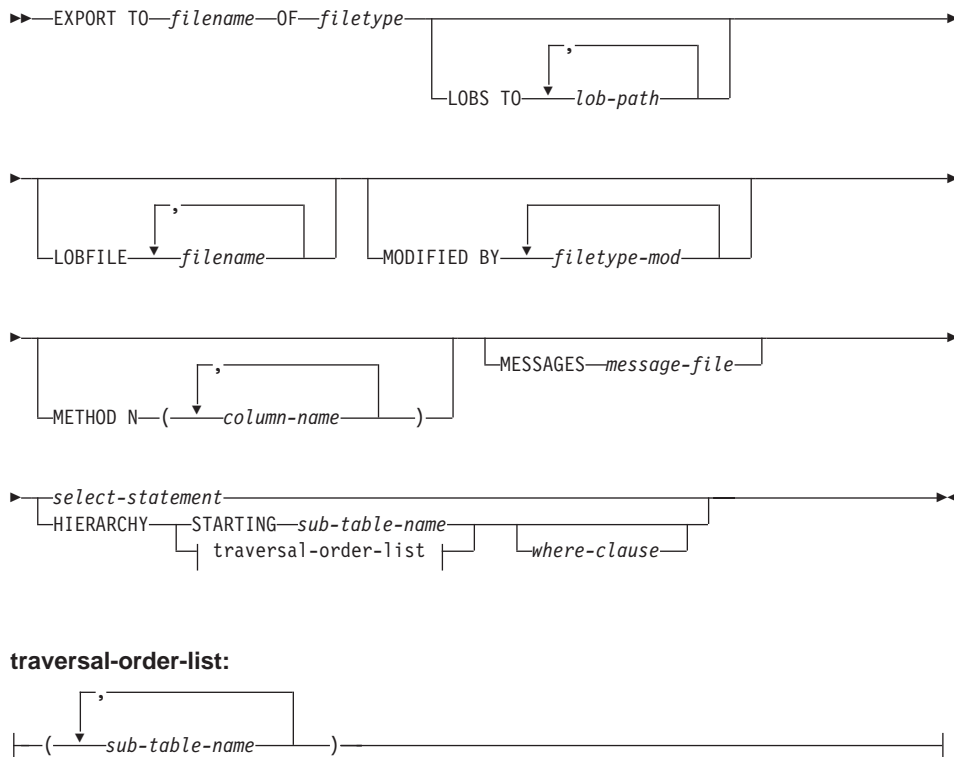
- *sysadm*
- *dbadm*

または、関係する各表または視点に対する CONTROL または SELECT 特権

必須接続

データベース。暗黙接続が可能な場合には、省略時データベースへの接続が確立されます。

コマンド構文



コマンド・パラメーター

HIERARCHY traversal-order-list

指定した走査順序を使用して副階層をエクスポートします。すべての副表は、PRE-ORDER 方式でリストされていなければなりません。最初の副表名が、SELECT ステートメントのターゲット表名として使用されます。

HIERARCHY STARTING sub-table-name

省略時の走査順序 (ASC、DEL、または WSF ファイルの OUTER 順序、または PC/IXF データ・ファイルに保管されている順序) を使用して、*sub-table-name* から始まる副階層をエクスポートします。

LOBFILE filename

LOB ファイルに 1 つ以上の基本ファイル名を指定します。最初の名前の名前スペースがいっぱいになると、2 番目の名前が使用され、以下 3 番目、4 番目と続きます。

エクスポート操作中に LOB ファイルを作成すると、まずこのリストから現行パス (*lob-path* で指定されたパス) に現行の基本ファイル名を追加し、それに 3 桁の順序番号を追加したファイル名が構成されます。現行 LOB パスがディレクトリー /u/foo/lob/path で、現行 LOB ファイル名が bar の場合、LOB ファイルは、/u/foo/lob/path/bar.001、 /u/foo/lob/path/bar.002 (以下 003、004 と続く) といった名前と順序で作成されます。

LOBS TO lob-path

LOB ファイルが保管される、ディレクトリーへの 1 つ以上のパスを指定します。最初のパスのファイル・スペースがいっぱいになると、2 番目のパスが使用され、以下 3 番目、4 番目と続きます。

MESSAGES message-file

エクスポート操作中に生じ得る警告およびエラー・メッセージの宛先を指定します。宛先ファイルがすでに存在している場合、エクスポート・ユーティリティーは情報を追加します。 *message-file* を省略すると、メッセージは標準出力に書き込まれます。

METHOD N column-name

出力ファイルで使用される 1 つ以上の列名を指定します。このパラメーターが指定されない場合、表の列名が使用されます。このパラメーターは WSF および IXF ファイルでのみ有効ですが、階層データをエクスポートするときは無効です。

MODIFIED BY filetype-mod

追加オプションを指定します (263ページの表5 を参照)。

OF filetype

次のような出力ファイルのデータ形式を指定します。

- DEL (区切り文字付き ASCII 形式)。さまざまなデータベース・マネージャーやファイル・マネージャーで使用します。
- WSF (ワークシート形式)。以下のプログラムで使用します。
 - ロータス 1-2-3
 - Lotus Symphony

注: BIGINT または DECIMAL データをエクスポートする場合、タイプ DOUBLE の範囲内の値のみが正確にエクスポートされます。この範囲内にはない値もエクスポートされますが、オペレーティング・システムによっては、これらの値のインポートまたはエクスポートの結果、データに間違いが生じる場合があります。

- SELECT ステートメントで列が指定してある場合を除き、ほとんどの表属性である IXF (統合交換フォーマット、PC バージョン) と、既存の索引が IXF ファイルに保管されます。このフォーマットを使うと、表は再作成されます。一方、他のファイル形式を使用する場合、データをそこにインポートするには表が存在していなければなりません。IXF ファイル形式での表の属性のリストについては、*データ移動ユーティリティー 手引きおよび解説書* の「インポート」の章にある「エクスポートされた表の再作成」を参照してください。

ファイル形式の詳細については、*データ移動ユーティリティー 手引きおよび解説書* の付録『Export/Import/Load ユーティリティー・ファイル形式』を参照してください。

select-statement

エクスポートされるデータを戻す SELECT ステートメントを指定します。SELECT ステートメントによってエラーが発生する場合、メッセージ・ファイル (または標準出力) にメッセージが書き込まれます。エラー・コードが SQL0012W、SQL0347W、SQL0360W、SQL0437W、または SQL1824W である場合、エクスポート操作は続行します。これ以外のエラー・コードの場合、操作は停止します。

TO filename

データのエクスポート先のファイルの名前を指定します。このファイルへの完全パスが指定されていない場合、エクスポート・ユーティリティーは現行のディレクトリーおよび省略時のドライブを宛先として使用します。

すでに存在するファイルの名前を指定した場合、エクスポート・ユーティリティーはファイルの内容を上書きします。情報の追加は行いません。

例

次に示すのは、SAMPLE データベース (このデータベースにユーザーは接続している必要がある) にある STAFF 表から、myfile.ixf に IXF 形式の出力で情報をエクスポートする方法の一例です。データベース接続が DB2 コネクトを介していない場合、索引定義 (もしあれば) は出力ファイルに保管されません。そうでない場合、データのみが保管されます。

```
db2 export to myfile.ixf of ixf messages msgs.txt select * from staff
```

次に示すのは、SAMPLE データベースの STAFF 表 (ユーザーが接続している必要がある) から部門 20 の従業員に関する情報を、awards.ixf に IXF 形式の出力でエクスポートする方法の一例です。

```
db2 export to awards.ixf of ixf messages msgs.txt select * from staff  
where dept = 20
```

次に示すのは、LOB を DEL ファイルにエクスポートする方法の一例です。

```
db2 export to myfile.del of del lobs to mylobs  
lobfile lobs1, lobs2 modified by lobsinfile  
select * from emp_photo
```

次の例は、LOB を DEL ファイルにエクスポートし、最初のディレクトリーに適合しないファイルに 2 番目のディレクトリーを指定する方法を示します。

```
db2 export to myfile.del of del  
lobs to /db2exp1, /db2exp2 modified by lobsinfile  
select * from emp_photo
```

次の例は、単一引用符をストリング区切り文字、セミコロンを列区切り文字、コンマを小数点として使用して、データを DEL ファイルにエクスポートする方法を示します。データをデータベースにインポートし直す場合にも同じ規則を使用するべきです。

```
db2 export to myfile.del of del  
modified by chardel'' coldel; decpt,  
select * from staff
```

使用上の注意

エクスポート操作を開始する前に、すべての表操作が完了し、すべてのロックが保留解除になっていることを確認してください。これは、WITH HOLD でオープンされた、すべてのカーソルをクローズした後で COMMIT または ROLLBACK を発行することによって行われます。

SELECT ステートメントでは表の別名を使用できます。

メッセージ・ファイルに置かれるメッセージには、メッセージ検索サービスから戻される情報が含まれています。各メッセージは改行してから始まります。

DEL 形式ファイルへエクスポートするために 254 よりも長い文字データの列が選択されると、エクスポート・ユーティリティーは警告メッセージを生成します。

PC/IXF インポートはデータベース間でデータを移動する場合に使用します。行区切り文字を含む文字データを区切り文字付き ASCII (DEL) ファイルにエクスポートし、テキスト転送プログラムにより処理を行うと (たとえば、OS/2 と AIX システム間で移動する)、行区切り文字を含むフィールドは長さが伸縮します。

PC/IXF ファイル形式の指定では、OS/2 (IBM OS/2 拡張サービス、OS/2 拡張エディション、および DB2 (OS/2 版)) のデータベースと、DB2 (AIX 版) のデータベースの間で、エクスポート (OS/2 と AIX の間でファイルをバイナリー・コピー) とインポートによってデータを移行することが可能です。ソースと宛先のデータベースが両方とも同じクライアントからアクセス可能である場合、ファイルのコピーというステップは必要ありません。

DB2 コネクトは、DB2 (OS/390 版)、DB2 (VM および VSE 版)、および DB2 (OS/400 版) などの DRDA サーバーから表をエクスポートするために使用できます。PC/IXF エクスポートだけがサポートされています。

エクスポート・ユーティリティーは、AIX システムから呼び出される場合、複数部分からなる PC/IXF ファイルを作成しません。

エクスポート・ユーティリティーは、提供される SELECT ステートメントが、SELECT * FROM tablename という形式である場合、IXF ファイルの表の NOT NULL WITH DEFAULT 属性を保管します。

タイプ表をエクスポートする場合、副選択ステートメントは、ターゲット表名と WHERE 文節を指定することによってのみ表現することができます。階層をエクスポートするとき、全選択と選択ステートメント は指定できません。

IXF 以外のファイル形式の場合は、走査順序リストを指定するようお勧めします。このリストは、階層を走査する方法やエクスポートする副表を DB2 に指示します。このリストを指定しない場合、階層内のすべての表がエクスポートされ、省略時の順序は OUTER 順序になります。または、OUTER 関数によって指定される順序である、省略時の順序を使用することができます。

注: インポート操作の間も同じ走査順序を使用します。ロード・ユーティリティーは、階層や副階層のロードをサポートしていません。

DB2 データ・リンク・マネージャーに関する考慮事項

表と、DATALINK 列が参照する対応ファイルの整合性の取れたコピーがエクスポートでコピーされるよう保証するには、以下のようにします。

1. コマンド `QUIESCE TABLESPACES FOR TABLE tablename SHARE` を発行します。
これにより、EXPORT の実行時に進行中の更新トランザクションがないことが保証されます。
2. EXPORT コマンドを発行します。
3. 各データ・リンク・サーバーで `dlfm_export` ユーティリティーを実行します。`dlfm_export` への入力は、エクスポート・ユーティリティーで生成される制御ファイル名です。これにより、制御ファイル内にリストされるファイルの tar (または同等の) アーカイブが作成されます。
4. コマンド `QUIESCE TABLESPACES FOR TABLE tablename RESET` を発行します。
これにより、表は更新に使用できるようになります。

EXPORT は SQL アプリケーションとして実行されます。SELECT ステートメントを満たす行と列がデータベースから抽出されます。DATALINK 列の場合、SELECT ステートメントはスカラー関数を指定できません。

EXPORT が正常に実行されると、以下のファイルが生成されます。

- EXPORT コマンドで指定したエクスポート・データ・ファイル。このファイルの DATALINK 列の値の形式は、434 ページ で説明されています。DATALINK 列の値が SQL NULL 値である場合、他のデータ・タイプと同じ処理が行われます。
- 制御ファイル `server_name`。このファイルは、データ・リンク・サーバーごとに生成されます (Windows NT オペレーティング・システムでは、単一の制御ファイル `ctrlfile.lst` をすべてのデータ・リンク・サーバーが使用します)。これらの制御ファイルは、ディレクトリー `<data-file path>%dlfm%YYYYMMDD%HHMMSS` に入れられます (Windows NT オペレーティング・システムの場合、`ctrlfile.lst` はディレクトリー `<data-file path>%dlfm%YYYYMMDD%HHMMSS` に入れられます)。YYYYMMDD は日付 (年月日)を、HHMMSS は時刻 (時、分、秒) を表します。

ファイルをデータ・リンク・サーバーからエクスポートするため、**dlfm_export** ユーティリティーが提供されています。このユーティリティーは、宛先データ・リンク・サーバーにファイルを復元するのに使用できる、アーカイブ・ファイルを生成します。

表 5. 有効なファイル・タイプ修飾子 (エクスポート)

修飾子	説明
すべてのファイル形式	
lobsinfile	<i>lob-path</i> には、LOB 値を含むファイルへのパスを指定します。
DEL (区切り付き ASCII) ファイル形式	
chardelx	<p><i>x</i> は単一文字ストリング区切り文字です。省略時値は二重引用符 (") です。指定した文字は二重引用符の代わりに、文字ストリングを囲むために使用されます。^a</p> <p>単一引用符 (') も次のように文字ストリング区切り文字として指定できます。</p> <p style="text-align: center;">modified by chardel''</p>
coldelx	<p><i>x</i> は単一文字列区切り文字です。省略時値はコンマ (,) です。指定した文字はコンマの代わりに、列の終わりを表すために使用されます。^a</p> <p>次の例では、coldel; によってエクスポート・ユーティリティーは、すべてのセミコロン (;) を列区切り文字として解釈します。</p> <p style="text-align: center;">db2 "export to temp of del modified by coldel; select * from staff where dept = 20"</p>
datesiso	日付形式。これによって、すべての日付データ値は ISO 形式 ("YYYY-MM-DD") でエクスポートされます。 ^b
decplusblank	正符号文字。これによって正の 10 進値の先頭に正符号 (+) ではなく、ブランク・スペースが置かれます。省略時のアクションでは、正の 10 進数の前に正符号 (+) が付けられます。
decptx	<i>x</i> は、小数点としてピリオドと置換される単一文字です。省略時値はピリオド (.) です。指定した文字はピリオドの代わりに、小数点文字として使用されます。 ^a

表 5. 有効なファイル・タイプ修飾子 (エクスポート) (続き)

修飾子	説明
dldelx	<i>x</i> は単一文字の DATALINK 区切り文字です。省略時値はセミコロン (;) です。指定した文字はセミコロンの代わりに、DATALINK 値のフィールド間区切り文字として使用されます。DATALINK 値には副値が複数個含まれる場合があるため、これが必要になります。 ^a 注: 行、列、または文字ストリング区切り文字と同じ文字を <i>x</i> に指定することはできません。
nodoubledel	二重文字区切り文字の認識を抑制します。詳細については、265ページの『区切り文字の制限』を参照してください。
WSF ファイル形式	
1	ロータス 1-2-3 リリース 1、または ロータス 1-2-3 リリース 1a と互換性のある WSF ファイルを作成します。 ^c これが省略時値です。
2	Lotus Symphony リリース 1.0 と互換性のある WSF ファイルを作成します。 ^c
3	ロータス 1-2-3 バージョン 2、または Lotus Symphony リリース 1.1 と互換性のある WSF ファイルを作成します。 ^c
4	DBCS 文字を含む WSF ファイルを作成します。
<p>注:</p> <ol style="list-style-type: none"> MODIFIED BY オプションでサポートされていないファイル・タイプが指定されても、エクスポート・ユーティリティーが警告を発することはありません。サポートされていないファイル・タイプを使おうとすると、エクスポート操作は失敗し、エラー・コードが戻されます。 ^a 265ページの『区切り文字の制限』に、区切り文字の指定変更として使用できる文字に適用される制限のリストが示されています。 ^b エクスポート・ユーティリティーは、通常は以下を書き出します。 <ul style="list-style-type: none"> YYYYMMDD 形式の日付データ "YYYY-MM-DD" 形式の char(date) データ "HH.MM.SS" 形式の時間データ "YYYY-MM-DD-HH.MM.SS.uuuuuu" 形式のタイム・スタンプ・データ SELECT ステートメントでエクスポート操作のために指定される日時列に含まれるデータも、これらの形式になります。 ^c filetype-mod パラメーター・ストリングの中で、ロータス 1-2-3 の場合は L、Symphony の場合は S を指定することにより、これらのファイルを特定の製品を指すようにできます。1 つの値または製品指定文字だけを指定できます。 	

区切り文字の制限

選択した区切り文字が移動されるデータの一部になっていないことを確認するのは、ユーザーの責任において行ってください。区切り文字がデータの一部になっている場合、予期しないエラーが発生する場合があります。データを移動する際は、以下の制限が列、ストリング、DATALINK、および小数点区切り文字に適用されます。

- 区切り文字は相互に排他的です。
- 2 進ゼロ、改行文字、改行、またはブランク・スペースを区切り文字にすることはできません。
- 省略時の小数点 (.) は、ストリング区切り文字にすることはできません。
- 次の文字は、ASCII ファミリー・コード・ページと EBCDIC ファミリー・コード・ページで、仕様が異なります。
 - シフトイン (0x0F) およびシフトアウト (0x0E) 文字は、EBCDIC MBCS データ・ファイルの場合に、区切り文字にすることができません。
 - MBCS、EUC、または DBCS コード・ページの区切り文字は、0x40 より大きくすることはできません。ただし、EBCDIC MBCS データの省略時の小数点 0x4b は例外です。
 - ASCII コード・ページまたは EBCDIC MBCS コード・ページにおけるデータ・ファイルの省略時区切り文字は、以下のとおりです。
 - " (0x22、二重引用符; ストリング区切り文字)
 - , (0x2c、コンマ; 列区切り文字)
 - EBCDIC SBCS コード・ページにおけるデータ・ファイルの省略時区切り文字は、以下のとおりです。
 - " (0x7F、二重引用符; ストリング区切り文字)
 - , (0x6B、コンマ; 列区切り文字)
 - ASCII データ・ファイルの省略時の小数点は 0x2e (ピリオド) です。
 - EBCDIC データ・ファイルの省略時の小数点は 0x4B (ピリオド) です。
 - サーバーのコード・ページがクライアントのコード・ページと異なっている場合は、非省略時の区切り文字を 16 進表示で指定するようお勧めします。たとえば、次のように指定します。

```
db2 load from ... modified by charde10x0C colde1X1e ...
```

DEL ファイルでの二重文字区切り文字の認識サポートに関する以下の情報は、エクスポート、インポート、およびロード・ユーティリティーに適用されません。

- 文字区切り文字を、DEL ファイルの文字ベースのフィールド内で使用することができます。これは、タイプが CHAR、VARCHAR、LONG VARCHAR、

EXPORT

または CLOB (lobsinfile が指定されている場合を除く) のフィールドに適用されます。文字区切り文字で囲まれている文字区切り文字の対は、データベースにインポートまたはロードされます。たとえば、次のように指定します。

```
"What a "nice" day!"
```

これは、次のようにインポートされます。

```
What a "nice" day!
```

エクスポートの場合は、逆の規則が適用されます。たとえば、次のように指定します。

```
I am 6" tall.
```

これは、次のように DEL ファイルにエクスポートされます。

```
"I am 6"" tall."
```

- DBCS 環境では、パイプ (|) 桁区切り文字はサポートされません。

以下も参照

332ページの『IMPORT』

411ページの『LOAD』

FORCE APPLICATION

サーバー上で保守を行えるようにするため、ローカルまたはリモートのユーザーやアプリケーションをシステムから強制終了します。

考慮事項: 割り込みできない操作 (たとえば、RESTORE DATABASE) を強制終了する場合、データベースが利用可能になるには、その操作の再実行が正常終了しなければなりません。

効力範囲

このコマンドは、\$HOME/sql11lib/db2nodes.cfg ファイルにリストされているすべてのノードに影響を与えます。

区分データベース環境では、このコマンドを発行するのは、強制終了されているアプリケーションの調整プログラム・ノードからでなくてもかまいません。区分データベース環境内の任意のノード (データベース区画サーバー) から発行することができます。

許可

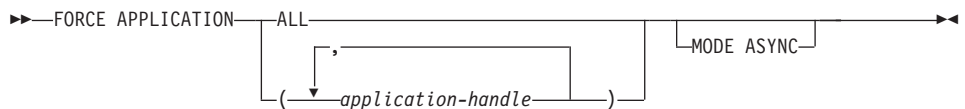
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

インスタンス。 リモート・サーバーからユーザーを強制終了する場合、最初にそのサーバーに接続する必要があります。接続が存在しない場合、このコマンドはローカルで実行されます。

コマンド構文



コマンド・パラメーター

APPLICATION

ALL すべてのアプリケーションがデータベースから切断されます。

FORCE APPLICATION

application-handle

エージェントの終了を指定します。

364ページの『LIST APPLICATIONS』を使用して値をリストします。

MODE ASYNC

コマンドは、指定したすべてのユーザーが終了するのを待たずに戻ってきます。コマンドは、機能を正常に発行するか、またはエラー（無効な構文などの）を発見するとすぐに戻ります。

現在サポートしているモードはこのモードだけです。

例

次の例は、*application-handle* の値が 41408 と 55458 の 2 つのユーザーをデータベースから強制的に切断します。

```
db2 force application ( 41408, 55458 )
```

使用上の注意

db2stop は強制終了の間は実行できません。データベース・マネージャーは、**db2start** を必要とせずに、後続のデータベース・マネージャー操作を処理できるようにするため、活動状態のままになっています。

データベースの保全性を確保するため、終了できるのは、アイドル中のユーザー、または割り込み可能なデータベース操作を実行中のユーザーだけです。

データベースを作成しているユーザーは強制終了できません。

FORCE が出された後も、データベースはまだ接続要求を受諾します。すべてのユーザーを完全に強制終了するためには、追加の FORCE が必要になる場合があります。

以下も参照

154ページの『ATTACH』

364ページの『LIST APPLICATIONS』

GET ADMIN CONFIGURATION

DB2 管理サーバーと関係のあるデータベース・マネージャー構成ファイル内の、個々の項目の値を戻します。DB2 管理サーバーは、DB2 サーバーのリモート管理を使用可能にする特別な DB2 インスタンスです。以下のデータベース・マネージャー構成パラメーターが表示されます。

- AGENT_STACK_SZ
- AUTHENTICATION
- DIAGLEVEL
- DIAGPATH
- DISCOVER
- DISCOVER_COMM
- FILESERVER
- IPX_SOCKET
- NNAME
- OBJECTNAME
- QUERY_HEAP_SZ
- SVCENAME
- SYSADM_GROUP
- SYSCTRL_GROUP
- SYSMANT_GROUP
- TPNAME
- TRUST_ALLCLNTS
- TRUST_CLNTAUTH

注: インストール・プログラムによって設定される SVCENAME パラメーターは、ユーザーが修正することはできません。管理サーバー・サービス名は、TCP/IP ポート (523) に登録済みの DB2 を使用するよう設定します。

上記のパラメーターの詳細については、292ページの『GET DATABASE MANAGER CONFIGURATION』を参照してください。

GET ADMIN CONFIGURATION

効力範囲

このコマンドは、同じ \$HOME/sql1lib ディレクトリーを共用するすべてのノードの情報を戻します。このコマンドは、それらの任意のノードから発行できます。

許可

なし

必須接続

なし、またはインスタンス。インスタンスとの接続は、ローカルの DBM 構成操作を実行する場合には必ずしも必要ではありませんが、リモートの DBM 構成操作の場合には必須です。リモート・インスタンスのデータベース・マネージャ構成を表示するには、まず最初にそのインスタンスと接続することが必要です。

コマンド構文



コマンド・パラメーター

なし

例

以下に示すのは、GET ADMIN CONFIGURATION の出力例です。


```

Admin Server Configuration
Node type = Database Server with local and remote clients
Database manager configuration release level      = 0x0900
Diagnostic error capture level                   (DIAGLEVEL) = 4
Diagnostic data directory path                   (DIAGPATH)  =
SYSADM group name                               (SYSADM_GROUP) =
SYSCTRL group name                              (SYSCTRL_GROUP) =
SYSMAINT group name                             (SYSMAINT_GROUP) =

Database manager authentication                 (AUTHENTICATION) = SERVER
Trust all clients                              (TRUST_ALLCLNTS) = YES
Trusted client authentication                  (TRUST_CLNTAUTH) = CLIENT
Query heap size (4KB)                         (QUERY_HEAP_SZ) = 250
TCP/IP Service name                            (SVCENAME)     = 30676
APPC Transaction program name                  (TPNAME)       =
IPX/SPX File server name                       (FILESERVER)   =
IPX/SPX DB2 server object name                 (OBJECTNAME)   =
IPX/SPX Socket number                          (IPX_SOCKET)   = 87A2

Discovery mode                                 (DISCOVER)     = SEARCH
Discovery communication protocols              (DISCOVER_COMM) = TCPIP

```

使用上の注意

リモート・インスタンス（または別のローカル・インスタンス）への接続が存在する場合、それらのインスタンスに接続されたサーバーの ADMIN 構成パラメーターが戻されます。そのようなインスタンスが存在しない場合には、ローカルの ADMIN 構成パラメーターが戻されます。

エラーが生じた場合には、戻された情報は無効になります。構成ファイルが無効な場合には、エラー・メッセージが戻されます。そのような場合には、データベース・マネージャーを再インストールして回復を実行する必要があります。

データベース・マネージャー出荷時の省略時値に構成パラメーターを設定するには、528ページの『RESET ADMIN CONFIGURATION』を使用してください。

上記のパラメーターの詳細については、[管理の手引き](#) を参照してください。

以下も参照

528ページの『RESET ADMIN CONFIGURATION』

599ページの『UPDATE ADMIN CONFIGURATION』

GET AUTHORIZATIONS

GET AUTHORIZATIONS

データベース構成ファイルおよび許可システム・カタログ視点
(SYSCAT.DBAUTH) 内で検出した値から現行ユーザーの権限を報告します。

許可

なし

必須接続

データベース。暗黙接続が可能な場合には、省略時データベースへの接続が確立されます。

コマンド構文

▶—GET AUTHORIZATIONS—▶

コマンド・パラメーター

なし

例

以下に示すのは、GET AUTHORIZATIONS の出力例です。

```
Administrative Authorizations for Current User
Direct SYSADM authority           = NO
Direct SYSCtrl authority          = NO
Direct SYSMAINT authority         = NO
Direct DBADM authority            = YES
Direct CREATETAB authority        = YES
Direct BINDADD authority          = YES
Direct CONNECT authority          = YES
Direct CREATE_NOT_FENC authority  = YES
Direct IMPLICIT_SCHEMA authority  = YES
Direct LOAD authority             = YES
Indirect SYSADM authority         = YES
Indirect SYSCtrl authority        = NO
Indirect SYSMAINT authority       = NO
Indirect DBADM authority          = NO
Indirect CREATETAB authority      = YES
Indirect BINDADD authority        = YES
Indirect CONNECT authority        = YES
Indirect CREATE_NOT_FENC authority = NO
Indirect IMPLICIT_SCHEMA authority = YES
Indirect LOAD authority           = NO
```

使用上の注意

直接権限は、ユーザー ID に対する権限を与える明示コマンドによって獲得されます。それに対し、間接権限とは、ユーザーが所属するグループによって獲得された権限を基盤としている権限のことをいいます。

注: PUBLIC は、全ユーザーが所属することになる特殊なグループです。

GET CLI CONFIGURATION

GET CLI CONFIGURATION

db2cli.ini ファイルの内容をリスト表示します。ファイル全体または指定したセクションをリスト表示することができます。

db2cli.ini ファイルは、DB2 コール・レベル・インターフェース (CLI) 構成ファイルとして使用されます。このファイルには、DB2 CLI およびそれを使用するアプリケーションの動作を変更するために使用できるさまざまなキーワードと値が含まれます。このファイルは複数のセクションに分かれており、それぞれのセクションはデータベース別名に対応します。このファイルと CLI/ODBC 構成キーワードの詳細については、[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。

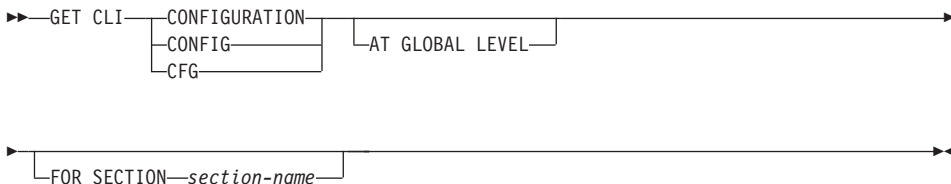
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

AT GLOBAL LEVEL

LDAP ディレクトリーの省略時値 CLI 構成パラメーターを表示します。

注: このパラメーターは Windows オペレーティング・システム上だけで有効です。

FOR SECTION section-name

キーワードがリスト表示されるセクションの名前。指定しない場合、すべてのセクションがリスト表示されます。

例

以下の出力例は、2 つのセクションがある db2cli.ini ファイルの内容を表しています。

```
[tstcli1x]
uid=userid
pwd=password
autocommit=0
TableType="'TABLE','VIEW','SYSTEM TABLE'"
[tstcli2x]
SchemaList="'OWNER1','OWNER2','CURRENT SQLID'"
```

使用上の注意

このコマンドで指定するセクション名では、大文字小文字の区別がありません。たとえば、db2cli.ini ファイルのセクション名 (大括弧で区切られる) が小文字であり、コマンドで指定したセクション名が大文字であっても、正しいセクションがリスト表示されます。

PWD (パスワード) キーワードの値がリスト表示されることはありません。代わりに、5 つのアスタリスク (*****) がリスト表示されます。

LDAP (Lightweight Directory Access Protocol) が使用可能な場合、CLI 構成パラメーターを、マシン・レベルに加えてユーザー・レベルでも設定できます。ユーザー・レベルの CLI 構成は、LDAP ディレクトリーに保持されます。指定されたセクションがユーザー・レベルで存在する場合、ユーザー・レベルでそのセクションの CLI 構成が戻されます。そうでない場合、マシン・レベルの CLI 構成が戻されます。

ユーザー・レベルの CLI 構成は、LDAP ディレクトリーに保持され、ローカル・マシンでキャッシュされます。CLI 構成をユーザー・レベルで読み取る場合、DB2 は常にキャッシュから読み取ります。キャッシュは、次のときに更新されます。

- ユーザーが CLI 構成を更新するとき。
- ユーザーが REFRESH LDAP コマンドを使用して、明示的に CLI 構成の最新表示を強制するとき。

LDAP 環境では、ユーザーは LDAP ディレクトリーにカタログ化されたデータベースに対して、省略時 CLI 設定値のセットを構成することができます。LDAP カタログ化データベースが、データ・ソース名 (DSN) として、クライアント構成アシスタント (CCA) または ODBC 構成ユーティリティーのどちら

GET CLI CONFIGURATION

かを使用して追加されると、省略時の CLI 設定が LDAP ディレクトリーにある場合には、それらはローカル・マシン上のその DSN 用に構成されます。省略時の CLI 設定を表示するには、AT GLOBAL LEVEL 文節を指定する必要があります。

以下も参照

509ページの『REFRESH LDAP』

602ページの『UPDATE CLI CONFIGURATION』

GET CONNECTION STATE

接続状態を表示します。以下の状態のいずれかが表示されます。

- 接続可能かつ接続済み
- 接続可能で未接続
- 接続不可能で接続済み
- 暗黙接続可能 (暗黙接続が利用可能な場合に限る)

上記の接続状態の詳細については、*SQL 解説書* を参照してください。

このコマンドを実行することにより、接続状態に関する情報だけでなく、データベースの接続モード (SHARE もしくは EXCLUSIVE) もわかりますし、接続されているデータベースがあればその別名および名前を知ることができます。

許可

なし

必須接続

なし

コマンド構文

▶—GET CONNECTION STATE—▶

コマンド・パラメーター

なし

例

以下に示すのは、GET CONNECTION STATE の出力例です。

```
Database Connection State
Connection state      = Connectable and Connected
Connection mode      = SHARE
Local database alias = SAMPLE
Database name        = SAMPLE
```

GET CONNECTION STATE

使用上の注意

このコマンドは、タイプ 2 接続には適用されません (566ページの『SET CLIENT』を参照)。

GET DATABASE CONFIGURATION

特定のデータベース構成ファイル内にある個々の項目の値を返します。

効力範囲

このコマンドは、それが実行されたノードに対してだけ情報を返します。

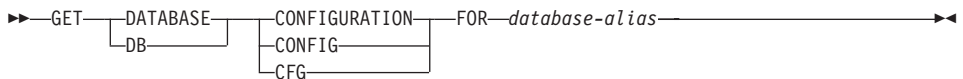
許可

なし

必須接続

インスタンス。明示的な接続は必要ありません。データベースがリモートとして示されている場合、リモート・ノードへのインスタンス接続はコマンドの持続期間の間、ずっと確立されたままになります。

コマンド構文



コマンド・パラメーター

FOR database-alias

構成を表示したいデータベースの別名を指定します。

例

注:

1. プラットフォームが異なると、プラットフォーム固有のパラメーターを反映して、出力の結果に微妙な違いが生じる場合があります。
2. キーワードが括弧で囲まれているパラメーターは、606ページの『UPDATE DATABASE CONFIGURATION』を使用して変更できます。
3. キーワードを持たないフィールドは、データベース・マネージャーが保守します。そのようなフィールドを更新することはできません。

以下に示すのは、GET DATABASE CONFIGURATION の出力例です (AIX で発行)。

```

  Database Configuration for Database sample
  Database configuration release level      = 0x0900
  Database release level                   = 0x0900
  Database territory                       = US
  
```

GET DATABASE CONFIGURATION

```
Database code page = 819
Database code set = IS08859-1
Database country code = 1
Dynamic SQL Query management (DYN_QUERY_MGMT) = DISABLE
Directory object name (DIR_OBJ_NAME) =
Discovery support for this database (DISCOVER_DB) = ENABLE
Default query optimization class (DFT_QUERYOPT) = 5
Degree of parallelism (DFT_DEGREE) = 1
Continue upon arithmetic exceptions (DFT_SQLMATHWARN) = NO
Default refresh age (DFT_REFRESH_AGE) = 00000000000000.000000
Number of frequent values retained (NUM_FREQVALUES) = 10
Number of quantiles retained (NUM_QUANTILES) = 20
Backup pending = NO
Database is consistent = YES
Rollforward pending = NO
Restore pending = NO
Multi-page file allocation enabled = NO
Log retain for recovery status = NO
User exit for logging status = NO
Data Links Token Expiry Interval (sec) (DL_EXPINT) = 60
Data Links Number of Copies (DL_NUM_COPIES) = 1
Data Links Time after Drop (days) (DL_TIME_DROP) = 1
Data Links Token in Uppercase (DL_UPPER) = NO
Data Links Token Algorithm (DL_TOKEN) = MACO
Database heap (4KB) (DBHEAP) = 1200
Catalog cache size (4KB) (CATALOGCACHE_SZ) = 64
Log buffer size (4KB) (LOGBUFSZ) = 8
Utilities heap size (4KB) (UTIL_HEAP_SZ) = 5000
Buffer pool size (pages) (BUFFPAGE) = 1000
Extended storage segments size (4KB) (ESTORE_SEG_SZ) = 16000
Number of extended storage segments (NUM_ESTORE_SEGS) = 0
Max storage for lock list (4KB) (LOCKLIST) = 100
Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ) = 128
Sort list heap (4KB) (SORTHEAP) = 256
SQL statement heap (4KB) (STMTHEAP) = 2048
Default application heap (4KB) (APPLHEAPSZ) = 128
Package cache size (4KB) (PCKCACHESZ) = (MAXAPPLS*8)
Statistics heap size (4KB) (STAT_HEAP_SZ) = 4384
Interval for checking deadlock (ms) (DLCHKTIME) = 10000
Percent. of lock lists per application (MAXLOCKS) = 10
Lock timeout (sec) (LOCKTIMEOUT) = -1
Changed pages threshold (CHNGPGS_THRESH) = 60
Number of asynchronous page cleaners (NUM_IOCLEANERS) = 1
Number of I/O servers (NUM_IOSERVERS) = 3
Index sort flag (INDEXSORT) = YES
Sequential detect flag (SEQDETECT) = YES
Default prefetch size (pages) (DFT_PREFETCH_SZ) = 32
Default number of containers = 1
Default tablespace extentsize (pages) (DFT_EXTENT_SZ) = 32
Max number of active applications (MAXAPPLS) = 40
Average number of active applications (AVG_APPLS) = 1
Max DB files open per application (MAXFILOP) = 64
Log file size (4KB) (LOGFILLSIZ) = 1000
Number of primary log files (LOGPRIMARY) = 3
Number of secondary log files (LOGSECOND) = 2
```

GET DATABASE CONFIGURATION

```
Changed path to log files          (NEWLOGPATH) =
Path to log files                  = /home/jessicae/
                                   jessicae/NODE0000/
                                   SQL00001/SQLLOGDIR/
First active log file              =
Group commit count                 (MINCOMMIT) = 1
Percent log file reclaimed before soft ckcpt (SOFTMAX) = 100
Log retain for recovery enabled    (LOGRETAIN) = OFF
User exit for logging enabled      (USEREXIT) = OFF
Auto restart enabled               (AUTORESTART) = ON
Index re-creation time             (INDEXREC) = SYSTEM (RESTART)
Default number of loadrec sessions (DFT_LOADREC_SES) = 1
Number of database backups to retain (NUM_DB_BACKUPS) = 12
Recovery history retention (days) (REC_HIS_RETENTN) = 366
TSM management class              (TSM_MGMTCLASS) =
TSM node name                      (TSM_NODENAME) =
TSM owner                          (TSM_OWNER) =
TSM password                       (TSM_PASSWORD) =
```

上記のフィールドを以下に説明します。小文字で表示された名前を持つパラメーターは、データベース・マネージャーが保守しており、更新することはできません。データベース構成パラメーターの詳細については、[管理の手引き](#) を参照してください。

APP_CTL_HEAP_SZ

このパラメーターは、アプリケーション制御ヒープの最大サイズ (4KB ページ単位) を判別します。ヒープは、MPP または SMP システム内の 1 つのノードで、同一のアプリケーションの代理として作動しているエージェントの間の情報を共有することを必要とします。複合のアプリケーションを実行している場合、または MPP 構成に大量のノードがある場合、このヒープ・サイズを大きくしてください。

APPLHEAPSZ

個々のエージェントで利用可能なアプリケーション・ヒープのサイズ (ページ単位) を示します。

AUDIT_BUF_SZ

データベースの監査に使用するバッファのサイズを 4KB 単位で指定します。値が非ゼロである場合、監査機能は、監査レコードを生成するステートメントの実行について、レコードを非同期にディスクに書き込みます。

AUTORESTART

たとえば、データベースの接続が異常切断した場合、または直前のセッションでデータベースが正常に終了しなかった場合などに、データベース・マネージャーが `RESTART DATABASE` を接続に自動的に発行できるかどうかを示します。

GET DATABASE CONFIGURATION

OFF は、コマンドの発行を手動で行う必要のあることを示します。

ON は、データベース・マネージャーが自動的にコマンドを発行してくれることを示します。

AVG_APPLS

活動アプリケーションの平均数。このパラメーターは、SQL 最適化プログラムによって使用されます。実行時にどのくらいのバッファー・プールを、選択されたアクセス・プランのために利用できるかを見定めるのに役立ちます。

backup_pending (バックアップ保留)

NO は、データベースが使用できる状態にあることを示します。

YES は、データベースを使用できる状態にするには、オフライン・バックアップを実行しなければならないことを示します。

BUFFPAGE

バッファー・プールのサイズ (ページ単位) を示します。バッファー・プールは、データベースから読み取られたデータを保管および操作するために使用します。このパラメーターは、CREATE BUFFERPOOL、ALTER BUFFERPOOL、または 455 ページの『MIGRATE DATABASE』を介して、バッファー・プールのサイズが明示的に -1 に設定されている場合にのみ使用します。バッファー・プールのサイズは、通常 SQL ステートメントを介して制御されます。

CATALOGCACHE_SZ

(*dbheap* から割り振られた) 内部カタログ・キャッシュのサイズ (ページ単位) を制御します。SQL コンパイラーによって使用され、通常、参照オブジェクト (表または制約など) のパック記述子を保留します。

CHNGPGS_THRESH

変更されたページのしきい値。非同期ページ・クリーナーが現在非活動状態の場合、それらが開始される際には、変更されたページのレベル (パーセンテージ) がどの程度になるかを示します。

codepage (データベースのコード・ページ)

データベースのコード・ページを示します。

codeset (データベースのコード・セット)

データベースのコード・セットを示します。

COPYPROTECT (OS/2 のみ)

コピー保護属性が使用可能です。

country (データベースの国別コード)

データベースの国別コードを示します。

database_consistent (データベースの一貫性)

NO は、トランザクション、または他の一部のタスクがデータベースで保留中であることを示します。またこの時点でデータに一貫性がないことも示しています。

YES は、すべてのトランザクションがコミットまたはロールバックされ、それゆえデータに一貫性があることを示します。

database_level (データベースのリリース・レベル)

データベースのリリース・レベル。データベースを使用するデータベース・マネージャーのリリース・レベルを示します。

DBHEAP

オープン・カーソルのうち、そのデータベースにアクセスするものすべてに関する制御情報を保留するのに使われる、データベース・ヒープのサイズ (ページ単位) を示します。 *logbufsz* と *catalogcache_sz* の両方が、 *dbheap* から割り振られます。

DFT_DEGREE

このパラメーターは、 CURRENT DEGREE 特殊レジスターおよび DEGREE バインド・オプションに省略時値を指定します。

DFT_EXTENT_SZ

表スペースの省略時エクステント・サイズ (ページ単位)。

DFT_LOADREC_SES

ロード回復セッションの省略時の数。テーブル・ロードの回復中に使われることになるセッションの省略時数を示します。このパラメーターは、ロールフォワード回復が可能な場合にかぎり、採用できます。

DFT_PREFETCH_SZ

表スペースの省略時事前取り出しサイズ (ページ単位)。

DFT_QUERYOPT

照会最適化クラスは、SQL 照会をコンパイルする際に、最適化プログラムが使用する最適化の異なる程度を指示するのに使用します。このパラメーターは、SET CURRENT QUERY OPTIMIZATION ステートメントも QUERYOPT バインド・コマンドも使用しない場合に、省略時照会最適化クラスの設定によって、付加的な柔軟性を提供します。

DFT_REFRESH_AGE

CURRENT REFRESH AGE 特殊レジスターが指定されていない場合に、REFRESH AGE に使用される省略時値。このパラメーターは、要約表が動的 SQL 照会の処理を最適化すると見なされるかどうかを判断するために使用します。

GET DATABASE CONFIGURATION

DFT_SQLMATHWARN

算術計算エラーおよび検索変換エラーを、SQL ステートメントのコンパイル時にエラーまたは警告として扱うかどうかを決める省略時値 (YES または NO) を設定します。

DIR_OBJ_NAME

DCE 名前空間でのオブジェクト名。この場合のオブジェクト名は、ディレクトリー内のデータベース・マネージャー・インスタンス (またはデータベース) を表します。この値と *dir_path_name* 値を連結すると、*dir_type* パラメーターで指定されたディレクトリー・サービスが管理する名前空間にある、データベース・マネージャー・インスタンスまたはデータベースを固有に識別できる大域名になります。

DISCOVER_DB

このパラメーターを `DISABLE` に設定することにより、サーバーに対してクライアントがディスクバリー要求を出す場合に、データベースについての情報をクライアントに戻さないようにできます。

DL_EXPINT

DB2 データ・リンク・マネージャーにのみ適用されます。このパラメーターは、生成されたファイル・アクセス・トークンが有効である時間間隔 (秒単位) を指定します。トークンが有効である秒数は、生成時から始まります。データ・リンク・ファイルシステム・フィルターは、この有効時間を含むトークンの妥当性を検査します。

DL_NUM_COPIES

DB2 データ・リンク・マネージャーにのみ適用されます。このパラメーターは、ファイルがデータベースにリンクされるときにアーカイブ・サーバー (TSM サーバーなど) で作成される、ファイルの追加コピーの数を指定します。

DL_TIME_DROP

DB2 データ・リンク・マネージャーにのみ適用されます。このパラメーターは、`DROP TABLE`、`DROP DATABASE`、または `DROP TABLESPACE` の発行後にアーカイブ・サーバー (TSM サーバーなど) でファイルが保存される時間間隔 (日単位) を指定します。

DL_TOKEN

DB2 データ・リンク・マネージャーにのみ適用されます。このパラメーターは、`DATALINK` ファイル・アクセス制御トークンで使用されるアルゴリズムを指定します。

DL_UPPER

DB2 データ・リンク・マネージャーにのみ適用されます。このパラメ

ーターは、ファイル・アクセス制御トークンが大文字のみを使用するか、またはトークンに大文字小文字の両方を含めることができるかを示します。

DLCHKTIME

デッドロックを調べる時間間隔 (ミリ秒単位)。あるデータベースに接続されたアプリケーションすべてに見られるデッドロックを、データベース・マネージャーが調べる頻度を定義します。

DYN_QUERY_MGMT

このパラメーターは、DB2 クエリー・パトローラーのインストール先に関係します。このパラメーターが **ENABLE** に設定され、動的照会のコストがユーザーまたはグループのしきい値 (クエリー・パトローラー・ユーザー・プロファイル表で指定) を超える場合、この照会は DB2 クエリー・パトローラーによってトラップされます。トラップ・コストのしきい値は、クエリー・パトローラーでユーザーが設定する、照会トラップのトリガーになります。動的照会がトラップされると、ダイアログが表示されて、ランタイム・パラメーターを指定するように指示されます。このパラメーターを **DISABLE** に設定すると、照会はトラップされません。

ESTORE_SEG_SZ

このパラメーターによって、データベースの拡張メモリー・セグメントごとのページ数が示されます。この構成パラメーターを設定することは、プラットフォーム依存の考慮事項です。

INDEXREC

無効な索引が再作成される条件を示します。省略時の設定値は **SYSTEM** で、データベース・マネージャー構成パラメーター *indexrec* の値が使用されます。

以下の出力値を使用することもできます。

- **SYSTEM(ACCESS)**
- **SYSTEM(RESTART)**
- **ACCESS**
- **RESTART**

INDEXSORT

索引の分類フラグです。索引を作成する際に、索引キーを分類するかどうかを示します。

LOCKLIST

ロック・リストに割り振られた最大記憶域 (ページ単位) を示します。

GET DATABASE CONFIGURATION

LOCKTIMEOUT

アプリケーションがロックの取得に要する秒数を示します。

LOGBUFSZ

ディスクに書き込む前に、ログ・レコードをバッファーするのに使われるページ数を示します。 *dbheap* から割り振られます。

LOGFILSIZ

データ回復用のログ・ファイルに割り振られた、ディスク記憶域のサイズ (ページ単位) を示します。このパラメーターは、各 1 次および 2 次ログ・ファイルのサイズを定義します。

loghead (最初のアクティブ・ログ・ファイル)

ログのヘッドを識別するためのパラメーター。アクティブ・ログのヘッドを含むログ・ファイルの名前を示します。次に書き込まれるログ・レコードは、アクティブ・ログのヘッドから始まります。

logpath (ログ・ファイルへのパス)

ログ・ファイルのロケーション。このパラメーターには、ロギング目的で使われる現行パスが含まれます。

LOGPRIMARY

データベースの回復に用いることのできる 1 次ログ・ファイルの数を示します。

LOGRETAIN

アクティブ・ログ・ファイルがロールフォワード回復による使用のために保存されるか、またはデータ複製キャプチャー・プログラムによる使用のために保存されるかを示します。

log_retain_status (状況保存のためのログ保存)

ロールフォワード回復のために、ログ・ファイルが保存されるかどうかを示します。

LOGSECOND

データベースの回復に用いることのできる 2 次ログ・ファイル数を示します。

MAXAPPLS

1 度にデータベースに接続することのできるアプリケーション・プログラム (ローカルとリモートの両方) の最大数を示します。

MAXFILOP

アプリケーション・プログラムが 1 度にオープンすることのできるデータベース・ファイルの最大数を示します。

MAXLOCKS

ある特定のアプリケーション・プログラムが使用できるロック・リストの最大パーセンテージを示します。

MINCOMMIT

指定されたデータベースにグループ分けできる SQL コミットの数を示します。SQL コミットのグループ化により、コミット実行時の入出力およびログ活動の制御が容易になります。

multipage_alloc

複数ページ・ファイル割り振りは、挿入パフォーマンスを向上する目的で使用します。これは、SMS 表スペースにのみ適用されます。使用可能な場合、すべての SMS 表スペースが影響を受けます。個々の SMS 表スペースに選択の余地はありません。

NEWLOGPATH

あるデータベースの回復ログ・ファイルへの代替パスを示します。

newlogpath ディレクトリは完全に修飾されたディレクトリだけを受諾するので、このパラメーターには絶対パスを指定する必要があります。

NUM_DB_BACKUPS

このパラメーターによって、データベースに保存されるデータベース・バックアップの数が指定されます。バックアップの指定数に達すると、古いバックアップは回復履歴ファイルで有効期限が切れたとしてマークされます。バックアップが有効期限が切れたとしてマークされると、物理バックアップが保管場所から除去されます。次のデータベース・バックアップで、有効期限が切れたエントリが履歴ファイルから削除されます。

NUM_ESTORE_SEGS

このパラメーターによって、データベースが使用可能な拡張記憶域メモリー・セグメントの数が示されます。

NUM_FREQVALUES

保存された頻度値の数。WITH DISTRIBUTION オプションを 561 ページの『RUNSTATS』で指定した場合に、収集する「最頻度値」の数を指定するのに使用されます。

NUM_IOCLEANERS

ある特定のデータベースの非同期ページ・クリーナーの数を示します。

NUM_IOSERVERS

ある特定のデータベースの入出力サーバーの数を示します。入出力サー

GET DATABASE CONFIGURATION

バーは、データベース・エージェントに代わって、バックアップまたは復元などのユーティリティによる事前取り出し入出力や非同期入出力を実行するのに使われます。

NUM_QUANTILES

列の変位値の数。 WITH DISTRIBUTION オプションを 561ページの『RUNSTATS』で指定した場合に、収集する変位数の数を制御します。

numsegs (省略時のコンテナ数)

省略時の SMS 表スペース内で作成されるコンテナ数を判別します。

PCKCACHESZ

パッケージのキャッシュおよび動的 SQL ステートメントに応じて使用される記憶容量を指定します。

以下の場合、279ページの『GET DATABASE CONFIGURATION』の出力にラベル (calculated) が表示されます。

- 内部値が -1 である
- MAXAPPLS*8 が 32 未満である。この場合、32 がラベル (calculated) とともに表示されます。

REC_HIS_RETENTN

回復活動記録の保存期間。このパラメーターは、バックアップされた活動記録情報が保存される日数を示すのに使われます。

release (データベース構成のリリース・レベル)

データベース構成ファイルのリリース・レベルを示します。

restore_pending (復元の保留)

このパラメーターは、データベース中に RESTORE PENDING 状態が存在するかどうかを示します。

rollfwd_pending (ロールフォワード保留)

ロールフォワード回復手順がデータベースに必要かどうかを示します。

以下の値を指定することができます。

NO データベースも表スペースもロールフォワード保留状態にはないことを示します。

DATABASE

データベースをまず最初にロールフォワードする必要があることを示します。

TABLESPACES

データベース内の 1 つまたは複数の表スペースが、ロールフォワード回復を必要としていることを示します。

SEQDETECT

データベースの順次検出が使用可能になるかそれとも使用不能になるかを示します。

SOFTMAX

このパラメーターは、ソフト・チェックポイントで取得される頻度を指定し、破壊後に回復されるログの数を指定するために使用されます。

SORTHEAP

アプリケーション・プログラムの分類別に、利用可能な私用メモリー・ページの数を示します。

STAT_HEAP_SZ

統計ヒープ・サイズ (ページ単位)。配布統計の収集時に、すべての表統計の作成および収集に使われるヒープの最大サイズを示します。

STMTHEAP

SQL ステートメントのコンパイルに使用可能なヒープ・サイズ (ページ単位) を示します。

territory (データベース領域)

データベースのテリトリーを示します。

TSM_MGMTCLASS

TSM 管理クラスは、サーバーがバックアップ・バージョンを管理する方法、またはバックアップされたオブジェクトの複写の保存方法を指定します。管理クラスは、TSM 管理者から割り当てられます。一度割り当てられると、このパラメーターは管理クラス名に設定されます。任意の TSM バックアップの実行時、データベース・マネージャーはこのパラメーターを使用して、管理クラスを TSM に渡します。

TSM_NODENAME

このパラメーターは、省略時の設定を TSM 製品に関連付けられたノード名に一時変更するのに使用します。他のノードから TSM にバックアップがとられたデータベースを復元する場合には、ノード名が必要になります。

TSM_OWNER

このパラメーターは、省略時の設定を TSM 製品に関連付けられた所

GET DATABASE CONFIGURATION

有者に一時変更するのに使用します。他のノードから TSM にバックアップがとられたデータベースを復元する場合には、所有者名が必要になります。

TSM_PASSWORD

このパラメーターは、省略時の設定を TSM 製品に関連付けられたパスワードに一時変更するのに使用します。他のノードから TSM にバックアップがとられたデータベースを復元する場合には、パスワードが必要になります。

USEREXIT

データベースが次にオープンする際に、ログ・ファイルを保存または検索するためのユーザー出口機能呼び出すことができるかどうかを示します。

OFF は、ユーザー出口機能呼び出すことができないことを示します。

ON は、ユーザー出口機能呼び出すことができることを示します。

user_exit_status (状況をロギングするためのユーザー出口)

OFF は、ユーザー出口機能呼び出してアーカイブ・ログ・ファイルを保管させることはできないことを示します。

ON は、ユーザー出口機能呼び出してアーカイブ・ログ・ファイルを保管できることを示します。

UTIL_HEAP_SZ

ユーティリティ・ヒープ・サイズ。バックアップ、復元、およびロード・ユーティリティによって同時に使用できる共用メモリの最大サイズを示します。

使用上の注意

エラーが生じた場合には、戻された情報は無効になります。構成ファイルが無効な場合には、エラー・メッセージが戻されます。その場合には、データベースをバックアップ版から復元しなければなりません。

データベース・マネージャー構成パラメーターをデータベース・マネージャーの省略時値に設定するには、531ページの『RESET DATABASE CONFIGURATION』を使用してください。

DB2 構成パラメーターの詳細については、[管理の手引き](#)を参照してください。

以下も参照

531ページの『RESET DATABASE CONFIGURATION』

606ページの『UPDATE DATABASE CONFIGURATION』

GET DATABASE MANAGER CONFIGURATION

GET DATABASE MANAGER CONFIGURATION

データベース・マネージャー構成ファイル内の、個々の項目の値を返します。

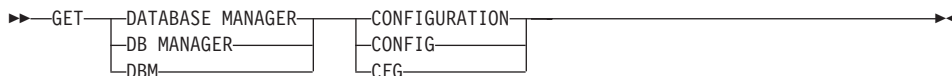
許可

なし

必須接続

なし、またはインスタンス。インスタンスとの接続は、ローカルの DBM 構成操作を実行する場合には必ずしも必要ではありませんが、リモートの DBM 構成操作の場合には必須です。リモート・インスタンスのデータベース・マネージャー構成を表示するには、まず最初にそのインスタンスと接続する必要があります。

コマンド構文



コマンド・パラメーター

なし

例

注: ノード・タイプとプラットフォームによって、どの構成パラメーターをリストするかが決まります。

以下に示すのは、GET DATABASE MANAGER CONFIGURATION の出力例です (AIX で発行)。

```
Database Manager Configuration
Node type = Database Server with local clients
Database manager configuration release level          = 0x0900
CPU speed (millisec/instruction) (CPUSPEED) = 4.000000e-05
Max number of concurrently active databases (NUMDB) = 8
Data Links support (DATALINKS) = NO
Federated Database System Support (FEDERATED) = NO
Transaction processor monitor name (TP_MON_NAME) =
Default charge-back account (DFT_ACCOUNT_STR) =
Java Development Kit 1.1 installation path (JDK11_PATH) =
Diagnostic error capture level (DIAGLEVEL) = 3
Diagnostic data directory path (DIAGPATH) =
Default database monitor switches
Buffer pool (DFT_MON_BUFPOOL) = OFF
```

GET DATABASE MANAGER CONFIGURATION

```

Lock (DFT_MON_LOCK) = OFF
Sort (DFT_MON_SORT) = OFF
Statement (DFT_MON_STMT) = OFF
Table (DFT_MON_TABLE) = OFF
Unit of work (DFT_MON_UOW) = OFF
SYSADM group name (SYSADM_GROUP) = BUILD
SYSCTRL group name (SYSCTRL_GROUP) =
SYSMAINT group name (SYSMAINT_GROUP) =

Database manager authentication (AUTHENTICATION) = SERVER
Cataloging allowed without authority (CATALOG_NOAUTH) = YES
Trust all clients (TRUST_ALLCLNTS) = YES
Trusted client authentication (TRUST_CLNTAUTH) = CLIENT
Default database path (DFTDBPATH) = /home/jessicae
Database monitor heap size (4KB) (MON_HEAP_SZ) = 56
UDF shared memory set size (4KB) (UDF_MEM_SZ) = 256
Java Virtual Machine heap size (4KB) (JAVA_HEAP_SZ) = 512
Audit buffer size (4KB) (AUDIT_BUF_SZ) = 0
Backup buffer default size (4KB) (BACKBUFSZ) = 1024
Restore buffer default size (4KB) (RESTBUFSZ) = 1024
Sort heap threshold (4KB) (SHEAPTHRES) = 20000
Directory cache support (DIR_CACHE) = YES
Application support layer heap size (4KB) (ASLHEAPSZ) = 15
Max requester I/O block size (bytes) (RQRIOBLK) = 32767
Query heap size (4KB) (QUERY_HEAP_SZ) = 1000
DRDA services heap size (4KB) (DRDA_HEAP_SZ) = 128
Priority of agents (AGENTPRI) = SYSTEM
Max number of existing agents (MAXAGENTS) = 200
Agent pool size (NUM_POOLAGENTS) = 4 (calculated)
Initial number of agents in pool (NUM_INITAGENTS) = 0
Max number of coordinating agents (MAX_COORDAGENTS) = MAXAGENTS
Max no. of concurrent coordinating agents (MAXCAGENTS) = MAX_COORDAGENTS
Max number of logical agents (MAX_LOGICAGENTS) = MAX_COORDAGENTS
Keep DARI process (KEEPDARI) = YES
Max number of DARI processes (MAXDARI) = MAX_COORDAGENTS
Initialize DARI process with JVM (INITDARI_JVM) = NO
Initial number of fenced DARI process (NUM_INITDARIS) = 0
Index re-creation time (INDEXREC) = RESTART
Transaction manager database name (TM_DATABASE) = 1ST_CONN
Transaction resync interval (sec) (RESYNC_INTERVAL) = 180
SPM name (SPM_NAME) =
SPM log size (SPM_LOG_FILE_SZ) = 256
SPM resync agent limit (SPM_MAX_RESYNC) = 20
SPM log path (SPM_LOG_PATH) =
TCP/IP Service name (SVCENAME) = xjessicae
APPC Transaction program name (TPNAME) =
IPX/SPX File server name (FILESERVER) =
IPX/SPX DB2 server object name (OBJECTNAME) =
IPX/SPX Socket number (IPX_SOCKET) = 879E

Discovery mode (DISCOVER) = SEARCH
Discovery communication protocols (DISCOVER_COMM) =
Discover server instance (DISCOVER_INST) = ENABLE
Directory services type (DIR_TYPE) = NONE
Directory path name (DIR_PATH_NAME) = /./:subsys/database/

```

GET DATABASE MANAGER CONFIGURATION

Directory object name	(DIR_OBJ_NAME) =
Routing information object name	(ROUTE_OBJ_NAME) =
Default client comm. protocols	(DFT_CLIENT_COMM) =
Maximum query degree of parallelism	(MAX_QUERYDEGREE) = ANY
Enable intra-partition parallelism	(INTRA_PARALLEL) = NO
No. of int. communication buffers(4KB)	(FCM_NUM_BUFFERS) = 512
Number of FCM request blocks	(FCM_NUM_RQB) = 256
Number of FCM connection entries	(FCM_NUM_CONNECT) = (FCM_NUM_RQB * 0.75)
Number of FCM message anchors	(FCM_NUM_ANCHORS) = (FCM_NUM_RQB * 0.75)

フィールドの意味は次のとおりです。

AGENT_STACK_SZ (OS/2 のみ)

オペレーティング・システムによって割り振りおよびコミットされたメモリーのエージェントごとのサイズ。このパラメーターによって、サーバー上にある各エージェントごとのページ数が示されます。

AGENTPRI

データベース・マネージャー・プロセスおよび特定のマシンの論理経路に割り当てられた実行優先順位。

ASLHEAPSZ

ローカルのクライアント・アプリケーションとデータベース・マネージャー・エージェント間の共用メモリーサイズ (ページ単位)。

AUDIT_BUF_SZ

データベースの監査に使用するバッファのサイズ (ページ単位)。

AUTHENTICATION

ユーザーの認証が行われる方法および箇所を判別します。値 CLIENT は、すべての認証がクライアントで行われることを示します。値が SERVER の場合には、ユーザー ID およびパスワードがクライアントからサーバーに送られ、結果として認証をサーバーで行うことができます。

BACKBUFSZ

バックアップ・ユーティリティーの呼び出し時にバッファ・サイズが指定されていない場合に、データベースをバックアップする際に使われるバッファのサイズ (ページ単位)。

CATALOG_NOAUTH

ユーザーがデータベースおよびノード、または DCS および ODBC ディレクトリーを、SYSADM 権限なしでカタログおよびアンカタログできるかどうかを指定します。このパラメーターの省略時値 (0) は、SYSADM 権限が必要であることを示します。このパラメーターが 1 (はい) に設定される場合、SYSADM 権限は必要ありません。

COMM_BANDWIDTH

区分データベース・システムのデータベース区画サーバー間で、ある操作を実行するのにかかる費用を算定するために SQL 最適化プログラムが使用する、通信帯域幅に対して計算される値 (MB/秒)。

CONN_ELAPSE (MPP のみ)

このパラメーターは、2つのノード間で TCP/IP 接続を確立するのにかかる秒数を指定します。このパラメーターで指定した時間内に試行が完了すれば、通信が確立されます。失敗した場合、別の試行が通信を確立します。接続の試行が MAX_CONNRETRIES パラメーターで指定した回数に達した場合、およびタイムアウトになった場合、エラーが戻されます。

CPUSPEED

ある操作を実行するのにかかる費用を算定するのに、SQL 最適化プログラムが使用する CPU 速度 (命令ごとにミリ秒)。このパラメーターの値は、データベース・マネージャーのインストール時に自動的に設定されますが、テスト・システムの実稼働環境に合わせて修正することも、ハードウェアをアップグレードする時の影響に合わせてアクセスすることもできます。

DATALINKS

このパラメーターは、データ・リンク・サポートが使用可能かどうかを指定します。

DFT_ACCOUNT_STR

省略時の会計ストリング。

DFT_CLIENT_ADPT

このパラメーターは、DCE ディレクトリー・サービスから抽出されるサーバー名を持つ NETBIOS プロトコル用の省略時クライアント・アダプターを定義します。このパラメーターは DCE の場合のみ使用できます。

DFT_CLIENT_COMM

特定のインスタンスのクライアント・アプリケーションがリモート接続に用いることのできる通信プロトコルを示します。DCE を構成する場合のみ使用されます。

DFT_MON_BUFPOOL

スナップショット・モニターのバッファー・プール・スイッチの省略時値。

DFT_MON_LOCK

スナップショット・モニターのロック・スイッチの省略時値。

GET DATABASE MANAGER CONFIGURATION

DFT_MON_SORT

スナップショット・モニターの分類スイッチの省略時値。

DFT_MON_STMT

スナップショット・モニターのステートメント・スイッチの省略時値。

DFT_MON_TABLE

スナップショット・モニターの表スイッチの省略時値。

DFT_MON_UOW

スナップショット・モニターの作業単位 (UOW) スwitchの省略時値。

DFTDBPATH

省略時のデータベース・パス。データベースの作成時にパスが指定されなかった場合には、このパラメーターに指定されたパスにデータベースが作成されます。

DIAGLEVEL

診断エラーのキャプチャー・レベルによって、エラー・ログ・ファイル (db2diag.log) に記録される診断エラーの重大度を判別します。

DIAGPATH

DB2 診断情報の完全修飾パス。

DIR_CACHE

ディレクトリー・キャッシュのサポート。YES に設定された場合、データベース・ノード、および DCS ディレクトリー・ファイルはメモリーにキャッシュされます。これによって、ディレクトリー・ファイルの入出力は取り除かれ、ディレクトリー情報の前入力に必要なディレクトリー探索も最小限に抑えることができるので、接続費用が大いに削減されます。

DIR_OBJ_NAME

DCE 名前空間でのオブジェクト名。この場合のオブジェクト名は、ディレクトリー内のデータベース・マネージャー・インスタンス (またはデータベース) を表します。この値と *dir_path_name* 値を連結すると、*dir_type* パラメーターで指定されたディレクトリー・サービスが管理する名前空間にある、データベース・マネージャー・インスタンスまたはデータベースを固有に識別できる大域名になります。

DIR_PATH_NAME

DCE 名前空間でのディレクトリー・パス名。この値と *dir_obj_name* パラメーターの値を組み合わせることで、大域名空間におけるデータベース・マネージャー・インスタンスの固有名を構成できます。

DIR_TYPE

ディレクトリー・サービスのタイプ。データベース・マネージャー・インスタンスが DCE 大域ディレクトリー・サービスを使用するかどうかを示します。

DISCOVER

このパラメーターは、クライアントまたはサーバーでサポートされるディスカバリー要求のタイプを定義します。ディスカバリー要求は、クライアント構成アシスタントまたはコントロール・センター・ツールから発行できます。探索ディスカバリーをサポートするには、SEARCH を指定します。探索ディスカバリーでは、DB2 クライアントが DB2 データベースのネットワークを探索します。認識済みディスカバリーをサポートするには、KNOWN を指定します。認識済みディスカバリーでは、ユーザーが指定した管理サーバーに対してディスカバリー要求が発行されます。DISABLE を指定すると、クライアントまたはサーバーはディスカバリー要求のどんなタイプのサポートも使用できません。

DISCOVER_COMM

クライアントが探索ディスカバリー要求を発行するために使用し、サーバーが探索ディスカバリー要求を listen するために使用する通信プロトコルを、このパラメーターは定義します。コンマで区切って、複数のプロトコルを指定することができます。パラメーターをブランクにしておくことも可能です。サポートするプロトコルは、TCPIP および NETBIOS です。

DISCOVER_INST

このパラメーターは、インスタンスのディスカバリーをクライアントで使用可能または使用不能にします。

DOS_RQRIOBLK

DOS リクエストの入出力ブロック・サイズ。DOS クライアント (OS/2 で実行中の DOS クライアントも含む) の場合にのみ適用されます。このパラメーターによって、クライアントおよびサーバーに割り振られる入出力ブロックのサイズが制御されます。

DRDA_HEAP_SZ

DRDA ヒープのサイズ (ページ単位) を示します。このヒープは、DRDA AS と DB2 コネクトが使用します。

FCM_NUM_ANCHORS

このパラメーターは、FCM メッセージ・アンカーの数を指定します。エージェントは、メッセージ・アンカーを使用して互いにメッセージを送信します。

GET DATABASE MANAGER CONFIGURATION

FCM_NUM_BUFFERS

このパラメーターは、インスタンス内のノード間での内部通信（メッセージ）に使用する 4KB バッファの数を指定します。

FCM_NUM_CONNECT

このパラメーターは、FCM 接続項目の数を指定します。エージェントは接続項目を使用して、互いにデータを渡します。

FCM_NUM_RQB

このパラメーターは、FCM 要求ブロックの数を指定します。要求ブロックは、FCM デーモンとエージェントの間で情報を渡す媒体です。

FEDERATED

合同データベース・オブジェクト・サポート。YES に設定すると、インスタンスは DB2 ファミリー、および他のデータベース・マネージャーが管理するデータに通称を使ってアクセスできます。

FILESERVER

IPX/SPX ファイル・サーバー名。データベース・マネージャー・サーバー・インスタンスのインターネットワーク・アドレスを登録した Novell NetWare ファイル・サーバーの名前を指定します。

注: 次の文字は無効です。 / ¥ : ; , * ?

INDEXREC

無効な索引が再作成される条件を示します。このパラメーターが使用されるのは、データベース構成パラメーター *indexrec* が、SYSTEM に設定されている場合です。

以下の出力値を使用することもできます。

- ACCESS
- RESTART

INITDARI_JVM

このパラメーターは、隔離された各 DARI プロセスが、開始時に Java 仮想マシン (JVM) をロードするかどうかを示します。このパラメーターによって、特に *num_initdaris* パラメーターと一緒に使用すると、隔離された Java ストアード・プロシーチャーの初期起動時間を削減できます。しかし、Java でない隔離ストアード・プロシーチャーは JVM を必要としないので、初期ロード時間が長くなる可能性があります。

INTRA_PARALLEL

このパラメーターは、データベース・マネージャーが区画内並列処理を使用できるかどうかを指定します。

GET DATABASE MANAGER CONFIGURATION

対称マルチプロセッサ (SMP) 環境では、このパラメーターの省略時値は YES です。SMP 以外の環境では、このパラメーターの省略時値は NO です。このパラメーターは、区分および非区分データベース・システムの両方で使用できます。このパラメーターが YES の場合に、並列パフォーマンス向上を利用できる操作には、データベース照会および索引作成があります。

IPX_SOCKET

IPX/SPX ソケット番号。「予約済み」のソケット番号を指定し、DB2 サーバーの IPX/SPX インターネットワーク・アドレス中の接続エンドポイントを表示します。

JAVA_HEAP_SZ

Java インタープリターが使用するヒープの最大サイズを判別します。非区分データベース・システムの場合、インスタンスに 1 つのヒープが割り当てられます。区分データベース・システムの場合、各データベース区画サーバーに 1 つのヒープが割り当てられます。

JDK11_PATH

このパラメーターは、Java 開発者キット 1.1 がインストールされているディレクトリーを指定します。Java インタープリターが使用する **CLASSPATH** および他の環境変数は、このパラメーターの値から計算されます。

KEEPDARI

各 DARI 呼び出しの後、DARI プロセスを保持するかどうかを示します。NO であれば、新規の DARI 処理が作成され、それぞれの DARI 呼び出しを終了させます。YES であれば、後続の DARI 呼び出しのために、その処理は拒否されます。また、関連したユーザー・アプリケーションがあるときにのみ終了されます。

MAX_CONNRETRIES (MPP のみ)

2 つのノード間の通信を確立する試行が、CONN_ELAPSE パラメーターで指定済みの値に達したために失敗した (たとえば、TCP/IP 通信を確立しようとしてタイムアウトになった) 場合、MAX_CONNRETRIES は、ノードで接続を再試行できる回数を指定します。このパラメーターに指定した値を超えると、エラーが戻されます。

MAX_COORDAGENTS

このパラメーターは、ノードに同時に存在できる調整エージェントの最大数を判別します。

GET DATABASE MANAGER CONFIGURATION

MAX_LOGICAGENTS

インスタンスに接続可能なアプリケーションの最大数を制御します。このパラメーターが省略時値 (`max_coordagents`) よりも大きい値に設定されると、コンセントレーター機能が活動化されます。

MAX_QUERYDEGREE

このパラメーターは、データベース・マネージャーのこのインスタンスで実行しているあらゆる SQL ステートメントに使用される並列処理の程度の最大値を指定します。ステートメントの実行時に、SQL ステートメントは並列操作のこの数値を超えて使用することはありません。マルチノード・システムでは、このパラメーターは単一ノードの並列処理の程度に適用されます。

MAX_TIME_DIFF (MPP のみ)

各ノードには、独自のシステム・クロックがあります。このパラメーターは、`db2nodes.cfg` ファイルにリストしたノード間に許可する、最大時差を指定します。

MAXAGENTS

どのデータベースが使用されているかにかかわらず、データベース・マネージャー・エージェントが同時にノードに存在できる最大数です。

MAXCAGENTS

データベース・マネージャー・トランザクションを同時に実行できるデータベース・マネージャー・エージェントの最大数。 `maxagents` の値を超えることはありません。

MAXDARI

データベース・サーバーに常駐することができる DARI プロセスの最大数。 `maxagents` の値を超えることはありません。

MAXTOTFILOP (OS/2 のみ)

アプリケーションごとのオープン・ファイルの最大数。あるデータベースに接続された特定のプロセスによって使用できるデータベースおよびアプリケーションのファイル・ハンドルの合計数を定義します。

MIN_PRIV_MEM (OS/2 のみ)

最小コミット済み私用メモリー。データベース・マネージャー・インスタンスの開始時 (`db2start`) に、データベース・サーバー・プロセスが私用仮想記憶として予約するページ数を示します。

MON_HEAP_SZ

データベース・システム・モニターのヒープ・サイズ。データベース・システム・モニターに割り振られたメモリーのサイズ (4KB ページ単位) を示します。

NNAME (OS/2 のみ)

ノードまたはワークステーションの名前。データベース・クライアントは *nname* を使用して、NetBIOS を使用するデータベース・サーバー (database server)・ワークステーションにアクセスします。したがって、データベース・サーバー (database server)・ワークステーションが *nname* で指定された名前を変更してしまうと、そのデータベース・サーバー (database server)・ワークステーションにアクセスするクライアントのすべてが、そのデータベース・サーバー (database server)・ワークステーションを再びカタログ化し直して、その新規の名前をわざわざ指定しなければならなくなります。

nodetype (ノード・タイプ)

ノードが、ローカルおよびリモート・クライアントのデータベース・サーバーとして構成されているか、またはクライアント、ローカル・クライアントを持つデータベース・サーバー、ローカルおよびリモート・クライアントを持つ区分データベース・サーバー、それともローカル・クライアントを持つサテライトとして構成されているかを示します。

NOTIFYLEVEL (Windows NT のみ)

このパラメーターは、通知ファイルに書き込まれるメッセージの重大度を決定するのに使用されます。

NUM_INITAGENTS

このパラメーターは、データベース・マネージャーの開始時に、エージェント・プールで作成されたエージェントの初期数を判別します。

NUM_INITDARIS

このパラメーターは、データベース・マネージャーの開始時に、DARI プールで作成されたアイドル隔離 DARI プロセスの初期数を判別します。このパラメーターを設定すると、隔離されたストアド・プロシージャの初期起動時間を削減できます。このパラメーターは、*keepdari* が指定されないと無視されます。

NUM_POOLAGENTS

このパラメーターは、エージェント・プールが増加できるサイズを指定します。エージェント・プールには、アイドル・エージェント (DB2/6000 バージョン 2 の場合)、MPP および SMP 関連サブエージェントが入っています。エージェントをさらに作成する場合、それらは実行を終えると終了し、プールには戻されません。

このパラメーターの値が、実行時に他の構成パラメーターを使用して計算される場合、

292ページの『GET DATABASE MANAGER CONFIGURATION』の

GET DATABASE MANAGER CONFIGURATION

出力中に示された値の右に (calculated) というラベルが表示されず。出力に -1 (calculated) と表示されている場合、その要求はクライアントから発行され、その値は使用不能であったことを示しています。

旧データベース・マネージャー構成パラメーターの *max_idleagents* は、608ページの『UPDATE DATABASE MANAGER CONFIGURATION』を介して更新することができ、*num_poolagents* に更新すると解釈されます。

NUMDB

同時に活動状態にできる (つまり、複数のアプリケーションを別々に接続できる) ローカル・データベースの最大数。

OBJECTNAME

このパラメーターは、NetWare ファイル・サーバー上のオブジェクトとしてデータベース・マネージャー・サーバー・インスタンスを表示します。NetWare ファイル・サーバーにはサーバーの IPX/SPX アドレスが保管され検索されます。値は大文字で入力しなければなりません。その値は NetWare ファイル・サーバー上で固有である必要があり、IPX/SPX ネットワーク全体で固有にすることをお勧めします。

注: 次の文字は無効です。 / ¥ ; , * ?

PRIV_MEM_THRESH (OS/2 のみ)

私用メモリーのしきい値。あるクライアントとの接続が終了するときに、そのクライアントと関連したメモリーをサーバーが解放しない場合のしきい値を設定します。

QUERY_HEAP_SZ

照会ヒープに割り振ることのできるメモリーの最大サイズ (ページ単位)。照会ヒープは、エージェントの私用メモリーにある各照会を保管するのに使われます。

release (データベース・マネージャー構成のリリース・レベル)

構成ファイルのリリース・レベル。

RESTBUFSZ

復元ユーティリティーの呼び出し時にバッファー・サイズが指定されていない場合に、データベースを復元する際に使われるバッファーのサイズ (4KB ページ単位)。

RESYNC_INTERVAL

トランザクション・マネージャー (TM) またはリソース・マネージャー (RM) が、TM または RM で見つかった未解決および未確定なト

GET DATABASE MANAGER CONFIGURATION

ランザクションすべての回復を再試行する時間間隔 (秒単位)。このパラメーターは、トランザクションが分散作業単位 (DUOW) 環境で実行中の場合にかぎり適用されます。

ROUTE_OBJ_NAME

経路指定情報のオブジェクト名。DRDA サーバーへのアクセスを試みるクライアント・アプリケーションすべてが使用する、経路指定情報のオブジェクト項目の省略時名を示します。DCE を構成する場合のみ使用されます。

RQRIOBLK

クライアント入出力のブロック・サイズ。リモート・アプリケーションとデータベース・サーバー上のデータベース・エージェント間の通信バッファのサイズ (バイト単位) を示します。

SHEAPTHRES

インスタンス全体の分類に取り分けることのできるメモリーの合計サイズ (ページ単位) を制限します。

SPM_LOG_FILE_SZ

このパラメーターは、同期点マネージャー (SPM) のログ・ファイルを 4KB ページ単位で識別します。ログ・ファイルは、`sqllib` の下にある、`spmlog` サブディレクトリーに入れられます。ログ・ファイルは、SPM が最初に始動した時点で作成されます。

SPM_LOG_PATH

このパラメーターは、同期点マネージャー (SPM) ログが書き込まれるディレクトリーを指定します。省略時には、ログは `sqllib` ディレクトリーに書き込まれます。このため、高ボリューム・トランザクション環境では、入出力ボトルネックが発生する場合があります。このパラメーターを使用して、現行の `sqllib` ディレクトリーよりも高速なディスクに SPM ログ・ファイルを置いてください。このようにすると、SPM エージェント間での並行性が向上します。

SPM_MAX_RESYNC

このパラメーターによって、再同期操作を同時に実行できるエージェントの数を識別します。

SPM_NAME

このパラメーターによって、同期点マネージャー (SPM) インスタンスの名前をデータベース・マネージャーに示します。`spm_name` は、システム・データベース・ディレクトリーに定義しておく必要があります。リモートの場合には、さらにノード・ディレクトリーでも定義しておく必要があります。

GET DATABASE MANAGER CONFIGURATION

SS_LOGON (OS/2 のみ)

このパラメーターの省略時値を受け入れると、DB2START または DB2STOP を発行する前に LOGON ユーザー ID およびパスワードが必要になります。

START_STOP_TIME (MPP のみ)

このパラメーターは、すべてのノードが 576ページの『START DATABASE MANAGER』、582ページの『STOP DATABASE MANAGER』、または 152ページの『ADD NODE』に回答しなければならない時間を分単位で指定します。

SVCENAME

サーバーでデータベース・マネージャー構成ファイルを更新するのに使用する名前です。この値は、*services* ファイルで指定した Connection Service 名と同じでなければなりません。

SYSADM_GROUP

システム管理 (*sysadm*) 権限を持つグループ名をデータベース・マネージャー・インスタンスに定義します。これは、データベース・マネージャー内の権限の最大レベルであり、すべてのデータベース・オブジェクトを制御します。

SYSCTRL_GROUP

システム制御 (*sysctrl*) 権限を持つグループ名をデータベース・マネージャー・インスタンスに定義します。このレベルには、システム・リソースに影響を及ぼす操作を可能にする特権がありますが、データに対する直接アクセスは行えません。

SYSMAINT_GROUP

システム保守 (*sysmaint*) 権限を持つグループ名をデータベース・マネージャー・インスタンスに定義します。このレベルには、あるインスタンスと関連したデータベースすべてに対する保守操作を可能にする権限がありますが、データに対する直接アクセスは行えません。

TM_DATABASE

各 DB2 インスタンスごとのトランザクション・マネージャー (TM) データベースの名前。

TP_MON_NAME

使用されるトランザクション処理 (TP) モニター製品の名前。

TPNAME

APPC 通信プロトコルを使用して、データベース・マネージャー・イン

スタンスに対して割り振り要求を発行した場合に、データベース・クライアントが使用する必要のあるリモート・トランザクション・プログラムの名前。

TRUST_ALLCLNTS

このパラメーターおよび TRUST_CLNTAUTH パラメーターは、ユーザーが妥当性検査されるデータベース環境を判別します。このパラメーターの省略時値を受け入れると、すべてのクライアントは承認されたクライアントとして扱われます。これはクライアントがセキュリティー・レベルを使用できるということであり、クライアントでそのユーザーを妥当性検査できます。プラットフォームまたはデータベース・プロトコルに基づいた、特定のクライアントに対してサーバーを保護するために、他のオプションを使用することができます。

TRUST_CLNTAUTH

このパラメーターおよび TRUST_ALLCLNTS パラメーターは、ユーザーが妥当性検査されるデータベース環境を判別します。このパラメーターの省略時値を受け入れると、承認されたクライアントのすべてのユーザーはクライアントで妥当性検査されます。

UDF_MEM_SZ

分離されたユーザー定義機能 (UDF) の場合、データベース・プロセスと UDF との間で共用されるメモリーの省略時の割り振りを示します。非分離プロセスの場合は、私用メモリー設定のサイズを示します。どちらにもあてはまる場合には、このメモリーを使用して、データを UDF に渡したり、データベースに戻したりします。

使用上の注意

リモート・インスタンス (または別のローカル・インスタンス) への接続が存在する場合、それらのインスタンスに接続されたサーバーのデータベース・マネージャー構成パラメーターが返されます。そのようなインスタンスが存在しない場合には、ローカルのデータベース・マネージャー構成パラメーターが返されます。

エラーが生じた場合には、返された情報は無効になります。構成ファイルが無効な場合には、エラー・メッセージが戻されます。そのような場合には、データベース・マネージャーを再インストールして回復を実行する必要があります。

データベース・マネージャーの出荷時の省略時値に構成パラメーターを設定するには、533ページの『RESET DATABASE MANAGER CONFIGURATION』を使用してください。

GET DATABASE MANAGER CONFIGURATION

上記のパラメーターの詳細については、[管理の手引き](#) を参照してください。

以下も参照

533ページの『[RESET DATABASE MANAGER CONFIGURATION](#)』

608ページの『[UPDATE DATABASE MANAGER CONFIGURATION](#)』

GET DATABASE MANAGER MONITOR SWITCHES

データベース・システム・モニター・スイッチの状況を表示します。モニター・スイッチは、データベース活動情報を収集するように、データベース・システム・モニターに指示します。データベース・システム・モニター・インターフェースを使用している各アプリケーションには、それ自体のモニター・スイッチの集合があります (310ページの『GET MONITOR SWITCHES』を参照)。監視中の任意のアプリケーションがオンの場合、データベース・マネージャー・レベル・スイッチがオンになります。監視中の任意のアプリケーション用に、現在データベース・システム・モニターがデータを収集しているかどうかを判別するために、このコマンドを使用します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

インスタンスまたはデータベース

- インスタンスへの接続や、データベースへの接続がない場合、省略時のインスタンス接続が作成されます。
- インスタンスへの接続とデータベース接続の両方がある場合、インスタンス接続が使用されます。

リモート・インスタンス、または別のローカル・インスタンスの設定値を表示するには、まず最初にそのインスタンスと接続することが必要です。

コマンド構文



コマンド・パラメーター

AT NODE nodenum

データベース・マネージャーのモニター・スイッチの状況を表示するノードを指定します。

GET DATABASE MANAGER MONITOR SWITCHES

GLOBAL

区分データベース・システム内のすべてのノードの集合結果を戻します。結果を集める方法については、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。

例

以下に示すのは、GET DATABASE MANAGER MONITOR SWITCHES の出力例です。

```
DBM System Monitor Information Collected
Switch list for node 1
Buffer Pool Activity Information (BUFFERPOOL) = ON   06-11-1997 10:11:01.738377
Lock Information (LOCK) = OFF
Sorting Information (SORT) = ON   06-11-1997 10:11:01.738400
SQL Statement Information (STATEMENT) = OFF
Table Activity Information (TABLE) = OFF
Unit of Work Information (UOW) = ON   06-11-1997 10:11:01.738353
```

使用上の注意

6 つの記録スイッチ (BUFFERPOOL、LOCK、SORT、STATEMENT、TABLE、および UOW) がありますが、省略時にはすべてオフになっています。613ページの『UPDATE MONITOR SWITCHES』を使用する場合には、どれかをオンにすることになります。特定のスイッチをオンにすると、このコマンドはそのスイッチがオンになった時点のタイム・スタンプも表示します。

データベース・モニターのデータ要素およびモニター・グループの要約は、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。

以下も参照

310ページの『GET MONITOR SWITCHES』

312ページの『GET SNAPSHOT』

535ページの『RESET MONITOR』

613ページの『UPDATE MONITOR SWITCHES』

GET INSTANCE

DB2INSTANCE 環境変数の値を返します。

許可

なし

必須接続

なし

コマンド構文

▶▶—GET INSTANCE—◀◀

コマンド・パラメーター

なし

例

以下に示すのは、GET INSTANCE の出力例です。

```
The current database manager instance is: smith
```

GET MONITOR SWITCHES

GET MONITOR SWITCHES

現行セッションのデータベース・システム・モニター・スイッチの状況を表示します。モニター・スイッチは、データベース活動情報を収集するように、データベース・システム・モニターに指示します。データベース・システム・モニター・インターフェースを使用している各アプリケーションには、それ自体のモニター・スイッチの集合があります。このコマンドはそれらを表示します。データベース・マネージャー・レベル・スイッチを表示するには、307ページの『GET DATABASE MANAGER MONITOR SWITCHES』を使用します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

インスタンス。インスタンス接続がない場合、省略時のインスタンス接続が作成されます。

リモート・インスタンス、または別のローカル・インスタンスの設定値を表示するには、まず最初にそのインスタンスと接続することが必要です。

コマンド構文

```
▶▶ GET MONITOR SWITCHES [ AT NODE nodenum | GLOBAL ] ▶▶
```

コマンド・パラメーター

AT NODE *nodenum*

モニター・スイッチの状況を表示するノードを指定します。

GLOBAL

区分データベース・システム内のすべてのノードの集合結果を戻します。結果を集める方法については、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。

例

以下に示すのは、GET MONITOR SWITCHES の出力例です。

```

Monitor Recording Switches
Switch list for node 1
Buffer Pool Activity Information (BUFFERPOOL) = ON 02-20-1997 16:04:30.070073
Lock Information (LOCK) = OFF
Sorting Information (SORT) = OFF
SQL Statement Information (STATEMENT) = ON 02-20-1997 16:04:30.070073
Table Activity Information (TABLE) = OFF
Unit of Work Information (UOW) = ON 02-20-1997 16:04:30.070073

```

使用上の注意

6 つの記録スイッチ (BUFFERPOOL、LOCK、SORT、STATEMENT、TABLE、および UOW) がありますが、省略時にはすべてオフになっています。613ページの『UPDATE MONITOR SWITCHES』を使用する場合には、どれかをオンにすることになります。特定のスイッチをオンにすると、このコマンドはそのスイッチがオンになった時点のタイム・スタンプも表示します。

データベース・モニターのデータ要素およびモニター・グループの要約は、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。

以下も参照

307ページの『GET DATABASE MANAGER MONITOR SWITCHES』

312ページの『GET SNAPSHOT』

535ページの『RESET MONITOR』

613ページの『UPDATE MONITOR SWITCHES』

GET SNAPSHOT

GET SNAPSHOT

状況情報を収集して、ユーザー用に出力を形式設定します。戻された情報は、コマンドを発行した時点でのデータベース・マネージャー操作状況のスナップショットを表しています。

効力範囲

区分データベース環境では、このコマンドは、`db2nodes.cfg` ファイル中のどのノードからでも呼び出すことができます。このコマンドは、呼び出されたノードまたは区画上でのみ活動します。

許可

以下のどれかが必要です。

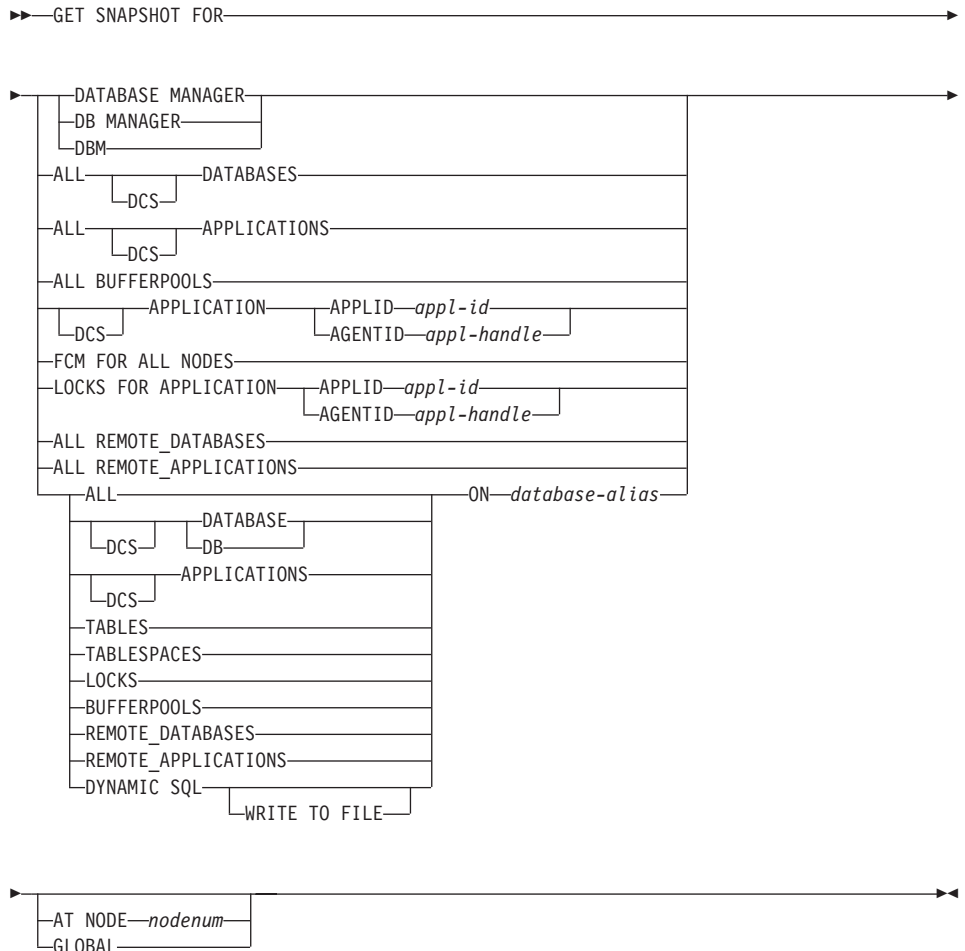
- `sysadm`
- `sysctrl`
- `sysmaint`

必須接続

インスタンス。インスタンス接続がない場合、省略時のインスタンス接続が作成されます。

リモート・インスタンスのスナップショットを獲得するには、まず最初にそのインスタンスと接続する必要があります。

コマンド構文



注: 統計を入手するには、モニター・スイッチをオンにする必要があります (613ページの『UPDATE MONITOR SWITCHES』を参照してください)。

コマンド・パラメーター

DATABASE MANAGER

活動データベース・マネージャー・インスタンスの統計を提供します。

ALL DATABAS

現行ノード上の活動データベースすべてに関する一般統計を提供します。

GET SNAPSHOT

ALL APPLICATIONS

現行ノード上のデータベースに接続された活動アプリケーションすべてに関する情報を提供します。

ALL BUFFERPOOLS

活動データベースすべてのバッファ・プール活動に関する情報を提供します。

APPLICATION APPLID appl-id

指定された ID を持つアプリケーションの情報だけを提供します。特定のアプリケーション ID を獲得するには、364ページの『LIST APPLICATIONS』を使用してください。

APPLICATION AGENTID appl-handle

指定されたアプリケーション・ハンドルを持つアプリケーションの情報だけを提供します。アプリケーション・ハンドルは 32 ビットの数字で、現在実行中のアプリケーションを固有に識別できるものです。特定のアプリケーション・ハンドルを知りたい場合には、364ページの『LIST APPLICATIONS』を使用してください。

FCM FOR ALL NODES

高速コミュニケーション・マネージャー (FCM) 統計をすべてのノードに提供します。

LOCKS FOR APPLICATION APPLID appl-id

アプリケーション ID によって識別される、指定したアプリケーションによって保留されているロックすべてに関する情報を提供します。

LOCKS FOR APPLICATION AGENTID appl-handle

アプリケーション・ハンドルによって識別される、指定したアプリケーションによって保留されているロックすべてに関する情報を提供します。

ALL REMOTE_DATABASES

現行ノード上の活動リモート・データベースすべてに関する一般統計を提供します。

ALL REMOTE_APPLICATIONS

現行ノードに接続された活動リモート・アプリケーションすべてに関する情報を提供します。

ALL ON database-alias

指定したデータベースのアプリケーション、表、表スペース、バッファ・プール、およびロックすべてに関する一般統計および情報を提供します。

DATABASE ON database-alias

特定のデータベースの一般統計を提供します。

APPLICATIONS ON database-alias

特定のデータベースに接続されたアプリケーションすべてに関する情報を提供します。

TABLES ON database-alias

特定のデータベース内の表に関する情報を提供します。これには、TABLE 記録スイッチがオンになった後にアクセスのあった表だけが含まれます。

TABLESPACES ON database-alias

特定のデータベースに表スペースに関する情報を提供します。

LOCKS ON database-alias

特定のデータベースに接続された各アプリケーションが保留するロックすべてに関する情報を提供します。

BUFFERPOOLS ON database-alias

指定したデータベースのバッファ・プール活動に関する情報を提供します。

REMOTE_DATABASES ON database-alias

指定されたデータベースの活動リモート・データベースすべてに関する一般統計を提供します。

REMOTE_APPLICATIONS ON database-alias

指定されたデータベースのリモート・アプリケーションに関する情報を提供します。

DYNAMIC SQL ON database-alias

データベースに対して SQL ステートメント・キャッシュの内容の瞬間ピクチャーを戻します。

WRITE TO FILE

スナップショットの結果が、サーバーでファイルに保管されるとともに、クライアントに戻されることを指定します。このコマンドは、データベース接続でのみ有効です。その後スナップショット・データは、表関数 SYSFUN.SQLCACHE_SNAPSHOT を介して、呼び出しが行われた同じ接続で照会することができます。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。

DCS 指定された文節に従って、このキーワードは以下のものに関する統計を要求します。

GET SNAPSHOT

- DB2 コネクト・ゲートウェイで現在実行されている特定の DCS アプリケーション
- すべての DCS アプリケーション
- 特定の DCS データベースに現在接続されているすべての DCS アプリケーション
- 特定の DCS データベース
- すべての DCS データベース

AT NODE nodenum

指定されたノードの結果を戻します。

GLOBAL

区分データベース・システム内のすべてのノードの集合結果を戻します。結果を集める方法については、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。

例

以下の出力例のリストでは、適切なデータベース・システム・モニターの記録スイッチがオンになっていないために、情報の一部が利用不可になっている場合があります (613ページの『UPDATE MONITOR SWITCHES』を参照してください)。情報が利用できない場合、Not Collected が出力に表示されます。

以下の出力リストに表示されたフィールドの詳細については、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。

以下に示すのは、データベース・マネージャー情報を要求した結果表示された一般的な出力です。

```
Database Manager Snapshot
Node type                      = Database Server with
                                local clients
Instance name                   = smith
Number of nodes in DB2 instance = 0
Database manager status        = Active
Product name                    =
Product identification          =
Service level                   =
Sort heap allocated             = 0
Post threshold sorts            = 0
Piped sorts requested           = 0
Piped sorts accepted            = 0
Start Database Manager timestamp = 02-25-1999 13:26:53.126518
Last reset timestamp            =
Snapshot timestamp              = 02-25-1999 13:45:42.257720
Remote connections to db manager = 0
Remote connections executing in db manager = 0
```

```

Local connections = 1
Local connections executing in db manager = 0
Active local databases = 1
High water mark for agents registered = 3
High water mark for agents waiting for a token = 0
Agents registered = 3
Agents waiting for a token = 0
Idle agents = 1
Committed private Memory (Bytes) = 3670016
Buffer Pool Activity Information (BUFFERPOOL) = ON 02-25-1999 13:32:14
Lock Information (LOCK) = ON 02-25-1999 13:32:40
Sorting Information (SORT) = ON 02-25-1999 13:32:40
SQL Statement Information (STATEMENT) = ON 02-25-1999 13:32:14
Table Activity Information (TABLE) = ON 02-25-1999 13:32:40
Unit of Work Information (UOW) = ON 02-25-1999 13:32:14
Agents assigned from pool = 2
Agents created from empty pool = 3
Agents stolen from another application = 0
High water mark for coordinating agents = 3
Max agents overflow = 0
Hash joins after heap threshold exceeded = 0
Total number of gateway connections = 0
Current number of gateway connections = 0
Gateway connections waiting for host reply = 0
Gateway connections waiting for client reply = 0
Gateway inactive connection pool agents = 0
Gateway connection pool agents stolen = 0

```

以下に示すのは、データベース情報を要求した結果表示された一般的な出力です。

```

Database Snapshot
Database name = SAMPLE
Database path = /home/smith/smith/NODE0000/SQL00001/
Input database alias =
Database status = Active
Catalog node number = 0
Catalog network node name =
Operating system running at database server = AIX
Location of the database = Local
First database connect timestamp = 02-25-1999 13:31:33.886214
Last reset timestamp =
Last backup timestamp =
Snapshot timestamp = 02-25-1999 13:40:08.337902
High water mark for connections = 1
Application connects = 1
Secondary connects total = 0
Applications connected currently = 1
Appls. executing in db manager currently = 0
Agents associated with applications = 1
Maximum agents associated with applications = 1
Maximum coordinating agents = 1
Locks held currently = 1
Lock waits = 0
Time database waited on locks (ms) = 0

```

GET SNAPSHOT

```
Lock list memory in use (Bytes)           = 432
Deadlocks detected                       = 0
Lock escalations                         = 0
Exclusive lock escalations               = 0
Agents currently waiting on locks        = 0
Lock Timeouts                            = 0
Total sort heap allocated                 = 0

Total sorts                              = 0
Total sort time (ms)                     = 0
Sort overflows                           = 0
Active sorts                             = 0
High water mark for database heap        = 316084
Buffer pool data logical reads           = 1
Buffer pool data physical reads          = 0
Asynchronous pool data page reads        = 0
Buffer pool data writes                   = 0
Asynchronous pool data page writes       = 0
Buffer pool index logical reads           = 0
Buffer pool index physical reads          = 0
Asynchronous pool index page reads        = 0
Buffer pool index writes                  = 0
Asynchronous pool index page writes       = 0
Total buffer pool read time (ms)          = 0
Total buffer pool write time (ms)         = 0
Total elapsed asynchronous read time      = 0
Total elapsed asynchronous write time     = 0
Asynchronous read requests               = 0
LSN Gap cleaner triggers                  = 0
Dirty page steal cleaner triggers         = 0
Dirty page threshold cleaner triggers     = 0
Time waited for prefetch (ms)            = 0
Direct reads                             = 0
Direct writes                             = 0
Direct read requests                      = 0
Direct write requests                     = 0
Direct reads elapsed time (ms)            = 0
Direct write elapsed time (ms)            = 0
Database files closed                     = 0
Data pages copied to extended storage     = 0
Index pages copied to extended storage    = 0
Data pages copied from extended storage   = 0
Index pages copied from extended storage  = 0
Commit statements attempted               = 2
Rollback statements attempted             = 0
Dynamic statements attempted              = 10
Static statements attempted               = 2
Failed statement operations               = 0
Select SQL statements executed            = 2
Update/Insert/Delete statements executed = 0
DDL statements executed                   = 0
Internal automatic rebinds                = 0
Internal rows deleted                     = 0
Internal rows inserted                     = 0
Internal rows updated                     = 0
```



```

Internal commits = 1
Internal rollbacks = 0
Internal rollbacks due to deadlock = 0
Rows deleted = 0
Rows inserted = 0
Rows updated = 0
Rows selected = 16
Binds/precompiles attempted = 0
Log space available to the database (Bytes) = 0
Log space used by the database (Bytes) = 0
Maximum secondary log space used (Bytes) = 0
Maximum total log space used (Bytes) = 0
Secondary logs allocated currently = 0
Log pages read = 0
Log pages written = 0
Appl id holding the oldest transaction = 0
Package cache lookups = 2
Package cache inserts = 1
Package cache overflows = 0
Package cache high water mark (Bytes) = 108757
Application section lookups = 10
Application section inserts = 1
Catalog cache lookups = 1
Catalog cache inserts = 1
Catalog cache overflows = 0
Catalog cache heap full = 0
Number of hash joins = 0
Number of hash loops = 0
Number of hash join overflows = 0
Number of small hash join overflows = 0

```

以下に示すのは、DCS データベース情報を要求した結果表示された一般的な出力です。

```

DCS Database Snapshot
DCS database name = DCSDB
Host database name = GILROY
First database connect timestamp = 02-25-1999 17:00:05.003421
Most recent elapsed time to connect = 0.001200
Most recent elapsed connection duration = 3.443780
Host response time (sec.ms) = 0.000320
Last reset timestamp =
Number of SQL statements attempted = 12
Commit statements attempted = 6
Rollback statements attempted = 2
Failed statement operations = 4
Total number of gateway connections = 0
Current number of gateway connections = 1
Gateway conn. waiting for host reply = 0
Gateway conn. waiting for client reply = 1
Gateway communication errors to host = 0
Timestamp of last communication error = None
High water mark for gateway connections = 1

```

GET SNAPSHOT

```
Rows selected                = 0
Outbound bytes sent          = 0
Outbound bytes received      = 0
```

以下に示すのは、(アプリケーション ID、アプリケーション・ハンドル、全アプリケーション、または特定のデータベース上の全アプリケーションのいずれかを指定して) アプリケーション情報を要求した結果表示された一般的な出力です。

```
Application Snapshot
Application handle           = 3
Application status          = UOW Waiting
Status change time         = 02-25-1999 13:33:41.446676
Application code page       = 819
Application country code    = 1
DUOW correlation token      = *LOCAL.smith.990225183133
Application name            = db2bp
Application ID              = *LOCAL.smith.990225183133
Sequence number            = 0001
Connection request start timestamp = 02-25-1999 13:31:33.886214
Connect request completion timestamp = 02-25-1999 13:31:34.434114
Application idle time       = 6 minutes and 42 seconds
Authorization ID            = SMITH
Client login ID             = smith
Configuration NNAME of client =
Client database manager product ID = SQL06000
Process ID of client application = 27918
Platform of client application = AIX
Communication protocol of client = Local Client
Outbound communication address =
Outbound communication protocol = APPC
Inbound communication address =
Database name               = SAMPLE
Database path               = /home/smith/smith/
                             NODE0000/SQL00001/
Client database alias       = sample
Input database alias        =
Last reset timestamp        =
Snapshot timestamp          = 02-25-1999 13:40:23.773540
The highest authority level granted =
  Direct DBADM authority
  Direct CREATETAB authority
  Direct BINDADD authority
  Direct CONNECT authority
  Direct CREATE_NOT_FENC authority
  Direct IMPLICIT_SCHEMA authority
  Direct LOAD authority
  Indirect SYSADM authority
  Indirect CREATETAB authority
  Indirect BINDADD authority
  Indirect CONNECT authority
  Indirect IMPLICIT_SCHEMA authority
  Indirect LOAD authority
Coordinating node number    = 0
```

```

Current node number                = 0
Coordinator agent process or thread ID = 26160
Agents stolen                       = 0
Agents waiting on locks             = 0
Maximum associated agents           = 1
Priority at which application agents work = 0
Priority type                        = Dynamic
Locks held by application           = 1
Lock waits since connect            = 0
Time application waited on locks (ms) = 0
Deadlocks detected                  = 0
Lock escalations                    = 0
Exclusive lock escalations          = 0
Number of Lock Timeouts since connected = 0
Total time UOW waited on locks (ms) = 0

Total sorts                        = 0
Total sort time (ms)                = 0
Total sort overflows                = 0
Data pages copied to extended storage = 0
Index pages copied to extended storage = 0
Data pages copied from extended storage = 0
Index pages copied from extended storage = 0
Buffer pool data logical reads      = 1
Buffer pool data physical reads     = 0
Buffer pool data writes             = 0
Buffer pool index logical reads     = 0
Buffer pool index physical reads    = 0
Buffer pool index writes            = 0
Total buffer pool read time (ms)    = 0
Total buffer pool write time (ms)   = 0
Time waited for prefetch (ms)      = 0
Direct reads                        = 0
Direct writes                       = 0
Direct read requests                = 0
Direct write requests               = 0
Direct reads elapsed time (ms)      = 0
Direct write elapsed time (ms)     = 0
Number of SQL requests since last commit = 5
Commit statements                   = 2
Rollback statements                  = 0
Dynamic SQL statements attempted    = 10
Static SQL statements attempted     = 2
Failed statement operations          = 0
Select SQL statements executed       = 2
Update/Insert/Delete statements executed = 0
DDL statements executed              = 0
Internal automatic rebinds          = 0
Internal rows deleted                = 0
Internal rows inserted               = 0
Internal rows updated                = 0
Internal commits                     = 1
Internal rollbacks                   = 0
Internal rollbacks due to deadlock  = 0
Binds/precompiles attempted         = 0

```

GET SNAPSHOT

```
Rows deleted = 0
Rows inserted = 0
Rows updated = 0
Rows selected = 16
Rows read = 25
Rows written = 0
UOW log space used (Bytes) = 0
Previous UOW completion timestamp = 02-25-1999 13:31:34.434114
Elapsed time of last completed uow (sec.ms) = 0.919533380
UOW start timestamp = 02-25-1999 13:33:41.392167
UOW stop timestamp =
UOW completion status =
Open remote cursors = 0
Open remote cursors with blocking = 0
Rejected Block Remote Cursor requests = 0
Accepted Block Remote Cursor requests = 2
Open local cursors = 0
Open local cursors with blocking = 0
Total User CPU Time used by agent (s) = 0.100000
Total System CPU Time used by agent (s) = 0.020000
Package cache lookups = 2
Package cache inserts = 1
Application section lookups = 10
Application section inserts = 1
Catalog cache lookups = 1
Catalog cache inserts = 1
Catalog cache overflows = 0
Catalog cache heap full = 0
Most recent operation = Select
Most recent operation start timestamp = 02-25-1999 13:33:41.394260
Most recent operation stop timestamp = 02-25-1999 13:33:41.446740
Agents associated with the application = 1
Number of hash joins = 0
Number of hash loops = 0
Number of hash join overflows = 0
Number of small hash join overflows = 0
Statement type = Dynamic SQL Statement
Statement = Select
Section number = 201
Application creator = NULLID
Package name = SQLC28A4
Cursor name = SQLCUR201
Statement node number = 0
Statement start timestamp = 02-25-1999 13:33:41.394260
Statement stop timestamp = 02-25-1999 13:33:41.446740
Elapsed time of last completed stmt(sec.ms) = 0.000000
Total user CPU time = 0.000000
Total system CPU time = 0.000000
SQL compiler cost estimate in timerons = 30
SQL compiler cardinality estimate = 47
Degree of parallelism requested = 1
Number of agents working on statement = 1
Number of subagents created for statement = 1
Statement sorts = 0
Total sort time = 0
```

```

Sort overflows                = 0
Rows read                      = 8
Rows written                   = 0
Rows deleted                   = 0
Rows updated                   = 0
Rows inserted                  = 0
Rows fetched                   = 0
Number of subsections         = 0
Dynamic SQL statement text:
select * from org

```

以下に示すのは、(DCS アプリケーション ID、DCS アプリケーション・ハンドル、全 DCS アプリケーション、または特定のデータベース上の全 DCS アプリケーションのいずれかを指定して) DCS アプリケーション情報を要求した結果表示された一般的な出力です。

```

          DCS Application Snapshot
Client application ID          = 09151251.04D6.980521202839
Sequence number                = 0001
  Authorization ID            = NEWTON
Application name               = db2bp
  Application handle          = 0
  Application status          = waiting for request
  Status change time         = 05-21-1998 16:35:27.670354
  Client DB alias            = MVSDB
  Client node                 = antman
  Client release level       = SQL05020
  Client platform            = AIX
  Client protocol            = TCP/IP
  Client codepage            = 819
  Process ID of client application = 35754
  Client login ID            = user1
  Host application ID        = G9151251.G4D7.980521202840
  Sequence number            = 0000
  Host DB name                = GILROY
  Host release level         = DSN05011
  Host CCSID                  = 500
Outbound communication address = 9.21.21.92 5021
Outbound communication protocol = TCP/IP
Inbound communication address  = 9.31.12.34 334
First database connect timestamp = 05-21-1998 16:28:39.517919
Time spent on gateway processing = 0.334215
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         = 392
Outbound bytes sent            = 136
Outbound bytes received        = 178
Inbound bytes sent             = 190
Number of open cursors         = 0
Application idle time          = 53 seconds

```

GET SNAPSHOT

```
UOW completion status = Committed - Commit Statement
Previous UOW completion timestamp =
UOW start timestamp = 05-21-1998 16:35:27.252375
UOW stop timestamp = 05-21-1998 16:35:27.670290
Inbound bytes received for UOW = 180
Outbound bytes sent for UOW = 136
Outbound bytes received for UOW = 178
Inbound bytes sent for UOW = 190
Most recent operation = Static Commit
Most recent operation start timestamp = 05-21-1998 16:35:27.284183
Most recent operation stop timestamp = 05-21-1998 16:35:27.670290
Statement = Static Commit
Section number = 0
Application creator = NULLID
Package name = SQLC28A0
SQL compiler cost estimate in timerons = 0
SQL compiler cardinality estimate = 0
Statement start timestamp = 05-21-1998 16:35:27.284183
Statement stop timestamp = 05-21-1998 16:35:27.670290
Rows fetched = 0
Time spent on gateway processing = 0.333740
Inbound bytes received for statement = 0
Outbound bytes sent for statement = 10
Outbound bytes received for statement = 54
Inbound bytes sent for statement = 0
```

以下に示すのは、バッファ・プール情報を要求した結果表示された一般的な出力です。

```
Bufferpool Snapshot
Bufferpool name = IBMDEFAULTBP
Database name = SAMPLE
Database path = /home/user1/user1/...
               NODE0000/SQL00011/
Input database alias = SAMPLE
Buffer pool data logical reads = 32
Buffer pool data physical reads = 13
Buffer pool data writes = 0
Buffer pool index logical reads = 55
Buffer pool index physical reads = 23
Total buffer pool read time (ms) = 364
Total buffer pool write time (ms) = 0
Database files closed = 0
Asynchronous pool data page reads = 0
Asynchronous pool data page writes = 0
Buffer pool index writes = 0
Asynchronous pool index page reads = 0
Asynchronous pool index page writes = 0
Total elapsed asynchronous read time = 0
Total elapsed asynchronous write time = 0
Asynchronous read requests = 0
Direct reads = 34
Direct writes = 0
Direct read requests = 4
Direct write requests = 0
```

```

Direct reads elapsed time (ms)      = 1
Direct write elapsed time (ms)     = 0
Data pages copied to extended storage = 0
Index pages copied to extended storage = 0
Data pages copied from extended storage = 0
Index pages copied from extended storage = 0

```

以下に示すのは、表情報を要求した結果表示された一般的な出力です。

```

                Table Snapshot
First database connect timestamp    = 12-27-1999 23:28:58.699766
Last reset timestamp                =
Snapshot timestamp                  = 12-27-1999 23:42:02.881998
Database name                       = FOO
Database path                       = /home/mckeough/mckeough/NODE0000/SQL00001/
Input database alias                = FOO
Number of accessed tables           = 3
Table List
Table Schema                       = MCKEOUGH
Table Name                         = FOO
Table Type                          = User
Rows Read                          = 0
Rows Written                        = 4
Overflows                          = 0
Page Reorgs                        = 0
Table Schema                       = SYSIBM
Table Name                         = SYSCOLUMNS
Table Type                          = Catalog
Rows Read                          = 0
Rows Written                        = 1
Overflows                          = 0
Page Reorgs                        = 0
Table Schema                       = SYSIBM
Table Name                         = SYSUSERAUTH
Table Type                          = Catalog
Rows Read                          = 0
Rows Written                        = 1
Overflows                          = 0
Page Reorgs                        = 0

```

以下に示すのは、表スペース情報を要求した結果表示された一般的な出力です。

```

                Tablespace Snapshot
First database connect timestamp    = 04-04-1997 14:29:55.197659
Last reset timestamp                =
Snapshot timestamp                  = 04-04-1997 14:32:14.151875
Database name                       = SAMPLE
Database path                       = /home/user1/user1/NODE0000/SQL00011/
Input database alias                = SAMPLE
Number of accessed tablespaces      = 3
Tablespace name                     = SYSCATSPACE
Data pages copied to extended storage = 0
Index pages copied to extended storage = 0
Data pages copied from extended storage = 0

```

GET SNAPSHOT

```
Index pages copied from extended storage = 0
  Buffer pool data logical reads         = 26
  Buffer pool data physical reads        = 11
Asynchronous pool data page reads       = 0
Buffer pool data writes                  = 0
Asynchronous pool data page writes      = 0
Buffer pool index logical reads         = 55
Buffer pool index physical reads        = 23
Asynchronous pool index page reads      = 0
Buffer pool index writes                = 0
Asynchronous pool index page writes     = 0
  Total buffer pool read time (ms)      = 342
  Total buffer pool write time (ms)     = 0
Total elapsed asynchronous read time    = 0
Total elapsed asynchronous write time   = 0
Asynchronous read requests              = 0
Direct reads                            = 34
Direct writes                           = 0
Direct read requests                     = 4
Direct write requests                   = 0
Direct reads elapsed time (ms)          = 1
Direct write elapsed time (ms)          = 0
  Number of files closed                 = 0
Tablespace name                          = TEMPSPACE1
Data pages copied to extended storage    = 0
Index pages copied to extended storage   = 0
Data pages copied from extended storage  = 0
Index pages copied from extended storage = 0
  Buffer pool data logical reads         = 0
  Buffer pool data physical reads        = 0
  Asynchronous pool data page reads     = 0
  Buffer pool data writes                = 0
  Asynchronous pool data page writes    = 0
  Buffer pool index logical reads        = 0
  Buffer pool index physical reads       = 0
  Asynchronous pool index page reads    = 0
  Buffer pool index writes               = 0
  Asynchronous pool index page writes   = 0
  Total buffer pool read time (ms)      = 0
  Total buffer pool write time (ms)     = 0
  Total elapsed asynchronous read time  = 0
  Total elapsed asynchronous write time = 0
  Asynchronous read requests            = 0
  Direct reads                          = 0
  Direct writes                         = 0
  Direct read requests                   = 0
  Direct write requests                  = 0
  Direct reads elapsed time (ms)        = 0
  Direct write elapsed time (ms)        = 0
    Number of files closed               = 0
Tablespace name                          = USERSPACE1
Data pages copied to extended storage    = 0
Index pages copied to extended storage   = 0
Data pages copied from extended storage  = 0
Index pages copied from extended storage = 0
```



```

    Buffer pool data logical reads      = 6
    Buffer pool data physical reads    = 2
    Asynchronous pool data page reads = 0
    Buffer pool data writes             = 0
    Asynchronous pool data page writes = 0
    Buffer pool index logical reads     = 0
    Buffer pool index physical reads    = 0
    Asynchronous pool index page reads = 0
    Buffer pool index writes           = 0
    Asynchronous pool index page writes = 0
    Total buffer pool read time (ms)   = 22
    Total buffer pool write time (ms)  = 0
    Total elapsed asynchronous read time = 0
    Total elapsed asynchronous write time = 0
    Asynchronous read requests        = 0
    Direct reads                      = 0
    Direct writes                     = 0
    Direct read requests              = 0
    Direct write requests             = 0
    Direct reads elapsed time (ms)    = 0
    Direct write elapsed time (ms)    = 0
    Number of files closed            = 0

```

以下に示すのは、ロック情報を要求した結果表示された一般的な出力です。

```

Database Lock Snapshot
Database name          = F00
Database path         = /home/newton/newton/NODE0000/SQL00001/
Input database alias  = F00
Locks held            = 22
Applications currently connected = 1
Agents currently waiting on locks = 0
Snapshot timestamp    = 12-27-1999 23:41:59.166963
Application handle    = 2
Application ID        = *LOCAL.newton.991228042858
Sequence number       = 0001
Application name      = db2bp
Authorization ID      = NEWTON
Application status    = UOW Waiting
Status change time   =
Application code page = 819
Locks held           = 5
Total wait time (ms) = 0
Lock Object Name     = 33
Node number lock is held at = 0
Object Type          = Table
Tablespace Name      = SYSCATSPACE
Table Schema         = SYSIBM
Table Name           = SYSUSERAUTH
Mode                 = IX
Status               = Granted
Lock Escalation      = NO
Lock Object Name     = 558
Node number lock is held at = 0
Object Type          = Row
Tablespace Name      = SYSCATSPACE
Table Schema         = SYSIBM
Table Name           = SYSTABAUTH
Mode                 = W
Status               = Granted

```

GET SNAPSHOT

```
Lock Escalation           = NO
Lock Object Name          = 13
Node number lock is held at = 0
Object Type               = Table
Tablespace Name          = SYSCATSPACE
Table Schema              = SYSIBM
Table Name                = SYSTABAUTH
Mode                      = IX
Status                    = Granted
Lock Escalation           = NO
Lock Object Name          = 3078
Node number lock is held at = 0
Object Type               = Row
Tablespace Name          = SYSCATSPACE
Table Schema              = SYSIBM
Table Name                = SYSTABLES
Mode                      = W
Status                    = Granted
Lock Escalation           = NO
Lock Object Name          = 2
Node number lock is held at = 0
Object Type               = Table
Tablespace Name          = USERSPACE1
Table Schema              = NEWTON
Table Name                = FOO
Mode                      = Z
Status                    = Granted
Lock Escalation           = NO
```

以下に示すのは、動的 SQL 情報を要求した結果表示された一般的な出力です。

```
Dynamic SQL Snapshot Result
Database name              = SAMPLE
Database path              = /home/smith/smith/NODE0000/SQL00001/
Number of executions      = 2
Number of compilations    = 1
Worst preparation time (ms) = 126
Best preparation time (ms) = 126
Rows deleted               = 0
Rows inserted              = 0
Rows read                  = 24
Rows updated               = 0
Rows written               = 0
Statement sorts           = 0
Total execution time (sec.ms) = 0.060226
Total system cpu time (sec.ms) = 0
Total user cpu time (sec.ms) = 0
Statement text             = select * from org
```

使用上の注意

リモート・インスタンス (または別のローカル・インスタンス) からスナップショットを獲得するには、まず最初にそのインスタンスと接続することが必要です。別のインスタンスに常駐するデータベースの別名が指定されている場合には、エラー・メッセージが返されます。

統計を獲得するには、データベース・システム・モニターをオンにしない場合があります。

以下の条件のいずれかが真の場合には、表情報を要求してもデータは返ってきません。

- TABLE 記録スイッチがオフである。
- スイッチをオンにして以来、アクセスのあった表がない。
- 最後に RESET MONITOR コマンドを発行して以来、アクセスのあった表がない。

以下も参照

310ページの『GET MONITOR SWITCHES』

364ページの『LIST APPLICATIONS』

535ページの『RESET MONITOR』

HELP

HELP

ユーザーはインフォメーション・センターからヘルプを呼び出すことができません。

このコマンドは UNIX ベースのシステムでは使用できません。

許可

なし

必須接続

なし

コマンド構文

→ HELP character-string →

コマンド・パラメーター

HELP character-string

SQL コマンドや DB2 コマンド、またはインフォメーション・センターにリストされているその他の項目。

例

以下に示すのは、HELP コマンドの例です。

- db2 help
このコマンドは、DB2 インフォメーション・センターをオープンします。インフォメーション・センターでは、DB2 に関する情報が、作業、解説書、ブックなどに分類されています。これは、パラメーターを指定しないで **db2ic** コマンドを呼び出すのと同じです。
- db2 help drop
このコマンドは、Web ブラウザーをオープンし、SQL DROP ステートメントに関する情報を表示します。これは、コマンド **db2ic -j drop** を呼び出すのと同じです。**db2ic** コマンドは、DROP と呼ばれるステートメントまたはコマンドを、最初に *SQL 解説書* で検索し、次に *コマンド解説書* で検索して、最初に検出された情報を表示します。
- db2 help 'drop database'

このコマンドは、より洗練された検索を開始し、DROP DATABASE コマンドに関する情報を表示します。

使用上の注意

インフォメーション・センターがユーザーのシステムにインストールされている必要があります。DB2 ライブラリーの HTML ブックは、`%sqllib%doc%html` サブディレクトリーになければなりません。

コマンド行プロセッサは、コマンドが成功したかどうかを知ることができないため、エラー状態を報告できません。

外部ファイルのデータを、サポートされているファイル形式で表、階層、または視点に挿入します。より高速な代替方法は 411 ページの『LOAD』です。しかしロード・ユーティリティーでは、階層レベルのデータのロードはサポートされていません。

許可

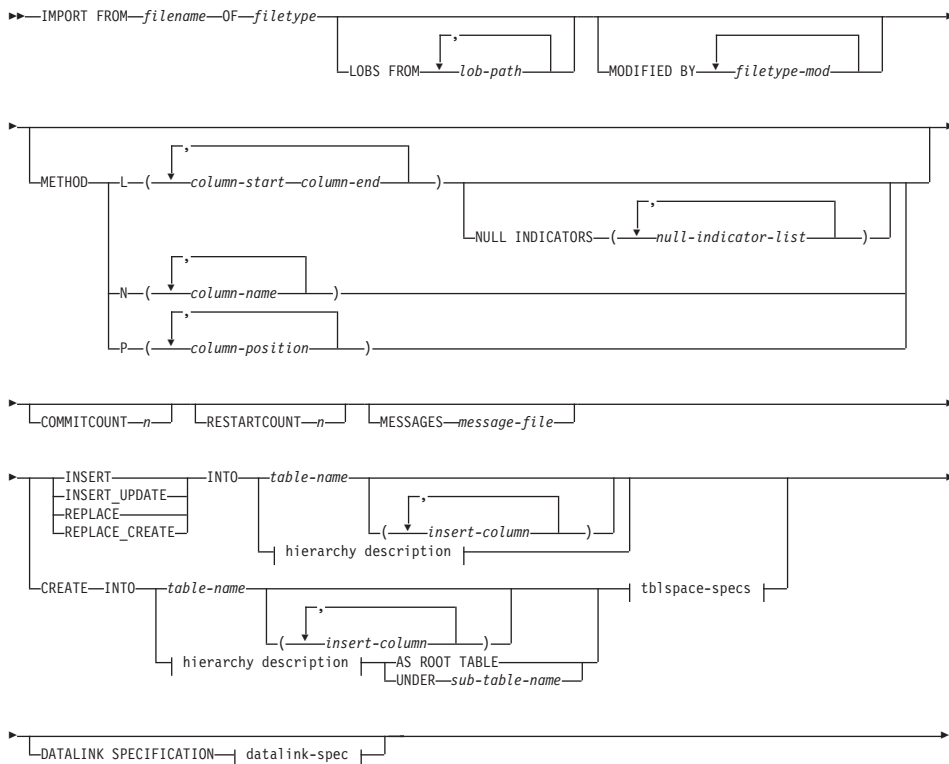
- INSERT オプションを使用して IMPORT コマンドを実行する場合、以下のどれかが必要です。
 - *sysadm*
 - *dbadm*
 - 関係する表または視点のそれぞれに対する CONTROL 特権
 - 関係する表または視点のそれぞれに対する INSERT および SELECT 特権
- INSERT_UPDATE、REPLACE、または REPLACE_CREATE オプションを使用して既存の表に IMPORT コマンドを実行する場合は、以下のどれかが必要です。
 - *sysadm*
 - *dbadm*
 - 表または視点に対する CONTROL 特権
- CREATE または REPLACE_CREATE オプションを使って、存在しない表または階層に IMPORT するには、以下のいずれかが必要です。
 - *sysadm*
 - *dbadm*
 - データベースに対する CREATETAB 権限、および以下のどちらか。
 - 表のスキーマ名が存在しない場合は、データベースに対する IMPLICIT_SCHEMA 権限
 - 表のスキーマ名が存在する場合は、スキーマに対する CREATEIN 特権
 - 階層全体に対して REPLACE_CREATE オプションが使用されている場合は、階層内のすべての副表に対する CONTROL 特権
- REPLACE オプションを使用して既存の階層に IMPORT するには、以下のいずれかが必要です。
 - *sysadm*
 - *dbadm*
 - 階層内のすべての副表に対する CONTROL 特権

必須接続

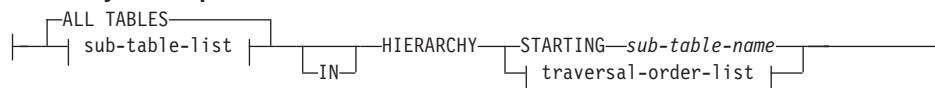
データベース。暗黙接続が可能な場合には、省略時データベースへの接続が確立されます。

IMPORT

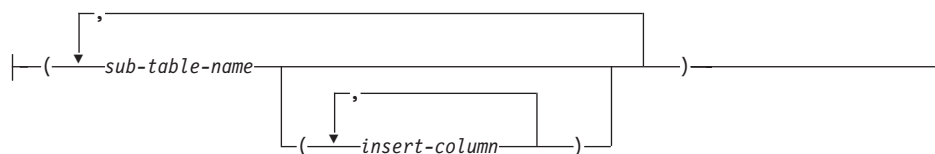
コマンド構文



hierarchy description:

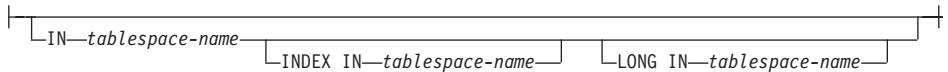
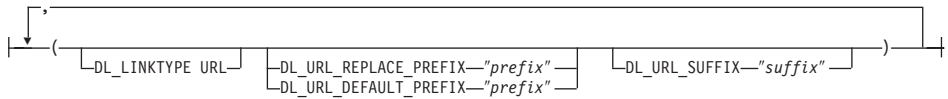


sub-table-list:



traversal-order-list:



tblspace-specs:**datalink-spec:****コマンド・パラメーター****ALL TABLES**

暗黙のキーワード (階層のみ)。階層をインポートする場合、走査順序で指定されるすべての表をインポートすることが省略時値です。

AS ROOT TABLE

1 つまたは複数の副表を、独立した表階層として作成します。

COMMITCOUNT n

n 個のレコードがインポートされるたびに COMMIT を実行します。

CREATE

表定義と行の内容を作成します。DB2 の表、副表、または階層からエクスポートされたデータの場合、索引も作成されます。このオプションが階層に対するものである場合で、DB2 からデータがエクスポートされた場合には、タイプ階層も作成されます。このオプションは、IXF ファイルの場合にのみ使用することができます。

注: データが MVS ホスト・データベースからエクスポートされたもので、ページ・サイズで計算した長さが 254 より少ない LONGVAR フィールドを含んでいる場合、CREATE は行が長過ぎるために失敗します。この場合、その表は手動で作成します。そして、IMPORT に INSERT を指定して呼び出すか、または LOAD コマンドを使用してください。

DATALINK SPECIFICATION

各 DATALINK 列ごとに、それぞれ 1 つの列指定を括弧で囲んで指定できます。各列指定は、1 つまたは複数の DL_LINKTYPE、接頭部、および DL_URL_SUFFIX 指定で構成されます。接頭部指定は、DL_URL_REPLACE_PREFIX または DL_URL_DEFAULT_PREFIX のどちらかになります。

DATALINK 列指定の数は、表で定義されている DATALINK の数と同じだけ定義できます。指定の順序は、挿入列 リストの中での DATALINK 列の順序、または表定義内での順序 (挿入列 リストが指定されていない場合) に従います。

DL_LINKTYPE

指定した場合は、列定義の LINKTYPE に一致していなければなりません。そうすることによって、列定義に LINKTYPE URL が指定されている場合に DL_LINKTYPE URL が受け入れ可能になります。

DL_URL_DEFAULT_PREFIX "prefix"

これを指定すると、同じ列内のすべての DATALINK 値の省略時接頭部になります。ここでいう接頭部とは、URL 指定の「スキーム・ホスト・ポート」部分のことです。(分散ファイル・システム (DFS) の場合、接頭部とは URL 指定の「スキーム・セル名とファイル・スペースの接合」部分のことです。)

接頭部の例

```
"http://server"  
"file://server"  
"file:"  
"http://server:80"  
"dfs://.../cellname/fs"
```

列のデータの中に接頭部がない場合、DL_URL_DEFAULT_PREFIX で省略時の接頭部が指定されているなら、列の値の接頭部としてその省略時接頭部が付けられます (NULL でない場合)。

たとえば、DL_URL_DEFAULT_PREFIX で省略時接頭部が "http://toronto" として指定されている場合、

- 列入力値 "/x/y/z" は "http://toronto/x/y/z" として保管されます。
- 列入力値 "http://coyote/a/b/c" は "http://coyote/a/b/c" として保管されます。
- 列入力値 NULL は NULL として保管されます。

DL_URL_REPLACE_PREFIX "prefix"

この文節は、それ以前にエクスポート・ユーティリティーによって生成されたデータをロードまたはインポートする際に、ユーザーがデータに含まれるホスト名を別のホスト名に一括置換したい場合に便利です。指定する場合には、それがすべての非 NULL 列値の接頭部になります。列値にすでに接頭部があるなら、それは置き換えられます。列値に接頭部がないなら、DL_URL_REPLACE_PREFIX で指定される接頭部

がその列値の接頭部になります。分散ファイル・システム (DFS) の場合、接頭部とは URL 指定の「スキーム・セル名とファイル・スペースの接合」部分のことです。

たとえば、DL_URL_REPLACE_PREFIX で接頭部が "http://toronto" として指定されている場合、

- 列入力値 "/x/y/z" は "http://toronto/x/y/z" として保管されます。
- 列入力値 "http://coyote/a/b/c" は "http://toronto/a/b/c" として保管されます。"coyote" は "toronto" に置き換えられます。
- 列入力値 NULL は NULL として保管されます。

DL_URL_SUFFIX "suffix"

これを指定すると、それはその列のすべての非 NULL 列値に付加されます。これは実際には、DATALINK 値の URL 部分の「パス」構成要素に付加されます。

FROM filename

インポートするデータを含むファイルを指定します。パスを省略すると、現行の作業ディレクトリーが使用されます。

HIERARCHY

インポートする階層データを指定します。

IN tablespace-name

表を作成する表スペースを指定します。表スペースは存在している必要があります、REGULAR 表スペースでなければなりません。他の表スペースを指定しない場合、すべての表パーツはこの表スペースに保管されます。この文節を指定しない場合、表は許可 ID によって作成された表スペース中に作成されます。何も検出されない場合、その表は省略時表スペースの USERSPACE1 に入れます。USERSPACE1 が消去されていた場合、表作成は失敗します。

INDEX IN tablespace-name

表示に対して索引を作成する場合、索引を作成する表スペースを指定します。このオプションは、IN 文節で指定した 1 次表スペースが DMS 表スペースである場合のみ使用できます。指定した表スペースは存在している必要があります、かつ REGULAR DMS 表スペースでなければなりません。

注: どの表スペースに索引を含めるかは、表を作成するときのみ指定できます。

insert-column

データの挿入先となる表または視点内の列名を指定します。

IMPORT

INSERT

既存の表データを変更することなく、インポートしたデータを表に追加します。

INSERT_UPDATE

インポートしたデータ行をターゲット表に追加するか、または 1 次キーが一致するものがあればターゲット表の既存行を更新します。

INTO table-name

データのインポート先となるデータベース表を指定します。この表として、システム表、宣言一時表、または要約表は指定できません。

下位のサーバーの場合を除き、INSERT、INSERT_UPDATE、および REPLACE オプションには、完全修飾または非修飾の表名を使わなければならないようなときでも、別名を指定することができます。修飾子付き表名は、*schema.tablename* の形式です。 *schema* には、表作成時のユーザー名が入ります。

LOBS FROM lob-path

LOB ファイルを保管するパス (複数可) を指定します。LOB データ・ファイルの名前は、メイン・データ・ファイル (ASC、DEL、または IXF) の、LOB 列にロードされる列内に保管されます。 *lobsinfile* 修飾子が指定されていない場合、このオプションは無視されます。

LONG IN tablespace-name

ロング列の値 (LONG VARCHAR、LONG VARGRAPHIC、LOB データ・タイプ、またはソース・タイプとしてこれらが指定されている特殊タイプ) を保管する表スペースを指定します。このオプションは、IN 文節で指定した 1 次表スペースが DMS 表スペースである場合のみ使用できます。指定した表スペースは存在している必要があり、LONG DMS 表スペースでなければなりません。

MESSAGES message-file

インポート操作中に生じ得る警告およびエラー・メッセージの宛先を指定します。宛先ファイルがすでに存在している場合、インポート・ユーティリティーは情報を追加します。このファイルへの完全パスが指定されていない場合、このユーティリティーは現行のディレクトリーおよび省略時のドライブを宛先として使用します。 *message-file* を省略すると、メッセージは標準出力に書き込まれます。

METHOD

L データのインポートを開始する列および終了する列の番号を指定します。列の番号は、データの行の先頭からのバイト単位のオフセットです。この番号は 1 から始まります。

注: このメソッドは、ASC ファイルの場合にのみ使用することができ、そのファイル・タイプに対してのみ有効なオプションです。

N インポートする列の名前を指定します。

注: この方式は、IXF ファイルの場合にのみ使用することができます。

P インポートする入力データ・フィールドの索引 (1 から始まる) を指定します。

注: この方式は、IXF または DEL ファイルの場合にのみ使用でき、DEL ファイル・タイプに対してのみ有効なオプションです。

MODIFIED BY filetype-mod

追加オプションを指定します (348ページの表6 を参照)。

NULL INDICATORS null-indicator-list

このオプションは、METHOD L パラメーターを指定した場合だけ使用できます (つまり、入力ファイルが ASC ファイルの場合)。ヌル標識リストは、コンマで区切られた正の整数のリストで、各ヌル標識フィールドの列の番号を指定します。列の番号は、データの行の先頭からのバイト単位の、各ヌル標識フィールドのオフセットです。ヌル標識リストには、METHOD L パラメーターで定義された各データ・フィールドに対する 1 つの項目がなければなりません。列の番号がゼロであることは、対応するデータ・フィールドが必ずデータを含んでいることを示します。

ヌル標識列中の Y の値は、その列データがヌル値であることを指定します。ヌル標識列に Y 以外 の文字を指定した場合は、列データがヌル値ではなく、METHOD L オプションで指定された列データがインポートされることを指定することになります。

ヌル標識文字は MODIFIED BY オプションを使用して変更できます (348ページの表6 の nullindchar 修飾子の説明を参照)。

OF filetype

入力ファイル内のデータの形式を指定します。

- ASC (区切りなし ASCII 形式)
- DEL (区切り文字付き ASCII 形式)。さまざまなデータベース・マネージャーやファイル・マネージャーで使用します。
- WSF (ワークシート形式)。以下のプログラムで使用します。

IMPORT

- ロータス 1-2-3
- Lotus Symphony
- IXF (統合交換フォーマット、PC バージョン)。同一のあるいは別の DB2 表からエクスポートされたことを意味します。IXF ファイルには、表定義および既存の索引定義も含まれます。ただし、SELECT ステートメントに列が指定されている場合は除きます。

ファイル形式の詳細については、データ移動ユーティリティー 手引きおよび解説書の付録『Export/Import/Load ユーティリティー・ファイル形式』を参照してください。

REPLACE

データ・オブジェクトを切り捨てることによって表内の既存のデータすべてを削除してから、インポートしたデータを挿入します。表定義および索引定義は変更されません。表がない場合は、このオプションを使用できません。DATALINK 列を含む表では無効です。階層間でデータを移動する際にこのオプションを使用する場合は、階層全体に関係したデータだけが置き換えられます。副表は置き換えられません。

REPLACE_CREATE

表がすでにある場合には、データ・オブジェクトを切り捨てることによって表内の既存のデータすべてを削除し、表定義や索引定義は変えることなく、インポートしたデータを挿入します。

表がまだない場合には、行の内容だけでなく表定義と索引定義も作成します。

このオプションは、IXF ファイルの場合にのみ使用することができます。DATALINK 列を含む表では無効です。階層間でデータを移動する際にこのオプションを使用する場合は、階層全体に関係したデータだけが置き換えられます。副表は置き換えられません。

RESTARTCOUNT *n*

$n + 1$ の位置のレコードからインポート操作を開始するよう指定します。最初の n 個のレコードはスキップされます。

STARTING *sub-table-name*

階層専用キーワード。 *sub-table-name* から始まる省略時順を要求します。PC/IXF ファイルの場合、省略時順は入力ファイルに保管されている順です。PC/IXF ファイル形式の場合、省略時順は有効な唯一の順序です。

sub-table-list

タイプ表で INSERT または INSERT_UPDATE オプションを指定した場合、データのインポート先副表を指定するために副表名のリストが使われます。

traversal-order-list

タイプ表で INSERT、INSERT_UPDATE、または REPLACE オプションを指定した場合、インポートする階層内の副表の走査順序を指定するために副表名のリストを使います。

UNDER sub-table-name

1 つまたは複数の副表を作成する場合に親表を指定します。

例**例 1**

次に示すのは myfile.ixf から STAFF 表に情報をインポートする方法の一例です。

```
db2 import from myfile.ixf of ixf messages msg.txt insert into staff
SQL3150N The H record in the PC/IXF file has product "DB2 01.00", date
"19970220", and time "140848".
SQL3153N The T record in the PC/IXF file has name "myfile", qualifier
" ", and source " ".
SQL3109N The utility is beginning to load data from file "myfile".
SQL3110N The utility has completed processing. "58" rows were read from the
input file.
SQL3221W ...Begin COMMIT WORK. Input Record Count = "58".
SQL3222W ...COMMIT of any database changes was successful.
SQL3149N "58" rows were processed from the input file. "58" rows were
successfully inserted into the table. "0" rows were rejected.
```

例 2

下記に示す例は、DEL 形式のデータが含まれている入力ファイル delfile1 から、表 MOVIE TABLE をインポートする方法を示す例です。

```
db2 import from delfile1 of del
modified by dl|del|
insert into movietable (actorname, description, url_making_of, url_movie)
datalink specification (dl_url_default_prefix "http://narang"),
(dl_url_replace_prefix "http://bomdel" dl_url_suffix ".mpeg")
```

注:

1. この表には下記の 4 つの列が含まれています。

actorname	VARCHAR(n)
description	VARCHAR(m)
url_making_of	DATALINK (with LINKTYPE URL)
url_movie	DATALINK (with LINKTYPE URL)

2. 入力ファイルの中の DATALINK データのサブフィールド区切り文字は、縦線 (|) 文字です。
3. url_making_of の列値に接頭部文字列が含まれていないなら、"http://narang" が使用されます。
4. url_movie の非 NULL 列値には、接頭部として "http://bomdel" が付けられます。既存の値は置き換えられます。
5. url_movie の非 NULL 列値のパスには、".mpeg" が付加されます。たとえば、url_movie の列値が "http://server1/x/y/z" なら、それは "http://bomdel/x/y/z.mpeg" として保管されます。その値が "/x/y/z" なら "http://bomdel/x/y/z.mpeg" として保管されます。

例 3 (識別列がある表へのインポート)

TABLE1 には以下の 4 つの列があります。

- C1 VARCHAR(30)
- C2 INT GENERATED BY DEFAULT AS IDENTITY
- C3 DECIMAL(7,2)
- C4 CHAR(1)

TABLE2 は TABLE1 と同じですが、C2 が GENERATED ALWAYS 識別列である点が異なります。

DATAFILE1 のデータ・レコード (DEL 形式)

```
"Liszt"
"Hummel",,187.43, H
"Grieg",100, 66.34, G
"Satie",101, 818.23, I
```

DATAFILE2 のデータ・レコード (DEL 形式)

```
"Liszt", 74.49, A
"Hummel", 0.01, H
"Grieg", 66.34, G
"Satie", 818.23, I
```

以下のコマンドは、DATAFILE1 で行 1 および 2 への識別値の提供がないので、それらの行のための識別値を生成します。ただし、行 3 および 4 は、それぞれユーザー提供の識別値 100 と 101 が割り当てられます。

```
db2 import from datafile1.del of del replace into table1
```

DATAFILE1 を TABLE1 にインポートしてすべての行に対する識別値を生成するには、以下のコマンドのいずれかを発行します。


```
db2 import from datafile1.del of del method P(1, 3, 4) replace into table1 (c1, c3, c4)
db2 import from datafile1.del of del modified by identityignore replace into table1
```

DATAFILE2 を TABLE1 にインポートして各行に対する識別値を生成するには、以下のコマンドのいずれかを発行します。

```
db2 import from datafile2.del of del replace into table1 (c1, c3, c4)
db2 import from datafile2.del of del modified by identitymissing replace into table1
```

DATAFILE1 を TABLE2 に、識別関連のファイル・タイプ修飾子を使用せずにインポートした場合、行 1 と 2 は挿入されますが、行 3 と 4 は拒否されます。その理由は、それらが固有の非ヌル値を提供し、識別列が GENERATED ALWAYS であるからです。

使用上の注意

インポート操作を開始する前に、すべての表操作が完了し、すべてのロックが保留解除になっていることを確認してください。これは、WITH HOLD でオープンされた、すべてのカーソルをクローズした後で COMMIT または ROLLBACK を発行することによって行われます。

インポート・ユーティリティーは、SQL INSERT ステートメントを使ってターゲット表に行を追加します。ユーティリティーは入力ファイル中のデータの行ごとに 1 つの INSERT ステートメントを発行します。INSERT ステートメントが失敗すると、次のどちらかのアクションが起きます。

- 後続の INSERT ステートメントを正常に実行できると思われる場合、メッセージ・ファイルに警告メッセージが書き込まれ、処理は続行します。
- 後続の INSERT ステートメントが失敗しそうで、データベースの損傷の可能性がある場合、メッセージ・ファイルにエラー・メッセージが書き込まれ、処理は停止します。

ユーティリティーは、REPLACE または REPLACE_CREATE 操作中に、古い行が削除された後、自動 COMMIT を実行します。したがって、システムが失敗するか、またはアプリケーションが表オブジェクトの切り捨て後にデータベース・マネージャーに割り込む場合、古いデータはすべて失われます。これらのオプションを使用する前に、古いデータがもう必要ないことを必ず確認してください。

ログが CREATE、REPLACE、または REPLACE_CREATE 操作中にいっぱいになった場合、ユーティリティーは挿入されたレコード上で自動 COMMIT を実行します。システムが失敗した場合、またはアプリケーションが自動 COMMIT の後にデータベース・マネージャーに割り込んだ場合、部分データを持つ表がデータベースに残されます。REPLACE または

REPLACE_CREATE オプションを使用してインポート操作全体をやり直すか、または正常にインポートされる行の数に設定した RESTARTCOUNT パラメーターを指定して INSERT を使用してください。

省略時では、自動 COMMIT は INSERT または INSERT_UPDATE オプションには実行されません。しかし、COMMITCOUNT パラメーターがゼロでない場合は実行されます。フル・ログの結果は ROLLBACK です。

インポート・ユーティリティーが COMMIT を実行するときにはいつでも、2 つのメッセージがメッセージ・ファイルに書き込まれます。1 つはコミットされるレコードの数を示し、もう 1 つは正常に終了した COMMIT の後に書き込まれます。失敗の後にインポート操作を再始動する場合、最後に正常に終了した COMMIT から決定されたとおり、スキップするレコードの数を指定してください。

インポート・ユーティリティーは、小さい非互換性問題 (たとえば、文字データは埋め込みまたは切り捨てを使ってインポートでき、数値データは異なる数値データ型を使ってインポートできる) は受け入れますが、大きな非互換性に関する問題は受け入れません。

それ自体以外への依存があるオブジェクト表や、基本表に何らかの依存 (それ自体も含めて) があるオブジェクト視点を、REPLACE または REPLACE_CREATE することはできません。このような表または視点を置換するには、次のようにします。

1. 表が親であるすべての外部キーを除去します。
2. インポート・ユーティリティーを実行します。
3. 表を変更して外部キーを再作成します。

外部キーの再作成中にエラーが発生する場合、参照保全を保守するためにデータを変更してください。

参照制約および外部キー定義は、PC/IXF ファイルから表を作成する場合は保存されません。(基本キー定義は、データが前に SELECT * を使ってエクスポートされた場合、保存されます。)

リモート・データベースへのインポートには、入力データ・ファイルのコピー、出力メッセージ・ファイル、およびデータベースがサイズが大きくなる可能性に備えて、十分なディスク・スペースをサーバー上に確保することが必要です。

インポート操作がリモート・データベースに対して実行され、出力メッセージ・ファイルが非常に長い (60KB より長い) 場合、クライアント上でユーザーに戻されるメッセージ・ファイルがインポート操作中に欠落することがあります。メッセージ情報の最初の 30KB と最後の 30KB は、常に保存されます。

リモート・データベースへの PC/IXF ファイルのインポートは、PC/IXF ファイルがディスクットではなくハード・ディスク上にある場合、より早く行えます。

ASC、DEL、または WSF のファイル形式をインポートするためには、それ以前にデータベース表または階層がすでに存在していなければなりません。しかし、表がまだ存在していない場合でも、IMPORT CREATE または IMPORT REPLACE_CREATE を使えば、PC/IXF ファイルからデータをインポートする際に表が作成されます。タイプ表の場合、IMPORT CREATE によってタイプ階層と表階層も作成されます。

データ (階層データを含む) を別のデータベースに移動するには、PC/IXF インポートを使う必要があります。行区切り文字を含む文字データを区切り文字付き ASCII (DEL) ファイルにエクスポートし、テキスト転送プログラムにより処理を行うと (たとえば、OS/2 と AIX システム間で移動する)、行区切り文字を含むフィールドは長さが伸縮します。PC/IXF ファイル形式の指定では、OS/2 (IBM OS/2 拡張サービス、OS/2 拡張版、および DB2 (OS/2 版)) のデータベースと、DB2 (AIX 版) のデータベースの間で、エクスポート (OS/2 と AIX の間でファイルをバイナリー・コピー) とインポートによってデータを移行することが可能です。ソースと宛先のデータベースが両方とも同じクライアントからアクセス可能である場合、ファイルのコピーというステップは必要ありません。

ここでは、ASC ファイルおよび DEL ファイル内のデータは、インポートを実行するクライアント・アプリケーションのコード・ページのデータである想定します。別々のコード・ページにデータをインポートする場合、PC/IXF ファイル (異なるコード・ページへのインポートが考慮されたファイル) を使用することが推奨されています。PC/IXF ファイルとインポート・ユーティリティーが同一のコード・ページの場合、処理は通常のアプリケーションと同じようになります。両者が異なるコード・ページであっても、FORCEIN オプションが指定されている場合には、インポート・ユーティリティーにおいて、PC/IXF ファイル内のデータが、インポートを実行するアプリケーションと同一のコード・ページであるとみなされます。両者のコード・ページに共通の変換表がある場合にも、同じ処置になります。両者が異なるコード・ページであり、FORCEIN オプションも指定されていないが、変換表はある、という場合には、PC/IXF ファイル内のすべてのデータが、ファイル・コード・ページか

らアプリケーション・コード・ページへと変換されます。両者が異なるコード・ページにあり、FORCEIN オプションも指定されておらず、さらに変換もない場合には、インポート操作は失敗してしまいます。これは、DB2 (AIX 版) クライアントの PC/IXF ファイルにのみ適用されます。

列が 1012 個の限界に近づいている 8KB ページ上の表オブジェクトの場合、PC/IXF データ・ファイルをインポートすると、SQL ステートメントの最大サイズを超えているため、DB2 からエラーが戻されることがあります。このようになるのは、CHAR、VARCHAR、または CLOB 型の列の場合だけです。DEL または ASC ファイルのインポートでは、この制限は当てはまりません。PC/IXF ファイルを使って新しい表を作成している場合、別の方法として、71 ページの『db2look - DB2 統計および DDL 抽出ツール』を使ってソース表の DDL をダンプしてから、ステートメントを CLP から発行する、という方法があります。

DB2 コネクトを使うと、DB2 (OS/390 版)、DB2 (VM および VSE 版)、および DB2 (OS/400 版) などの DRDA サーバーにデータをインポートできます。PC/IXF インポート (INSERT オプション) だけがサポートされています。RESTARTCOUNT パラメーターもサポートされていますが、COMMITCOUNT パラメーターはサポートされていません。

タイプ表に対して CREATE オプションを使うと、PC/IXF ファイルの中で定義されているすべての副表が作成されます。副表定義は更新されません。タイプ表に対して CREATE 以外のオプションを使うと、走査順序リストによって、走査順序を指定できます。その場合、走査順序リストはエクスポート操作で使用されたものと一致していなければなりません。PC/IXF 形式の場合は、ターゲット副表名を指定して、ファイルに保管されている走査順序を使うだけです。

インポート・ユーティリティーを使って、前に PC/IXF ファイルにエクスポートされた表を回復できます。表は、エクスポート時の状態に戻ります。

データは、システム表、宣言一時表、または要約表にはインポートできません。

インポート・ユーティリティーを介して視点を作成することはできません。

DB2 では、複数パーツの PC/IXF ファイルを、OS/2 システムから AIX システムに各パーツをコピーすることによってインポートする操作がサポートされています。

Windows NT オペレーティング・システムの場合、

- 論理分割 PC/IXF ファイルのインポートはサポートされていません。
- 正しくない形式の PC/IXF または WSF ファイルのインポートはサポートされていません。

DB2 データ・リンク・マネージャーに関する考慮事項

DB2 インポート・ユーティリティーを実行する場合、その前に下記のことを実行しておいてください。

1. 適切なデータ・リンク・サーバーに、参照されるファイルをコピーする。
dlfm_import ユーティリティーを使って、**dlfm_export** ユーティリティーが生成するアーカイブから、ファイルを抽出することができます。
2. 必要な接頭部名を DB2 データ・リンク・マネージャーに登録する。さらに、必要ならデータベースに登録するなど、その他の管理作業も必要になることがあります。
3. 必要なら、SQL 表のためにエクスポートしたデータの (DATALINK 列の) URL に含まれているデータ・リンク・サーバー情報を更新します。(元の構成のデータ・リンク・サーバーがターゲットにおいても同じなら、そのデータ・リンク・サーバー名は更新する必要はありません。)
4. DB2 データ・リンク管理プログラム構成ファイルのターゲット構成で、データ・リンク・サーバーを定義する。

インポート・ユーティリティーがターゲット・システム上で実行される場合、DATALINK 列に関連したデータは SQL INSERT を使って、基礎となる DB2 表にロードされます (他の列の場合と同じ)。

挿入操作の間、DATALINK 列の処理では、ターゲット・データベースでの列指定に従って、該当するデータ・リンク・サーバー中のファイルがリンクされます。

入力ファイル内での DATALINK 情報の表示

DATALINK 情報が入力ファイル内でどのように表示されるかについては、434 ページを参照してください。

表 6. 有効なファイル・タイプ修飾子 (インポート)

修飾子	説明
すべてのファイル形式	
compound= <i>x</i>	<p><i>x</i> は 1~100 の数字です。データの挿入に非アトミック複合 SQL を使用し、1 回につき <i>x</i> ステートメントずつ挿入が試みられます。</p> <p>この修飾子が指定され、トランザクション・ログに十分な大きさが ない場合、インポート操作は失敗します。トランザクション・ログ は、COMMITCOUNT によって指定された行数か、または COMMITCOUNT が指定されていない場合はデータ・ファイルの行数 を入れる十分な大きさが必要です。したがって、トランザクショ ン・ログのオーバーフローを避けるために、COMMITCOUNT オプシ ョンを指定することをお勧めします。</p> <p>この修飾子は、INSERT_UPDATE モード、階層表、および修飾子 usedefaults、identitymissing、identityignore、 generatedmissing、generatedignore とは互換性がありません。</p>
generatedignore	<p>この修飾子は、インポート・ユーティリティーに、すべての生成列 のデータはデータ・ファイルに存在するが、それらを無視すべきこ とを知らせます。この結果として、生成列のすべての値は、このユ ーティリティーによって生成されます。この修飾子は、 generatedmissing 修飾子と共に使用することはできません。</p>
generatedmissing	<p>この修飾子を指定すると、ユーティリティーは、入力データ・ファ イルには生成列のデータが (NULL さえも) なく、したがって各行の 値が生成されると想定します。この修飾子は、generatedignore 修飾 子と共に使用することはできません。</p>
identityignore	<p>この修飾子は、インポート・ユーティリティーに、識別列のデータ はデータ・ファイルに存在するが、それらを無視すべきことを知ら せませす。この結果として、すべて識別値はこのユーティリティーに よって生成されます。この動作は、GENERATED ALWAYS および GENERATED BY DEFAULT 識別列のどちらの場合も同じです。つま り、GENERATED ALWAYS 列の場合、拒否される行はないという意 味です。この修飾子は、 identitymissing 修飾子と共に使用することはできません。</p>
identitymissing	<p>この修飾子を指定すると、ユーティリティーは、入力データ・ファ イルには識別列のデータが (NULL さえも) なく、したがって各行の 値が生成されると想定します。この動作は、GENERATED ALWAYS および GENERATED BY DEFAULT 識別列のどちらの場合も同じで す。この修飾子は、 identityignore 修飾子と共に使用することはできません。</p>
lobsinfile	<i>lob-path</i> には、LOB 値を含むファイルへのパスを指定します。
no_type_id	単一副表へのインポートの場合のみ有効。これを使う場合として典 型的な例は、正規の表からデータをエクスポートした後、この修飾 子を使ってインポート操作を呼び出してそのデータを単一の副表に 変換する場合です。

表 6. 有効なファイル・タイプ修飾子 (インポート) (続き)

修飾子	説明
nodefaults	<p>ターゲット表の列に対応するソース列が明示的に指定されていない場合、その表列がヌル値可なら、省略時値はロードされません。このオプションを指定せず、あるターゲット表列のためのソース列が明示的に指定されていない場合、以下のいずれかになります。</p> <ul style="list-style-type: none"> • 列に省略時値を指定できる場合、その省略時値がロードされます。 • 列がヌル値可で、省略時値がその列に指定できない場合、ヌル値がロードされます。 • 列がヌル値不可で、省略時値がその列に指定できない場合、エラーが戻され、ユーティリティーは処理を停止します。
usedefaults	<p>ターゲット表の列に対応するソース列を指定していても、1 つまたは複数の行インスタンスにデータが入っていない場合、省略時値がロードされます。欠落データの例は次のとおりです。</p> <ul style="list-style-type: none"> • DEL ファイルの場合、列に ".,;" が指定された場合 • ASC ファイルの場合、列のヌル標識が yes として設定された場合 • DEL/ASC/WSF ファイルの場合、列が不足している行、または元の指定では十分な長さのない行。 <p>このオプションを指定せず、行インスタンスのソース列にデータがない場合、以下のいずれかになります。</p> <ul style="list-style-type: none"> • 列がヌル値可なら、ヌル値がロードされます。 • 列がヌル値不可の場合、ユーティリティーは行を拒否します。
ASCII ファイル形式 (ASC/DEL)	

表 6. 有効なファイル・タイプ修飾子 (インポート) (続き)

修飾子	説明
dateformat="x"	<p>x は、ソース・ファイルの日付の形式です。^a 有効な日付要素は次のとおりです。</p> <p>YYYY - 年 (0000~9999 の範囲の 4 桁の数字) M - 月 (1~12 の範囲の 1 桁または 2 桁の数) MM - 月 (1~12 の範囲の 2 桁の数; M と同時には使用できない) D - 日 (1~31 の範囲の 1 桁または 2 桁の数) DD - 日 (1~31 の範囲の 2 桁の数; D と同時には使用できない) DDD - Day of the year (three digits ranging from 001 - 366; mutually exclusive with other day or month elements)</p> <p>省略時値の 1 が、指定されない各要素に割り当てられます。日付形式のいくつかの例を以下に示します。</p> <p>"D-M-YYYY" "MM.DD.YYYY" "YYYYDDD"</p>
implieddecimal	<p>暗黙の 10 進小数点の位置は、列定義によって指定されます。その位置が値の最後にあると想定されることはありません。たとえば、値 12345 が DECIMAL(8,2) 列にロードされる場合、123.45 としてであり、12345.00 ではありません。</p>
noeofchar	<p>オプションのファイル終了文字 x'1A' は、ファイル終了として識別されません。それが普通の文字であるかのようにして処理は継続されます。</p>

表 6. 有効なファイル・タイプ修飾子 (インポート) (続き)

修飾子	説明
timeformat="x"	<p>x は、ソース・ファイルの時刻の形式です。^a 有効な時刻要素は次のとおりです。</p> <p>H - Hour (one or two digits ranging from 0 - 12 for a 12 hour system, and 0 - 24 for a 24 hour system)</p> <p>HH - Hour (two digits ranging from 0 - 12 for a 12 hour system, and 0 - 24 for a 24 hour system; mutually exclusive with H)</p> <p>M - 分 (0~59 の範囲の 1 桁または 2 桁の数)</p> <p>MM - 分 (0~59 の範囲の 2 桁の数; M と同時には使用できない)</p> <p>S - 秒 (0~59 の範囲の 1 桁または 2 桁の数)</p> <p>SS - 秒 (0~59 の範囲の 2 桁の数; S と同時には使用できない)</p> <p>SSSSS - Second of the day after midnight (5 digits ranging from 00000 - 86399; mutually exclusive with other time elements)</p> <p>TT - 午前午後の標識 (AM または PM)</p> <p>省略時値の 0 が、指定されない各要素に割り当てられます。時刻形式のいくつかの例を以下に示します。</p> <p>"HH:MM:SS" "HH.MM TT" "SSSSS"</p>

表 6. 有効なファイル・タイプ修飾子 (インポート) (続き)

修飾子	説明
timestampformat="x"	<p>x は、ソース・ファイルのタイム・スタンプの形式です。^a 有効なタイム・スタンプ要素は次のとおりです。</p> <p>YYYY - 年 (0000~9999 の範囲の 4 桁の数字) M - 月 (1~12 の範囲の 1 桁または 2 桁の数) MM - 月 (1~12 の範囲の 2 桁の数; M (月) と同時には使用できない) D - 日 (1~31 の範囲の 1 桁または 2 桁の数) DD - 日 (1~31 の範囲の 2 桁の数; D と同時には使用できない) DDD - Day of the year (three digits ranging from 001 - 366; mutually exclusive with other day or month elements) H - Hour (one or two digits ranging from 0 - 12 for a 12 hour system, and 0 - 24 for a 24 hour system) HH - Hour (two digits ranging from 0 - 12 for a 12 hour system, and 0 - 24 for a 24 hour system; mutually exclusive with H) M - 分 (0~59 の範囲の 1 桁または 2 桁の数) MM - 分 (0~59 の範囲の 2 桁の数; M (分) と同時には使用できない) S - 秒 (0~59 の範囲の 1 桁または 2 桁の数) SS - 秒 (0~59 の範囲の 2 桁の数; S と同時には使用できない) SSSSS - Second of the day after midnight (5 digits ranging from 00000 - 86399; mutually exclusive with other time elements) UUUUUU - マイクロ秒 (000000 - 999999 の範囲の 6 桁の数) TT - 午前午後の標識 (AM または PM)</p> <p>YYYY、M、MM、D、DD、または DDD が指定されていない場合、省略時値として 1 が割り当てられます。他の要素が指定されていない場合には、省略時値として 0 が割り当てられます。以下に、タイム・スタンプ形式の例を示します。</p> <p>"YYYY/MM/DD HH:MM:SS.UUUUUU"</p> <p>次の例では、ユーザー定義の日時形式を含むデータを、schedule という表にインポートする方法を示します。</p> <pre>db2 import from delfile2 of del modified by timestampformat="yyyy.mm.dd hh:mm tt" insert into schedule</pre>
ASC (区切りなし ASCII) ファイル形式	

表 6. 有効なファイル・タイプ修飾子 (インポート) (続き)

修飾子	説明
nochecklengths	nochecklengths を指定した場合は、ソース・データの中にターゲット表列のサイズを超える列定義が含まれている場合であっても、各行のインポートが試みられます。コード・ページ変換によってソース・データが縮小されれば、そのような行であったとしても正常にインポートすることができます。たとえば、ソースに 4 バイトの EUC データがある場合、それがターゲットで 2 バイトの DBCS データに縮小されれば、必要なスペースは半分で済みます。列定義が一致しなくてもソース・データがきちんと収まるのが明らかな場合に、このオプションは特に便利です。
nullindchar=x	x は単一文字です。ヌル値を示す文字を x に変更します。 x の省略時値は \backslash です。 ^b EBCDIC データ・ファイルの場合、この修飾子は大文字小文字を区別しますが、英字の場合は区別しません。たとえば、ヌル標識文字を文字 N に指定した場合、 n もヌル標識として認識されます。
reclen=x	x は、32 767 以下の整数です。1 行につき x 文字ずつ読まれます。改行文字は行の終了にはなりません。
striptblanks	可変長フィールドにデータをインポートする場合に、後書きブランクをすべて切り捨てます。このオプションを指定しない場合、ブランク・スペースはそのまま保持されます。 次の例の場合、インポート・ユーティリティは、striptblanks によって後書きブランク・スペースを切り捨てます。 <pre>db2 import from myfile.asc of asc modified by striptblanks method 1 (1 10, 12 15) messages msgs.txt insert into staff</pre> このオプションは、striptnulls と一緒には指定できません。それらは相互に排他的なオプションです。 注: このオプションは、廃止された t オプション (後方互換性のためだけにサポートされる) に代わるものです。
striptnulls	可変長フィールドにデータをインポートする場合に、後書きヌル値 (0x00 文字) をすべて切り捨てます。このオプションを指定しない場合、ヌル値はそのまま保持されます。 このオプションは、striptblanks と一緒には指定できません。それらは相互に排他的なオプションです。 注: このオプションは、廃止された padwithzero オプション (後方互換性のためだけにサポートされる) に代わるものです。
DEL (区切り付き ASCII) ファイル形式	

表 6. 有効なファイル・タイプ修飾子 (インポート) (続き)

修飾子	説明
chardelx	<p>x は単一文字ストリング区切り文字です。省略時値は二重引用符 (") です。指定した文字は、文字ストリングを囲むために、二重引用符の代わりに使用されます。^{bc}</p> <p>単一引用符 (') も文字ストリング区切り文字として指定できます。次の例では、chardel'' の指定によって、すべての単一引用符 (') が文字ストリング区切り文字として解釈されます。</p> <pre>db2 "import from myfile.del of del modified by chardel'' method p (1, 4) insert into staff (id, years)"</pre>
coldelx	<p>x は単一文字列区切り文字です。省略時値はコンマ (,) です。指定した文字は、列の終わりを表すために、コンマの代わりに使用されます。^{bc}</p> <p>次の例では、coldel; の指定によって、すべてのセミコロン (;) が列区切りとして解釈されます。</p> <pre>db2 import from myfile.del of del modified by coldel; messages msgs.txt insert into staff</pre>
datesiso	日付形式。すべての日付データ値を ISO 形式でインポートします。
decplusblank	正符号文字。これによって正の 10 進値の先頭に正符号 (+) ではなく、ブランク・スペースが置かれます。省略時のアクションでは、正の 10 進数の前に正符号 (+) が付けられます。
decptx	<p>x は、小数点としてピリオドと置換される単一文字です。省略時値はピリオド (.) です。指定した文字は、小数点文字としてピリオドの代わりに使用されます。^{bc}</p> <p>次の例では、decpt; の指定によって、すべてのセミコロン (;) が小数点として解釈されます。</p> <pre>db2 "import from myfile.del of del modified by chardel' decpt; messages msgs.txt insert into staff"</pre>

表 6. 有効なファイル・タイプ修飾子 (インポート) (続き)

修飾子	説明
delprioritychar	<p>区切り文字の現在の省略時優先順位は、(1) レコード区切り文字、(2) 文字区切り文字、(3) 列区切り文字です。この修飾子を使用すると、区切り文字の優先順位が (1) 文字区切り文字、(2) レコード区切り文字、(3) 列区切り文字に戻り、以前の優先順位に依存している既存のアプリケーションが保護されます。構文は次のとおりです。</p> <pre>db2 import ... modified by delprioritychar ...</pre> <p>たとえば、次のような DEL データ・ファイルがあるとします。</p> <pre>"Smith, Joshua",4000,34.98<row delimiter> "Vincent,<row delimiter>, is a manager", 4005,44.37<row delimiter></pre> <p>delprioritychar 修飾子を指定すれば、このデータ・ファイルは 2 行だけになります。2 番目の <row delimiter> は 2 番目の行の最初のデータ列の一部として解釈されるのに対し、1 番目と 3 番目の <row delimiter> は実際のレコード区切り文字として解釈されます。この修飾子を指定しなかった場合は、このデータ・ファイルは 3 行のまま、各行は <row delimiter> によって区切られます。</p>
dldelx	<p><i>x</i> は単一文字の DATALINK 区切り文字です。省略時値はセミコロン (;) です。指定した文字はセミコロンの代わりに、DATALINK 値のフィールド間区切り文字として使用されます。DATALINK 値には副値が複数個含まれる場合があるため、これが必要になります。^{bc}</p> <p>注: 行、列、または文字ストリング区切り文字と同じ文字を <i>x</i> に指定することはできません。</p>
keepblanks	<p>タイプが CHAR、VARCHAR、LONG VARCHAR、または CLOB である各フィールドの前後の空白を保存します。このオプションを指定しないと、文字区切り文字で囲まれていないすべての前後の空白は除去され、表のすべての空白・フィールドにヌルが挿入されます。</p>
nodoubledel	<p>二重文字区切り文字の認識を押し止めます。詳細については、265 ページの『区切り文字の制限』を参照してください。</p>
IXF ファイル形式	
forcein	<p>コード・ページが一致していなくてもデータを受け入れ、コード・ページの変換を抑制するようにユーティリティに指示します。</p> <p>固定長ターゲット・フィールドは、データを入れるだけの十分な大きさがあるかどうかを検査されます。nochecklengths を指定した場合、そのような検査は実行されず、各行のインポートが試みられます。</p>

表 6. 有効なファイル・タイプ修飾子 (インポート) (続き)

修飾子	説明
indexif	既存の表に現在定義されている索引をすべて除去し、PC/IXF ファイルの索引定義に基づいて新しい索引を作成するようにユーティリティーに指示します。このオプションを使用できるのは、表の内容を置換する場合だけです。視点では使用できません。また、 <i>insert-column</i> を指定した場合には使用できません。
indexschema= <i>schema</i>	指定した <i>schema</i> を、索引作成時の索引名として使用します。 <i>schema</i> を指定しなかった場合 (しかしキーワード <i>indexschema</i> は指定した 場合) には、接続ユーザー ID が使用されます。このキーワードを指定しない場合、IXF ファイルのスキーマが使用されます。
nochecklengths	<i>nochecklengths</i> を指定した場合は、ソース・データの中にターゲット表列のサイズを超える列定義が含まれている場合であっても、各行のインポートが試みられます。コード・ページ変換によってソース・データが縮小されれば、そのような行であったとしても正常にインポートすることができます。たとえば、ソースに 4 バイトの EUC データがある場合、それがターゲットで 2 バイトの DBCS データに縮小されれば、必要なスペースは半分で済みます。列定義が一致しなくてもソース・データがきちんと収まるのが明らかな場合に、このオプションは特に便利です。

表 6. 有効なファイル・タイプ修飾子 (インポート) (続き)

修飾子	説明
注:	
1.	<p>MODIFIED BY オプションでサポートされていないファイル・タイプが指定されても、インポート・ユーティリティーが警告を発することはありません。サポートされていないファイル・タイプを使おうとすると、インポート操作は失敗し、エラー・コードが戻されます。</p>
2. ^a	<p>日付形式ストリングを囲む二重引用符は必須です。フィールド区切り文字には、a~z、A~Z、および 0~9 を含めることはできません。フィールド区切り文字は、文字区切り文字、または DEL ファイル形式のフィールド区切り文字と同じではありません。フィールド区切り文字は、要素の開始および終了位置が明確な場合は任意指定です。(修飾子によって) D、H、M、または S などの要素が使用される場合、項目が可変長であるためにあいまいさが存在することがあります。</p> <p>タイム・スタンプ形式の場合、月と分の記述子の両方に文字 M が使用されるので、あいまいさを避けるように注意する必要があります。月フィールドは、他の日付フィールドと隣接してなければなりません。分フィールドは、他の時間フィールドと隣接してなければなりません。以下に、いくつかのあいまいなタイム・スタンプ形式を示します。</p> <p>"M" (月または分のどちらにもとれる) "M:M" (月と分が区別がつかない) "M:YYYY:M" (両方とも月と解釈される) "S:M:YYYY" (時刻値と日付値の両方に隣接している)</p> <p>あいまいな場合、ユーティリティーはエラー・メッセージを報告し、操作は失敗します。</p> <p>以下に示すのは、明確なタイム・スタンプ形式です。</p> <p>"M:YYYY" (M (月)) "S:M" (M (分)) "M:YYYY:S:M" (M (月)...M (分)) "M:H:YYYY:M:D" (M (分)...M (月))</p>
	注: 二重引用符や円記号などの文字は、エスケープ文字が先行する必要があります (たとえば ¥)。
3. ^b	<p>この文字は、ソース・データのコード・ページで指定されている必要があります。</p> <p>(文字記号ではなく) 文字コード・ポイントは、xJJ または 0xJJ という構文で指定することができます (JJ はコード・ポイントの 16 進表示)。たとえば、列区切りとして # 文字を指定するには、以下のうちの 1 つを使用します。</p> <p>... modified by coldel# modified by coldel0x23 modified by coldelX23 ...</p>
4. ^c	<p>265ページの『区切り文字の制限』に、区切り文字の指定変更として使用できる文字に適用される制限のリストが示されています。</p>

以下も参照

257ページの『EXPORT』

IMPORT

411ページの『LOAD』

INITIALIZE TAPE

DB2 (Windows NT 版) は、ストリーム・テープ装置へのバックアップおよび復元をサポートしています。このコマンドは、テープを初期化するのに使います。

このコマンドは Windows NT でのみ使用可能です。

許可

なし

必須接続

なし

コマンド構文

```

▶▶—INITIALIZE TAPE—┬──ON—device──┬──USING—blksize──┬──▶▶

```

コマンド・パラメーター

ON device

有効なテープ装置名を指定します。省略時値は ¥¥.¥TAPE0 です。

USING blksize

装置のブロック・サイズを指定します (バイト単位)。値が装置のブロック・サイズとしてサポートされている範囲内であれば、装置は指定されたそのブロック・サイズで初期化されます。

注: 157ページの『BACKUP DATABASE』および 539ページの『RESTORE DATABASE』で指定するバッファ・サイズは、ここで指定するブロック・サイズで割り切れなければなりません。

このパラメーターに値を指定しなかった場合、装置は省略時のブロック・サイズで初期化されます。値ゼロを指定した場合は、装置は可変長のブロック・サイズで初期化されます。装置が可変長のブロック・モードをサポートしていない場合は、エラーが戻されます。

以下も参照

549ページの『REWIND TAPE』

575ページの『SET TAPE POSITION』

INVOKE STORED PROCEDURE

データベースのロケーションに保管されているプロシージャを呼び出します。データベース・アプリケーション・リモート・インターフェース (DARI) とも呼ばれます。サーバー・プロシージャがデータベースのロケーションで実行され、データをクライアント・アプリケーションに返します。

アプリケーション・プログラマーは、プログラムが 2 つの部分で実行されるよう、つまり一方はクライアント上、他方はサーバー上で実行されるように設計します。データベースにおけるサーバー・プロシージャは、クライアント・アプリケーションと同一のトランザクション内で実行されます。クライアント・アプリケーションとサーバー・プロシージャが同一のノード上にある場合、サーバー・プロシージャはローカルに実行されます。

注: このコマンドは、SQL CALL ステートメントによって置き換えられています (SQL 解説書を参照)。カタログ化されたストアード・プロシージャに対する CLP 内の CALL もサポートされています。

許可

データベースに対する CONNECT 特権。

必須接続

データベース

コマンド構文

```
► INVOKE program-name [ USING server-input-data ] ►
```

注: INVOKE を使って、入力または出力 SQLDA 構造を使用するサーバー・プロシージャや、データを返すサーバー・プロシージャを呼び出すことはしないでください。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。

コマンド・パラメーター

program-name

サーバー上で実行するプロシージャを指定します。このパラメーターは、以下の方法のいずれかを使って指定できます。

- 拡張子なしの *procName* によって。この表記では、*procName* という DARI ライブラリーをメモリーにロードするよう、データベース・

マネージャーに対して指示することになります。ここでは、実行される機能ルーチンの名前とライブラリー名が同一であることを前提にしています。たとえば、次のように指定します。

```
db2 invoke foo
```

の場合、F00 という名前の DARI ライブラリーがロードされ、そのライブラリー内の機能ルーチン F00() が実行されます。

データベース・マネージャーは、インスタンス所有者の省略時ディレクトリー \$HOME/sql/lib/function の中から DARI ライブラリーを検索します。

- 感嘆符 (!) で区切った名前を指定することによって (例: *procName!funcName*)。この表記では、*procName* という DARI ライブラリーをメモリーにロードするよう、データベース・マネージャーに対して指示することになります。実行される機能ルーチンは、*funcName* です。この規則により、いくつかの類似の機能ルーチンを同一の DARI ライブラリーにパッケージ化することが可能になります。
- 絶対パス名を指定することによって (例: */u/cche/procName!/funcName*)。この表記には、DARI ライブラリーの記憶パスが含まれます。この例では、DARI ライブラリー *procName* がディレクトリー */u/cche* に保管されています。実行される機能ルーチンは、*funcName* です。

注: DB2 プロダクトのさまざまなバージョンの間の移行性サポートのために、! 区切り文字は円記号 (¥) で置き換えることができます。

USING server-input-data

サーバー・ルーチンに渡す情報を指定します。特定の必要に応じて入力データを送信するために使うことのできる、可変長文字ストリングで自由形式の柔軟なパラメーターです。

LIST ACTIVE DATABASES

LIST ACTIVE DATABASES

GET SNAPSHOT FOR ALL DATABASES コマンドによってリスト表示される情報のサブセットを表示します (312ページの『GET SNAPSHOT』を参照)。活動中のデータベースは、すべてのアプリケーションが接続して使用できません。活動中のデータベースごとに、このコマンドは以下を表示します。

- データベース名
- データベースに現在接続しているアプリケーションの数
- データベース・パス

効力範囲

このコマンドは、`$HOME/sqllib/db2nodes.cfg` の中に指定されている任意のノードから発行できます。これらのどのノードからでも同一の情報が戻されます。

許可

以下のどれかが必要です。

- `sysadm`
- `sysctrl`
- `sysmaint`

コマンド構文

```
▶—LIST ACTIVE DATABASES—▶  
└─AT NODE—nodenum—┘  
└─GLOBAL—┘
```

コマンド・パラメーター

AT NODE *nodenum*

モニター・スイッチの状況を表示するノードを指定します。

GLOBAL

区分データベース・システム内のすべてのノードの集合結果を戻します。結果を集める方法については、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。

例

以下に示すのは、LIST ACTIVE DATABASES コマンドの出力例です。

LIST ACTIVE DATABASES

```
Active Databases
Database name          = TEST
Applications connected currently = 0
Database path          = /home/smith/smith/NODE0000/SQL00002/
Database name          = SAMPLE
Applications connected currently = 1
Database path          = /home/smith/smith/NODE0000/SQL00001/
```

以下も参照

312ページの『GET SNAPSHOT』

148ページの『ACTIVATE DATABASE』

243ページの『DEACTIVATE DATABASE』

LIST APPLICATIONS

LIST APPLICATIONS

活動中のすべてのデータベース・アプリケーションに関して、アプリケーション・プログラム名、許可 ID (ユーザー名)、アプリケーション・ハンドル、アプリケーション ID、およびデータベース名を標準出力に出力します。このコマンドでは、オプションとしてアプリケーションの順序番号、状況、状況変更時刻、およびデータベース・パスを表示することもできます。

効力範囲

このコマンドは、それが発行されたノードの情報だけを戻します。

許可

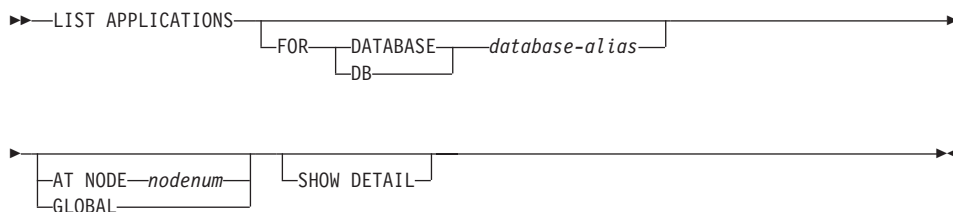
以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

インスタンス。 リモート・インスタンスのアプリケーションをリスト表示するには、まず最初にそのインスタンスに接続する必要があります。

コマンド構文



コマンド・パラメーター

FOR DATABASE database-alias

指定したデータベースに接続された各アプリケーションの情報が表示されます。データベース名情報は表示されません。このオプションを指定しない場合、このコマンドは、ユーザーが現在接続しているノードにあるデータベースと接続している各アプリケーションごとに、情報を表示します。

省略時のアプリケーション情報には、以下のものが含まれます。

- 許可 ID
- アプリケーション・プログラム名
- アプリケーション・ハンドル
- アプリケーション ID
- データベース名

AT NODE nodenum

モニター・スイッチの状況を表示するノードを指定します。

GLOBAL

区分データベース・システム内のすべてのノードの集合結果を戻します。結果を集める方法については、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。

SHOW DETAIL

出力には、以下の追加情報が含まれます。

- 順序番号
- アプリケーションの状況
- 状況変更時刻
- データベース・パス

注: このオプションを指定した場合には、出力をファイルにリダイレクトしておいて、エディターを使ってそのレポートを表示するのがよいでしょう。画面上に表示する場合には、出力行が折り返す場合もあります。

例

次に、LIST APPLICATIONS の出力例を示します。

Auth Id	Application Name	Appl. Handle	Application Id	DB Name	# of Agents
smith	db2bp_32	12	*LOCAL.smith.970220191502	TEST	1
smith	db2bp_32	11	*LOCAL.smith.970220191453	SAMPLE	1

注: 上記のフィールドについては、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。

使用上の注意

データベース管理者は、このコマンドの出力を問題判別の参考にすることができます。さらにこの情報は、データベース管理者がアプリケーションの中で、

LIST APPLICATIONS

312ページの『GET SNAPSHOT』または 267ページの『FORCE APPLICATION』を使用する場合に必要になります。

リモート・インスタンス (または別のローカル・インスタンス) のアプリケーションをリスト表示するには、まず最初にそのインスタンスに接続する必要があります。接続がすでに存在しているのに FOR DATABASE を指定し、かつ現行の接続とは異なるインスタンスにデータベースが存在している場合、このコマンドは失敗します。

LIST COMMAND OPTIONS

環境変数の現行の設定値のリストを表示します。

- **DB2BQTIME**
- **DB2DQTRY**
- **DB2RQTIME**
- **DB2IQTIME**
- **DB2OPTIONS**

許可

なし

必須接続

なし

コマンド構文

▶—LIST COMMAND OPTIONS—▶

コマンド・パラメーター

なし

例

以下に示すのは、LIST COMMAND OPTIONS の出力例です。

```

Command Line Processor Option Settings
Backend process wait time (seconds)      (DB2BQTIME) = 1
No. of retries to connect to backend     (DB2DQTRY) = 60
Request queue wait time (seconds)        (DB2RQTIME) = 5
Input queue wait time (seconds)          (DB2IQTIME) = 5
Command options                          (DB2OPTIONS) =
Option  Description                      Current Setting
-----
-a    Display SQLCA                        OFF
-c    Auto-Commit                          ON
-e    Display SQLCODE/SQLSTATE            OFF
-f    Read from input file                 OFF
-l    Log commands in history file         OFF
-n    Remove new line character            OFF
-o    Display output                       ON
-p    Display interactive input prompt     ON
-r    Save output to report file           OFF
-s    Stop execution on command error      OFF
-t    Set statement termination character   OFF

```

LIST COMMAND OPTIONS

-v	Echo current command	OFF
-w	Display FETCH/SELECT warning messages	ON
-z	Save all output to output file	OFF

使用上の注意

詳しくは、125ページの『コマンド行プロセッサのオプション』を参照してください。

LIST DATABASE DIRECTORY

システム・データベース・ディレクトリーの内容をリスト表示します。パスが指定されている場合、ローカル・データベース・ディレクトリーの内容が表示されます。

効力範囲

ON *path* パラメーターを指定しないでこのコマンドを発行すると、システム・データベース・ディレクトリーが戻されます。この情報はすべてのノードで同じです。

ON *path* パラメーターを指定すると、指定したパスのローカル・データベース・ディレクトリーが戻されます。この情報はノードによって異なります。

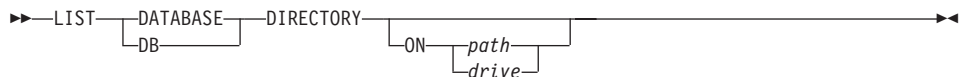
許可

なし

必須接続

なし。ディレクトリー操作は、ローカル・ディレクトリーだけに影響します。

コマンド構文



コマンド・パラメーター

ON path/drive

情報を表示するローカル・データベース・ディレクトリーを指定します。これを指定しない場合、システム・データベース・ディレクトリーの内容が表示されます。

例

次に示すのは、システム・データベース・ディレクトリーの場合の出力例です。

```
System Database Directory
Number of entries in the directory = 2
```

```
Database 1 entry:
Database alias           = SAMPLE
Database name           = SAMPLE
```

LIST DATABASE DIRECTORY

```
Database drive           = /home/smith
Database release level   = 8.00
Comment                  =
Directory entry type     = Indirect
Catalog node number      = 0
Database 2 entry:
Database alias           = GDB1
Database global name     = ../../cell_name/dir_name/gdb1
Database release level   = 8.00
Comment                  = DCE database
Directory entry type     = DCE
Catalog node number      = -1
```

次に示すのはローカル・データベース・ディレクトリーの場合の出力例です。

```
Local Database Directory on /u/smith
Number of entries in the directory = 1
```

```
Database 1 entry:
Database alias           = SAMPLE
Database name            = SAMPLE
Database directory       = SQL00001
Database release level   = 8.00
Comment                  =
Directory entry type     = Home
Catalog node number      = 0
Node number              = 0
```

各フィールドは、次のとおりです。

Database alias

データベースの作成時またはカタログ化時の *alias* パラメーターの値。データベースのカタログ化時に別名が入力されなかった場合、データベース・マネージャーは、データベースのカタログ化時の *database-name* パラメーターの値を使用します。

Database global name

DCE 名前空間内でデータベースを固有に識別する完全修飾名。

Database name

データベースのカタログ化時の *database-name* パラメーターの値。多くの場合、この名前はデータベース作成時点での名前です。

Local Database Directory

データベースが存在しているパス。このフィールドが表示されるのは、システム・データベース・ディレクトリーが走査された場合だけです。

Database directory/Database drive

データベースが存在しているディレクトリーまたはドライブ名。このフィールドが表示されるのは、ローカル・データベース・ディレクトリーが走査された場合だけです。

Node name

リモート・ノードの名前。この名前は、データベースおよびノードのカタログ化時に *nodename* パラメーターに入力した値に対応します。

Database release level

データベースに対して実行可能なデータベース・マネージャーのリリース・レベル。

Comment

データベースをカタログ化した時点で入力された、データベースに関連する注釈。

Directory entry type

データベースの存在場所。

- *remote* 項目には、別のノードにあるデータベースについて記述されます。
- *indirect* 項目にはローカルのデータベースについて記述されます。システム・データベース・ディレクトリーと同じノードにあるデータベースは、ローカル・データベース・ディレクトリーに対するホーム (*home*) 項目を間接的に参照しているとみなされるため、間接 (*indirect*) 項目とみなされます。
- *home* 項目は、そのデータベース・ディレクトリーがローカル・データベース・ディレクトリーと同じパスにあることを示します。
- *LDAP* 項目は、データベース位置情報が *LDAP* ディレクトリーに保管されることを示します。

システム・データベース・ディレクトリーにあるすべての項目は、リモート (*remote*) か間接 (*indirect*) です。システム・データベース・ディレクトリーにあるローカル・データベース・ディレクトリーの項目は、すべて間接 (*indirect*) 項目として表示されます。

Authentication

クライアントでカタログ化されている認証タイプで、システム、Kerberos、または DCE セキュリティーのどれと接続しているのかを判別するのに使われるもの。

Principal name

完全修飾の Kerberos または DCE プリンシパル名を指定します。

LIST DATABASE DIRECTORY

Catalog node number

どのノードをカタログ・ノードにするかを指定します (MPP システムのみ)。これは、CREATE DATABASE コマンドを発行したノードです。

Node number

db2nodes.cfg の中で、このコマンドを発行したノードに対して割り当てられている番号を指定します (MPP システムのみ)。db2nodes.cfg ファイルがない非 MPP システムの場合、ノード番号は常に 0 です。

使用上の注意

1 プロセスにつき、最大 8 つのデータベース・ディレクトリー・スキャンをオープンさせることができます。単一の DB2 セッション内で 9 つ以上の LIST DATABASE DIRECTORY コマンドを発行できないというこのバッチ・ファイルの制限を解決するためには、バッチ・ファイルをシェル・スクリプトに変換してください。"db2" 接頭部を使用すれば、コマンドごとに新しい DB2 セッションが生成されます。

以下も参照

230ページの『CHANGE DATABASE COMMENT』

235ページの『CREATE DATABASE』

LIST DATALINKS MANAGERS

指定されたデータベースに登録されている DB2 データ・リンク・マネージャーのリストを表示します。

許可

なし

コマンド構文

```
▶▶—LIST DATALINKS MANAGERS FOR—

|          |
|----------|
| DATABASE |
| DB       |

—dbname—▶▶
```

コマンド・パラメーター

DATABASE dbname

データベース名を指定します。

以下も参照

150ページの『ADD DATALINKS MANAGER』

LIST DCS APPLICATIONS

LIST DCS APPLICATIONS

DB2 コネクト エンタープライズ・エディションによってホスト・データベースに接続されているアプリケーションに関する情報を、標準出力に表示します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

インスタンス。 リモート・インスタンスの DCS アプリケーションをリスト表示するには、まず最初にそのインスタンスに接続する必要があります。

コマンド構文

```
▶▶—LIST DCS APPLICATIONS—▶▶  
    ┌—SHOW DETAIL—┐  
    └—EXTENDED—┘
```

コマンド・パラメーター

LIST DCS APPLICATIONS

省略時のアプリケーション情報の内容は、以下のとおりです。

- ホスト許可 ID (*username*)
- アプリケーション・プログラム名
- アプリケーション・ハンドル
- アウトバウンド・アプリケーション ID (*luwid*)

SHOW DETAIL

出力に以下の追加情報を含めることを指定します。

- クライアント・アプリケーション ID
- クライアント順序番号
- クライアント・データベース別名
- クライアント・ノード名 (*nname*)
- クライアント・リリース・レベル
- クライアント・コード・ページ

- アウトバウンド順序番号
- ホスト・データベース名
- ホスト・リリース・レベル

EXTENDED

拡張されたレポートを生成します。このレポートには、**SHOW
DETAIL** オプションを指定した場合に出力されるすべてのフィールドに加えて、以下の追加フィールドが含まれます。

- 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
- クライアント・コード・ページ
- クライアント・アプリケーションのプロセス ID
- ホスト・コード化文字セット ID (CCSID)

例

次に示すのは LIST DCS APPLICATIONS の出力例です。

Auth Id	Application Name	Appl. Handle	Outbound Application Id
DDCSUS1	db2bp_s	2	0915155C.139D.971205184245

次に示すのは LIST DCS APPLICATIONS EXTENDED の出力例です。

LIST DCS APPLICATIONS

```
                List of DCS Applications - Extended Report
Client application ID      = 09151251.0AD1.980529194106
Sequence number          = 0001
Authorization ID         = SMITH
Application name         = db2bp
Application handle       = 0
Application status       = waiting for reply
Status change time      = Not Collected
Client DB alias         = MVSDB
Client node             = antman
Client release level    = SQL05020
Client platform         = AIX
Client protocol         = TCP/IP
Client codepage         = 819
Process ID of client application = 38340
Client login ID         = user1
Host application ID     = G9151251.GAD2.980529194108
Sequence number        = 0000
Host DB name           = GILROY
Host release level     = DSN05011
Host CCSID             = 500
```

注:

1. アプリケーション状況 (Application status) の値は、以下のうちのいずれかです。

connect pending - outbound

ホスト・データベースとの接続要求が発行され、その接続が確立されるのを DB2 コネクトが待機している状態。

waiting for request

ホスト・データベースとの接続がすでに確立され、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 DB2 コネクトが待機している状態。

waiting for reply

SQL ステートメントがホスト・データベースに送られた状態。

2. 状況変更時刻 (Status change time) が表示されるのは、システム・モニターが処理中にその UOW スイッチがオンになっていた場合だけです。それ以外の場合は、Not Collected と表示されます。
3. 上記のフィールドについては、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。

使用上の注意

データベース管理者は、このコマンドを使用することによって、ゲートウェイへのクライアント・アプリケーション接続と、対応するゲートウェイからのホスト接続を一致させることができます。

またデータベース管理者は、エージェント ID 情報を使うことによって、指定したアプリケーションを DB2 コネクト・サーバーから切り離すことができます。

以下も参照

267ページの『FORCE APPLICATION』

LIST DCS DIRECTORY

LIST DCS DIRECTORY

データベース接続サービス (DCS) ディレクトリーの内容をリスト表示します。

許可

なし

必須接続

なし

コマンド構文

▶—LIST DCS DIRECTORY—▶

コマンド・パラメーター

なし

例

次に示すのは LIST DCS DIRECTORY の出力例です。

```
Database Connection Services (DCS) Directory
```

```
Number of entries in the directory = 1
```

```
DCS 1 entry:
```

```
Local database name           = DB2
Target database name          = DSN_DB_1
Application requestor name    =
DCS parameters                =
Comment                       = DB2/MVS Location name DSN_DB_1
DCS directory release level   = 0x0100
```

各フィールドは、次のとおりです。

Local database name

ターゲット・ホスト・データベースの別名を指定します。これは、DCS ディレクトリーの中にホスト・データベースをカタログ化した時点で入力した *database-name* パラメーターに対応します。

Target database name

アクセスできるホスト・データベースの名前を指定します。これは、

DCS ディレクトリーの中にホスト・データベースをカタログ化した時点で入力した *target-database-name* パラメーターに対応します。

Application requester name

アプリケーション・リクエスターまたはサーバー上に存在するプログラムの名前を指定します。

DCS parameters

アプリケーション・リクエスターで使用される接続および操作環境パラメーターを含むストリング。ホスト・データベースをカタログ化した時点でのパラメーター・ストリングに対応します。ストリングは二重引用符で囲み、パラメーターはコンマで区切る必要があります。

DCS パラメーターについては、*DB2 コネクト 使用者の手引き* を参照してください。

Comment

データベース項目の説明。

DCS directory release level

データベースが作成された分散データベース接続サービス・プログラムのバージョン番号を指定します。

使用上の注意

DCS ディレクトリーは、199ページの『CATALOG DCS DATABASE』を最初に呼び出した時点で作成されます。これは、DB2 のインストール先パス / ドライブにあり、DB2 コネクト・プログラムがインストールされている場合にワークステーションからアクセスできるホスト・データベースについての情報を提供します。ホスト・データベースとしては次のものが可能です。

- DB2 (MVS/ESA 版) データベース (システム/370 およびシステム/390 アーキテクチャー・ホスト・コンピューター)
- DB2 (VSE および VM 版) データベース (システム/370 およびシステム/390 アーキテクチャー・ホスト・コンピューター)
- AS/400 ホスト・コンピューター上の DB2 ユニバーサル・データベース (AS/400 版) データベース

LIST DRDA INDOUBT TRANSACTIONS

LIST DRDA INDOUBT TRANSACTIONS

DRDA リクエスターと DRDA サーバーの間の未確定トランザクションのリストを表示します。APPC コミット・プロトコルが使用されている場合は、パートナー LU 相互間の未確定トランザクションが表示されます。DRDA コミット・プロトコルが使用されている場合は、DRDA 同期点管理プログラム相互間の未確定トランザクションが表示されます。

許可

sysadm

必須接続

インスタンス

コマンド構文

▶—LIST DRDA INDOUBT TRANSACTIONS—▶
└─WITH PROMPTING─┘

コマンド・パラメーター

WITH PROMPTING

未確定トランザクションを処理することを指定します。このパラメーターを指定すると、対話式ダイアログ・モードが開始され、未確定トランザクションのコミットまたはロールバックが可能になります。このパラメーターを指定しない場合、未確定トランザクションは標準出力装置に書き込まれ、対話式ダイアログ・モードは開始されません。

注: 破棄 (forget) オプションはサポートされていません。未確定トランザクションをコミットまたはロールバックすると、そのトランザクションは自動的に破棄されます。

対話式ダイアログ・モードでは次のことが可能です。

- すべての未確定トランザクションのリスト表示 (1 を入力)
- 未確定トランザクション番号 *x* のリスト表示 (1 の後に有効なトランザクション番号を入力)
- 終了 (q を入力)
- トランザクション番号 *x* をコミット (c の後に有効なトランザクション番号を入力)

- トランザクション番号 x をロールバック (r の後に有効なトランザクション番号を入力)

注: コマンド文字と引き数の間は、ブランク・スペースで区切る必要があります。

トランザクションのコミットまたはロールバックを実行する前に、トランザクション・データが表示され、アクションを確認するように求められます。

使用上の注意

DRDA 未確定トランザクションが発生するのは、分散作業単位内の調整プログラムと参加プログラムとの間の通信が失われた場合です。分散作業単位では、ユーザーやアプリケーションが、単一の作業単位内で複数の場所にあるデータを読んだり更新したりできます。そのような作業には、2 フェーズ・コミットが必要となります。

第 1 のフェーズでは、すべての参加プログラムに対してコミットの準備を要求します。第 2 のフェーズでは、トランザクションをコミットまたはロールバックします。第 1 フェーズ終了後に調整プログラムまたは参加プログラムが使用できなくなると、分散トランザクションが未確定になります。

LIST DRDA INDOUBT TRANSACTIONS コマンドを発行するには、その前にアプリケーション・プロセスは、DB2 同期点管理プログラム (SPM) のインスタンスに接続する必要があります。CONNECT ステートメントの *dbalias* として、データベース・マネージャー構成パラメーター *spm_name* を使います。CONNECT ステートメントの使用方法については、*SQL 解説書* を参照してください。

コミットの調整に SPM を使う TCP/IP 接続では、DRDA 2 フェーズ・コミット・プロトコルを使います。APPC 接続では、LU 6.2 2 フェーズ・コミット・プロトコルを使います。

LIST HISTORY

LIST HISTORY

活動記録ファイルの中の項目のリストを表示します。活動記録ファイルには、回復と管理のさまざまな事象の記録が含まれています。回復事象には、データベース・レベルおよび表スペース・レベルの完全なバックアップ、復元、およびロールフォワードが含まれます。さらにログ記録される事象には、表スペース更新、統計実行、表の再編成、表の除去、およびロードが含まれます。

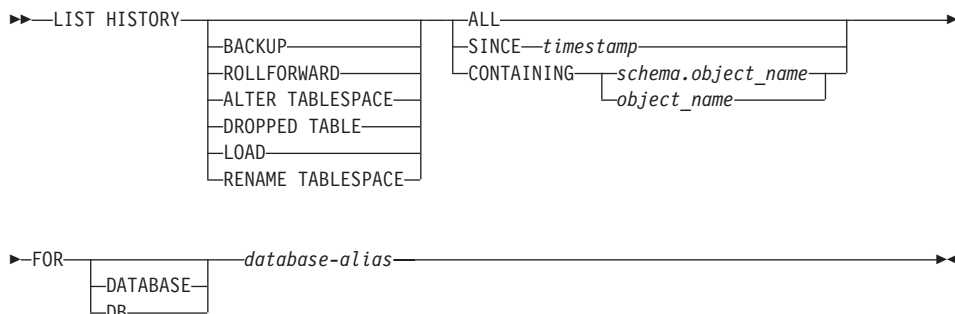
許可

なし

必須接続

インスタンス。明示的な接続は必要ありません。データベースがリモートとして示されている場合、リモート・ノードへのインスタンス接続はコマンドの持続期間の間、ずっと確立されたままになります。

コマンド構文



コマンド・パラメーター

HISTORY

現在活動記録ファイルの中に記録されている回復と管理に関する事象のすべてのリストを表示します。

BACKUP

バックアップおよび復元操作のリストを表示します。

ROLLFORWARD

ロールフォワード操作のリストを表示します。

ALTER TABLESPACE

ALTER TABLESPACE 操作のリストを表示します。

DROPPED TABLE

除去された表のリストを表示します。

LOAD ロード操作のリストを表示します。

RENAME TABLESPACE

RENAME TABLESPACE 操作のリストを表示します。

ALL 活動記録ファイルのうち、指定したタイプのすべての項目のリストを表示します。

SINCE timestamp

完全なタイム・スタンプ (形式は `yyyymmddhhnnss`)、または先頭の接頭部 (最小値は `yyyy`) を指定できます。指定したタイム・スタンプ以前のタイム・スタンプの項目のリストを表示します。

CONTAINING schema.object_name

この修飾名は表を固有に識別します。

CONTAINING object_name

この非修飾名は表スペースを固有に識別します。

FOR DATABASE database-alias

回復活動記録ファイルをリスト表示するデータベースを指定します。

例

```
db2 list history since 19980201 for sample
db2 list history backup containing userspace1 for sample
db2 list history dropped table all for db sample
```

使用上の注意

このコマンドによって生成されるレポートには、以下の記号が含まれます。

- 装置
 - A - TSM
 - C - クライアント
 - D - ディスク
 - K - ディスケット
 - L - ローカル
 - O - その他
 - P - パイプ
 - S - サーバー
 - T - テープ

LIST HISTORY

- U - ユーザー出口
- オブジェクト
 - D - データベース
 - P - 表スペース
 - T - 表
- 操作
 - B - バックアップ
 - C - コピー
 - D - 除去された表
 - F - ロールフォワード
 - G - 表の再編成
 - L - ロード
 - N - 表スペースの名前変更
 - Q - 静止
 - R - 復元
 - S - 統計の実行
 - T - 表スペースの更新
 - U - (将来の利用)
- タイプ
 - C - 表スペースの更新 (コンテナの追加)
 - E - ログの終了
 - F - オフライン
 - I - 挿入 (LOAD)
 - N - オンライン
 - P - 時刻
 - R - 表スペースの更新 (再調整)
 - S - 静止共用
 - U - 静止更新
 - X - 静止排他
 - Z - 静止リセット
- 項目の状況
 - D - 削除 (将来の利用)
 - E - 有効期限切れ

- I - 非活動
- N - コミット未完了
- Y - コミット済みまたは活動中

LIST INDOUBT TRANSACTIONS

LIST INDOUBT TRANSACTIONS

未確定トランザクションのリストを表示します。未確定トランザクションのコミット、ロールバック、または破棄を対話式で実行できます。

2 フェーズ・コミット・プロトコルは、以下のもので構成されます。

1. **PREPARE** フェーズ。このフェーズでは、リソース・マネージャーがログ・ページをディスクに書き込んでいるので、**COMMIT** または **ROLLBACK** プリミティブのどちらにも応答することができます。
2. **COMMIT** (または **ROLLBACK**) フェーズ。このフェーズでは、トランザクションの実際のコミットまたはロールバックが実行されます。

未確定トランザクションは、準備済みのトランザクションのうち、コミットまたはロールバックが実行されていないものです。

効力範囲

このコマンドは、このコマンドが実行されたノード上にある未確定トランザクションのリストを戻します。

許可

dbadm

必須接続

データベース。暗黙接続が可能な場合には、省略時データベースへの接続が確立されます。

コマンド構文

```
▶—LIST INDOUBT TRANSACTIONS—┐  
└─WITH PROMPTING─┘▶
```

コマンド・パラメーター

WITH PROMPTING

未確定トランザクションを処理することを指定します。このパラメーターを指定すると、対話式ダイアログ・モードが開始され、未確定トランザクションのコミット、ロールバック、または破棄が可能になります。このパラメーターを指定しない場合、未確定トランザクションは標準出力装置に書き込まれ、対話式ダイアログ・モードは開始されません。

対話式ダイアログ・モードでは次のことが可能です。

LIST INDOUBT TRANSACTIONS

- すべての未確定トランザクションのリスト表示 (l を入力)
- 未確定トランザクション番号 x のリスト表示 (l の後に有効なトランザクション番号を入力)
- 終了 (q を入力)
- トランザクション番号 x をコミット (c の後に有効なトランザクション番号を入力)
- トランザクション番号 x をロールバック (r の後に有効なトランザクション番号を入力)
- トランザクション番号 x を破棄 (f の後に有効なトランザクション番号を入力)

注: コマンド文字と引き数の間は、ブランク・スペースで区切る必要があります。

トランザクションのコミット、ロールバック、または破棄を実行する前に、トランザクション・データが表示され、アクションを確認するように求められます。

例

次に示すのは、LIST INDOUBT TRANSACTIONS によって生成されるダイアログ例です。

In-doubt Transactions for Database SAMPLE

```
1.  originator: XA
    appl_id: *LOCAL.DB2.95051815165159      sequence_no: 0001 status: i
    timestamp: 05-18-1997 16:51:59 auth_id: SMITH log_full: n type: RM
    xid: 53514C2000000017 00000000544D4442 00000000002F93DD A92F8C4FF3000000
    0000BD
```

```
2.  originator: XA
    appl_id: *LOCAL.DATABASE.950407161043 sequence_no: 0002 status: i
    timestamp: 04-07-1997 16:10:43 auth_id: JONES log_full: n type: RM
    xid: 53514C2000000017 00000000544D4442 00000000002F95FE B62F8C4FF3000000
    0000C1
```

```
  .
  .
  .
```

Enter in-doubt transaction command or 'q' to quit.
e.g. 'c 1' heuristically commits transaction 1.
c/r/f/l/q: c 1

```
1.  originator: XA
    appl_id: *LOCAL.DB2.95051815165159      sequence_no: 0001 status: i
```

LIST INDOUBT TRANSACTIONS

```
timestamp: 05-18-1997 16:51:59 auth_id: SMITH log_full: n type: RM
xid: 53514C2000000017 00000000544D4442 00000000002F93DD A92F8C4FF3000000
0000BD
```

Do you want to heuristically commit this in-doubt transaction ? (y/n) y

DB20000I "COMMIT INDOUBT TRANSACTION" completed successfully

c/r/f/l/q: c 5

DB20030E "5" is not a valid in-doubt transaction number.

c/r/f/l/q: 1

In-doubt Transactions for Database SAMPLE

```
1.  originator: XA
    appl_id: *LOCAL.DB2.95051815165159      sequence_no: 0001 status: c
timestamp: 05-18-1997 16:51:59 auth_id: SMITH log_full: n type: RM
xid: 53514C2000000017 00000000544D4442 00000000002F93DD A92F8C4FF3000000
0000BD
```

```
2.  originator: XA
    appl_id: *LOCAL.DATABASE.950407161043  sequence_no: 0002 status: i
timestamp: 04-07-1997 16:10:43 auth_id: JONES log_full: n type: RM
xid: 53514C2000000017 00000000544D4442 00000000002F95FE B62F8C4FF3000000
0000C1
```

```
.
.
.
```

c/r/f/l/q: r 2

```
2.  originator: XA
    appl_id: *LOCAL.DATABASE.950407161043  sequence_no: 0002 status: i
timestamp: 04-07-1997 16:10:43 auth_id: JONES log_full: n type: RM
xid: 53514C2000000017 00000000544D4442 00000000002F95FE B62F8C4FF3000000
0000C1
```

Do you want to heuristically rollback this in-doubt transaction ? (y/n) y

DB20000I "ROLLBACK INDOUBT TRANSACTION" completed successfully

c/r/f/l/q: 1 2

```
2.  originator: XA
    appl_id: *LOCAL.DATABASE.950407161043  sequence_no: 0002 status: r
timestamp: 04-07-1997 16:10:43 auth_id: JONES log_full: n type: RM
xid: 53514C2000000017 00000000544D4442 00000000002F95FE B62F8C4FF3000000
0000C1
```

c/r/f/l/q: f 2

```
2.  originator: XA
    appl_id: *LOCAL.DATABASE.950407161043  sequence_no: 0002 status: r
timestamp: 04-07-1997 16:10:43 auth_id: JONES log_full: n type: RM
```

LIST INDOUBT TRANSACTIONS

```
xid: 53514C2000000017 00000000544D4442 00000000002F95FE B62F8C4FF3000000  
0000C1
```

```
Do you want to forget this in-doubt transaction ? (y/n) y
```

```
DB20000I "FORGET INDOUBT TRANSACTION" completed successfully
```

```
c/r/f/l/q: l 2
```

```
2. originator: XA  
   appl_id: *LOCAL.DATABASE.950407161043 sequence_no: 0002 status: f  
   timestamp: 04-07-1997 16:10:43 auth_id: JONES log_full: n type: RM  
   xid: 53514C2000000017 00000000544D4442 00000000002F95FE B62F8C4FF3000000  
   0000C1
```

```
c/r/f/l/q: q
```

注: LIST INDOUBT TRANSACTIONS コマンドは、それぞれの未確定トランザクションでのデータベースの役割を示す、以下のタイプ 情報を戻します。

TM 未確定トランザクションは、データベースをトランザクション・マネージャー・データベースとして使用することを示します。

RM 未確定トランザクションは、データベースをリソース・マネージャーとして使用することを示します。つまり、それがトランザクションに参加する複数のデータベースの 1 つであっても、トランザクション・マネージャー・データベースではないことを示します。

使用上の注意

未確定トランザクションは、未確定状態のままになっているグローバル・トランザクションです。これは、2 フェーズ・コミット・プロトコルの第 1 フェーズ (つまり PREPARE フェーズ) を正常終了した後、トランザクション・マネージャー (TM)、または少なくとも 1 つのリソース・マネージャー (RM) のいずれかが使用できなくなった場合に発生します。RM がもう一度使用可能になり、TM が RM からの未確定状況情報に関するログを統合できるようになるまで、RM はトランザクションの分岐をコミットするのかわかれともロールバックするかわかりません。詳しくは、*管理の手引き* の中の、DB2 を XA 互換トランザクション・マネージャーと一緒に使用する方法について説明している章を参照してください。未確定トランザクションは MPP 環境にも存在させることができます。詳しくは、*管理の手引き* の中の、障害が発生したデータベース区画サーバーでトランザクション障害を回復する方法について説明している節を参照してください。

LIST INDOUBT TRANSACTIONS

現在接続されているデータベースに対して LIST INDOUBT TRANSACTIONS が出された場合、そのデータベースの未確定トランザクションに関する情報が戻されます。

コミットできるのは、状況が未確定 (i)、またはコミット肯定応答欠落 (m) のトランザクションだけです。

ロールバックできるのは、状況が未確定 (i) または終了済み (e) のトランザクションだけです。

破棄できるのは、状況がコミット済み (c) またはロールバック済み (r) のトランザクションだけです。

注: 2 フェーズ・コミットのコミット・フェーズでは、調整プログラム・ノードがコミットの肯定応答を待機します。(ノード障害などの理由で) 応答しないノードが 1 つ以上ある場合、そのトランザクションはコミット肯定応答欠落状態になります。

未確定トランザクション情報は、コマンドが出された時点でしか有効ではありません。対話式ダイアログ・モードに入ってしまうと、外部の活動のためにトランザクション状況が変更されることがあります。その場合、該当する状況にない未確定トランザクションを処理しようとする、エラー・メッセージが表示されます。

このタイプのエラーが発生した場合、ユーザーは対話式ダイアログを終了 (q) しなければなりません。そして、表示される情報を最新表示にするために、**LIST INDOUBT TRANSACTIONS WITH PROMPTING** コマンドを再発行する必要があります。

詳しくは、[管理の手引き](#) を参照してください。

LIST NODE DIRECTORY

ノード・ディレクトリーの内容をリスト表示します。

許可

なし

必須接続

なし

コマンド構文

```

▶—LIST—┐──┬──NODE DIRECTORY──┐──┬──▶
          └──┬──┘                └──┬──┘
              ADMIN                SHOW DETAIL
  
```

コマンド・パラメーター**ADMIN**

管理サーバー・ノードを指定します。

SHOW DETAIL

以下の情報を出力に含めることを指定します。

- リモート・インスタンス名
- システム
- オペレーティング・システムのタイプ

例

次に示すのは LIST NODE DIRECTORY の出力例です。

```
Node Directory
```

```
Number of entries in the directory = 6
```

```
Node 1 entry:
```

```

Node name           = DB2APPC1
Comment             = A remote APPC node
Protocol            = APPC
Symbolic destination name = db2inst1
Security type       = PROGRAM
  
```

```
Node 2 entry:
```

```

Node name           = DB2IPX1
Comment             = A remote IPX/SPX node
Protocol            = IPXSPX
  
```

LIST NODE DIRECTORY

File server name = netwsrv
Bindery object name = db2inst1

Node 3 entry:

Node name = DB2IPX2
Comment = IPX/SPX node using direct addr
Protocol = IPXSPX
File server name = *
Bindery object name = 09212700.400011527745.879E

Node 4 entry:

Node name = DB2NETB1
Comment = A remote NetBIOS node
Protocol = NETBIOS
Adapter number = 0
Server NNAME = DB2INST1

Node 5 entry:

Node name = DB2TCP1
Comment = A remote TCP/IP node
Protocol = TCPIP
Hostname = tcphost
Service name = db2inst1

Node 6 entry:

Node name = DB2TCP2
Comment = TCP/IP node using IP address
Protocol = TCPIP
Hostname = 9.21.15.235
Service name = db2inst2

共通フィールドは、次のとおりです。

Node name

リモート・ノードの名前。これは、ノードのカタログ化時に *nodename* に入力された名前に対応します。

Comment

ノードのカタログ化時に入力された、ノードに関連する注釈。ノード・ディレクトリー内の注釈を変更するには、そのノードをアンカタログしてから、別の注釈を付けてもう一度カタログ化します。

Protocol

ノード用にカタログ化された通信プロトコル。

注: 個々のノード・タイプに関連したフィールドについては、該当する CATALOG...NODE コマンドを参照してください。

使用上の注意

ノードは、個々のデータベース・クライアントで作成および保守されます。これには、そのクライアントからアクセスできるデータベースを含む各リモート・ワークステーションごとに 1 つの項目が含まれています。DB2 クライアントは、データベース接続やインスタンス接続が要求されると、常にノード・ディレクトリー内の通信エンドポイント情報を使います。

データベース・マネージャーは、CATALOG...NODE コマンドを処理するたびに、ノード項目を作成してそれをノード・ディレクトリーに追加します。その項目は、ノードが使用する通信プロトコルによって異なります。

ノード・ディレクトリーには、次のようなタイプのノード用ディレクトリーを含めることができます。

- APPC
- APPCLU
- APPN
- IPX/SPX
- ローカル
- 名前付きパイプ
- NetBIOS
- TCP/IP

以下も参照

186ページの『CATALOG APPC NODE』

189ページの『CATALOG APPCLU NODE』

191ページの『CATALOG APPN NODE』

204ページの『CATALOG IPX/SPX NODE』

215ページの『CATALOG LOCAL NODE』

218ページの『CATALOG NAMED PIPE NODE』

221ページの『CATALOG NETBIOS NODE』

226ページの『CATALOG TCP/IP NODE』

LIST NODEGROUPS

LIST NODEGROUPS

現行データベースに関連付けられているすべてのノード・グループのリストを表示します。

効力範囲

このコマンドは、`$HOME/sqllib/db2nodes.cfg` の中に指定されている任意のノードから発行できます。これらのどのノードからでも同一の情報が戻されません。

許可

システム・カタログ `SYSCAT.NODEGROUPS` および `SYSCAT.NODEGROUPDEF` の場合には、少なくとも以下の 1 つが必要です。

- `sysadm` または `dbadm` の権限
- CONTROL 特権
- SELECT 特権

必須接続

データベース

コマンド構文

```
▶▶ LIST NODEGROUPS [SHOW DETAIL] ▶▶
```

コマンド・パラメーター

SHOW DETAIL

以下の情報を出力に含めることを指定します。

- 区分化マップ ID
- ノード番号
- 使用中フラグ

例

以下に示すのは、LIST NODEGROUPS コマンドの出力例です。

```

NODEGROUP NAME
-----
IBMCATGROUP
IBMDEFAULTGROUP
IBMTEMPGROUP
  3 record(s) selected.

```

以下に示すのは、LIST NODEGROUPS SHOW DETAIL コマンドの出力例です。

```

NODEGROUP NAME          PMAP_ID NODE NUMBER          IN_USE
-----
IBMCATGROUP              0                0 Y
IBMDEFAULTGROUP         1                0 Y
  2 record(s) selected.

```

各フィールドは、次のとおりです。

NODEGROUP NAME

ノード・グループの名前。この名前は、ノード・グループのノードごとに繰り返されます。

PMAP_ID

区分化マップの ID。この ID は、ノード・グループのノードごとに繰り返されます。

NODE NUMBER

ノード番号。

IN_USE

次の 4 つの値のいずれか。

- Y** そのノードは、ノード・グループによって使用中です。
- D** REDISTRIBUTE NODEGROUP 操作の結果として、そのノードはノード・グループから除去中です。操作が完了すると、そのノードは LIST NODEGROUPS のレポートに含まれなくなります。
- A** そのノードはノード・グループには追加されていますが、区分化マップには追加されていません。ノード・グループ中の表スペース用のコンテナは、このノードに追加されています。REDISTRIBUTE NODEGROUP 操作が正常に終了した場合、値は Y に変わります。
- T** そのノードはノード・グループには追加されていますが、区分化マップには追加されていません。ノード・グループ中の表スペース用のコンテナは、このノードに追加されていません。表スペース・コンテナはノード・グループの表スペースごと

LIST NODEGROUPS

に、新しいノードに追加する必要があります。コンテナが正常に追加されると、値は A に変わります。

詳細については、*SQL 解説書* を参照してください。

以下も参照

505ページの『REDISTRIBUTE NODEGROUP』

LIST NODES

現行データベースに関連付けられているすべてのノードのリストを表示します。

効力範囲

このコマンドは、`$HOME/sql1lib/db2nodes.cfg` の中に指定されている任意のノードから発行できます。これらのどのノードからでも同一の情報が戻されます。

許可

なし

必須接続

データベース

コマンド構文

▶▶—LIST NODES—————▶▶

コマンド・パラメーター

なし

例

以下に示すのは、LIST NODES コマンドの出力例です。

```
NODE NUMBER
-----
      0
      2
      5
      7
      9
      5 record(s) selected.
```

以下も参照

505ページの『REDISTRIBUTE NODEGROUP』

LIST ODBC DATA SOURCES

LIST ODBC DATA SOURCES

使用可能なユーザーまたはシステム ODBC データ・ソースのすべてのリストを表示します。

ODBC (オープン・データベース・コネクティビティー) でのデータ・ソースという語は、指定したデータベースまたはファイル・システムのユーザー定義名のことです。この名前は、ODBC API を介してデータベースまたはファイル・システムにアクセスするときに使用されます。Windows NT および Windows 95 では、ユーザー・データ・ソースまたはシステム・データ・ソースのどちらであってもカタログ化できます。ユーザー・データ・ソースはそれをカタログ化したユーザーにのみ可視になりますが、システム・データ・ソースは他のすべてのユーザーから可視であり使用可能です。

このコマンドは、Windows NT、Windows 95、および Windows 3.1 でのみ使用可能です。

許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

USER ユーザー ODBC データ・ソースのみリスト表示します。キーワードを指定しない場合、これが省略時値です。

SYSTEM

システム ODBC データ・ソースのみリスト表示します。

例

以下に示すのは、LIST ODBC DATA SOURCES コマンドの出力例です。

LIST ODBC DATA SOURCES

User ODBC Data Sources	
Data source name	Description
SAMPLE	IBM DB2 ODBC DRIVER

以下も参照

224ページの『CATALOG ODBC DATA SOURCE』

597ページの『UNCATALOG ODBC DATA SOURCE』

LIST PACKAGES/TABLES

LIST PACKAGES/TABLES

現行データベースに関連付けられているパッケージまたは表のリストを表示します。

許可

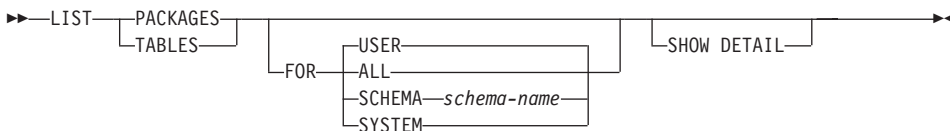
システム・カタログ SYSCAT.PACKAGES (LIST PACKAGES) と SYSCAT.TABLES (LIST TABLES) の場合には、少なくとも以下の 1 つが必要です。

- *sysadm* または *dbadm* の権限
- CONTROL 特権
- SELECT 特権

必須接続

データベース。暗黙接続が可能な場合には、省略時データベースへの接続が確立されます。

コマンド構文



コマンド・パラメーター

FOR FOR 文節を指定しないなら、USER のパッケージまたは表のリストが表示されます。

ALL データベース内のすべてのパッケージまたは表のリストが表示されます。

SCHEMA

指定されたスキーマのデータベース内のすべてのパッケージまたは表のリストのみ表示します。

SYSTEM

データベース内のすべてのパッケージまたは表のリストのみ表示します。

USER 現行ユーザーのデータベース内のすべてのユーザー・パッケージまたは表のリストを表示します。

SHOW DETAIL

表の完全名を表示します (LIST TABLES コマンドの場合のみ有効)。
このオプションを指定しなかった場合、名前は 30 文字で切り捨てられ、31 列目の ">" 記号が表名の切り捨て位置を表します。

例

次に示すのは LIST PACKAGES の出力例です。

Package	Schema	Bound by	Total sections	Valid	Format	Isolation level	Blocking
P1	SMITH	SMITH		1 Yes	0	CS	U

1 record(s) selected.

次に示すのは LIST TABLES の出力例です。

Table/View	Schema	Type	Creation time
DEPARTMENT	SMITH	T	1997-02-19-13.32.25.971890
EMP_ACT	SMITH	T	1997-02-19-13.32.27.851115
EMP_PHOTO	SMITH	T	1997-02-19-13.32.29.953624
EMP_RESUME	SMITH	T	1997-02-19-13.32.37.837433
EMPLOYEE	SMITH	T	1997-02-19-13.32.26.348245
ORG	SMITH	T	1997-02-19-13.32.24.478021
PROJECT	SMITH	T	1997-02-19-13.32.29.300304
SALES	SMITH	T	1997-02-19-13.32.42.973739
STAFF	SMITH	T	1997-02-19-13.32.25.156337

9 record(s) selected.

使用上の注意

LIST PACKAGES コマンドと LIST TABLES コマンドは、システム表への簡単なインターフェースを提供します。

以下の SELECT ステートメントは、システム表で検出した情報を戻します。このステートメントを拡張して、システム表が提供するその他の情報を選択することもできます。SELECT ステートメントを拡張するには、*SQL 解説書* および *管理の手引き* にあるカタログ視点の説明を使用してください。

```
select tabname, tabschema, type, create_time
from syscat.tables
order by tabschema, tabname;
```

```
select pkgname, pkgschema, boundby, total_sect,
       valid, format, isolation, blocking
from syscat.packages
```

LIST PACKAGES/TABLES

```
order by pkgschema, pkgname;

select tabname, tabschema, type, create_time
from syscat.tables
where tabschema = 'SYSCAT'
order by tabschema, tabname;

select pkgname, pkgschema, boundby, total_sect,
       valid, format, isolation, blocking
from syscat.packages
where pkgschema = 'NULLID'
order by pkgschema, pkgname;

select tabname, tabschema, type, create_time
from syscat.tables
where tabschema = USER
order by tabschema, tabname;

select pkgname, pkgschema, boundby, total_sect,
       valid, format, isolation, blocking
from syscat.packages
where pkgschema = USER
order by pkgschema, pkgname;
```

LIST TABLESPACE CONTAINERS

指定した表スペースのコンテナーのリストを表示します。

効力範囲

このコマンドは、それが実行されたノードに対してだけ情報を戻します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

必須接続

データベース

コマンド構文

```
▶▶—LIST TABLESPACE CONTAINERS FOR—tablespace-id—┬──────────────────┬──────────────────▶▶
└──────────┬──────────┘
             └──SHOW DETAIL──┘
```

コマンド・パラメーター

FOR *tablespace-id*

現行データベースで使用する表スペースを表す固有の整数。現行データベースで使用するすべての表スペースのリストを表示するには、405ページの『LIST TABLESPACES』を使います。

SHOW DETAIL

このオプションを指定しない場合、各コンテナーごとに以下の基本情報だけが表示されます。

- コンテナー ID
- 名前
- タイプ (ファイル、ディスク、またはパス)

このオプションを指定した場合は、各コンテナーに関して下記の付加的な情報が表示されます。

- ページの合計数
- 使用可能なページの数

LIST TABLESPACE CONTAINERS

- アクセス可能性 (yes または no)

例

次に示すのは LIST TABLESPACE CONTAINERS の出力例です。

Tablespace Containers for Tablespace 0

```
Container ID          = 0
Name                  = /home/smith/smith/NODE0000/
                      SQL00001/SQLT0000.0
Type                  = Path
```

次に示すのは、SHOW DETAIL を指定した場合の LIST TABLESPACE CONTAINERS の出力例です。

Tablespace Containers for Tablespace 0

```
Container ID          = 0
Name                  = /home/smith/smith/NODE0000/
                      SQL00001/SQLT0000.0
Type                  = Path
Total pages           = 895
Useable pages         = 895
Accessible            = Yes
```

以下も参照

405ページの『LIST TABLESPACES』

LIST TABLESPACES

現行データベースの表スペースのリストを表示します。

効力範囲

このコマンドは、それが実行されたノードに対してだけ情報を戻します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- *load*

必須接続

データベース

コマンド構文

```

▶▶—LIST TABLESPACES—┐
                        └—SHOW DETAIL—┘
  
```

コマンド・パラメーター

SHOW DETAIL

このオプションを指定しない場合、各表スペースごとに以下の基本情報だけが表示されます。

- 表スペース ID
- 名前
- タイプ (システム管理スペースまたはデータベース管理スペース)
- 内容 (すべてのデータ、長形式データのみ、または一時データ)
- 状態。現在の表スペースの状態を示す 16 進値。外から見える表スペースの状態は、特定の状態値の 16 進値を合計したものです。たとえば、状態が "quiesced: EXCLUSIVE" かつ "Load pending" の場合、その値は 0x0004 + 0x0008、つまり 0x000c となります。

LIST TABLESPACES

112ページの『db2tbst - 表スペース状態の獲得』を使うと、特定の16進値と関連した表スペース状態を取得できます。以下は、`sqlutil.h` に示されているビット定義です。

0x0	Normal
0x1	Quiesced: SHARE
0x2	Quiesced: UPDATE
0x4	Quiesced: EXCLUSIVE
0x8	Load pending
0x10	Delete pending
0x20	Backup pending
0x40	Roll forward in progress
0x80	Roll forward pending
0x100	Restore pending
0x100	Recovery pending (not used)
0x200	Disable pending
0x400	Reorg in progress
0x800	Backup in progress
0x1000	Storage must be defined
0x2000	Restore in progress
0x4000	Offline and not accessible
0x8000	Drop pending
0x2000000	Storage may be defined
0x4000000	StorDef is in 'final' state
0x8000000	StorDef was changed prior to rollforward
0x10000000	DMS rebalancer is active
0x20000000	TBS deletion in progress
0x40000000	TBS creation in progress
0x8	For service use only

このオプションを指定した場合は、各表スペースに関して下記の付加的な情報が表示されます。

- ページの合計数
- 使用可能なページの数
- 使用されたページの数
- 未使用ページの数
- 限界値 (ページ単位)
- ページ・サイズ (バイト単位)
- エクステンツ・サイズ (ページ単位)
- プリフェッチ・サイズ (ページ単位)
- コンテナーの数
- 最小回復時間 (0 以外の場合のみ表示)
- 状態変更表スペース ID (表スペース状態が "load pending" または "delete pending" の場合のみ表示)

- 状態変更オブジェクト ID (表スペース状態が "load pending" または "delete pending" の場合のみ表示)
- 静止者の数 (表スペース状態が "quiesced: SHARE"、"quiesced: UPDATE"、または "quiesced: EXCLUSIVE" の場合のみ表示)
- 各静止者ごとに表スペース ID とオブジェクト ID (静止者の数が 0 より大きい場合のみ表示)

例

下記に示すのは、LIST TABLESPACES SHOW DETAIL の 2 つの出力例です。

```

          Tablespaces for Current Database
Tablespace ID          = 0
Name                   = SYSCATSPACE
Type                   = System managed space
Contents               = Any data
State                  = 0x0000
  Detailed explanation:
    Normal
Total pages            = 895
Useable pages          = 895
Used pages             = 895
Free pages             = Not applicable
High water mark (pages) = Not applicable
Page size (bytes)     = 4096
Extent size (pages)   = 32
Prefetch size (pages) = 32
Number of containers   = 1
Tablespace ID          = 1
Name                   = TEMPSPACE1
Type                   = System managed space
Contents               = Temporary data
State                  = 0x0000
  Detailed explanation:
    Normal
Total pages            = 1
Useable pages          = 1
Used pages             = 1
Free pages             = Not applicable
High water mark (pages) = Not applicable
Page size (bytes)     = 4096
Extent size (pages)   = 32
Prefetch size (pages) = 32
Number of containers   = 1
Tablespace ID          = 2
Name                   = USERSPACE1
Type                   = System managed space
Contents               = Any data
State                  = 0x000c
  Detailed explanation:

```

LIST TABLESPACES

```
    Quiesced: EXCLUSIVE
    Load pending
Total pages                = 337
Useable pages              = 337
Used pages                 = 337
Free pages                 = Not applicable
High water mark (pages)   = Not applicable
Page size (bytes)         = 4096
Extent size (pages)       = 32
Prefetch size (pages)     = 32
Number of containers      = 1
State change tablespace ID = 2
State change object ID    = 3
Number of quiescers       = 1
    Quiescer 1:
        Tablespace ID      = 2
        Object ID          = 3
```

DB21011I In a partitioned database server environment, only the table spaces on the current node are listed.

```
    Tablespaces for Current Database
Tablespace ID              = 0
Name                       = SYSCATSPACE
Type                       = System managed space
Contents                   = Any data
State                      = 0x0000
    Detailed explanation:
        Normal
Total pages                = 1200
Useable pages              = 1200
Used pages                 = 1200
Free pages                 = Not applicable
High water mark (pages)   = Not applicable
Page size (bytes)         = 4096
Extent size (pages)       = 32
Prefetch size (pages)     = 32
Number of containers      = 1
Tablespace ID              = 1
Name                       = TEMPSPACE1
Type                       = System managed space
Contents                   = Temporary data
State                      = 0x0000
    Detailed explanation:
        Normal
Total pages                = 1
Useable pages              = 1
Used pages                 = 1
Free pages                 = Not applicable
High water mark (pages)   = Not applicable
Page size (bytes)         = 4096
Extent size (pages)       = 32
Prefetch size (pages)     = 32
Number of containers      = 1
Tablespace ID              = 2
Name                       = USERSPACE1
```

```

Type = System managed space
Contents = Any data
State = 0x0000
  Detailed explanation:
    Normal
Total pages = 1
Useable pages = 1
Used pages = 1
Free pages = Not applicable
High water mark (pages) = Not applicable
Page size (bytes) = 4096
Extent size (pages) = 32
Prefetch size (pages) = 32
Number of containers = 1
Tablespace ID = 3
Name = DMS8K
Type = Database managed space
Contents = Any data
State = 0x0000
  Detailed explanation:
    Normal
Total pages = 2000
Useable pages = 1952
Used pages = 96
Free pages = 1856
High water mark (pages) = 96
Page size (bytes) = 8192
Extent size (pages) = 32
Prefetch size (pages) = 32
Number of containers = 2
Tablespace ID = 4
Name = TEMP8K
Type = System managed space
Contents = Temporary data
State = 0x0000
  Detailed explanation:
    Normal
Total pages = 1
Useable pages = 1
Used pages = 1
Free pages = Not applicable
High water mark (pages) = Not applicable
Page size (bytes) = 8192
Extent size (pages) = 32
Prefetch size (pages) = 32
Number of containers = 1
DB21011I In a partitioned database server environment, only the table spaces
on the current node are listed.

```

使用上の注意

マルチノード環境では、このコマンドがデータベースのすべての表スペースを戻すわけではありません。すべての表スペースのリストを表示するには、SYSCAT.SYSTABLESPACES を照会します。

LIST TABLESPACES

表スペースのバランス調整操作中には、使用可能ページ数には新しく追加されたコンテナのページ数が含まれますが、バランス調整完了までの間、それらの新しいページは、未使用ページ数に反映されません。表スペースのバランス調整が実行されていない場合、使用されたページの数と未使用ページ数を合計すると、使用可能ページ数の値に等しくなります。

表スペースの詳細については、[管理の手引き](#) を参照してください。

以下も参照

112ページの『[db2tbst - 表スペース状態の獲得](#)』

403ページの『[LIST TABLESPACE CONTAINERS](#)』

LOAD

データを DB2 表にロードします。サーバー上に存在するデータは、ファイル、テープ、または名前付きパイプの形式にすることができます。リモートに接続されたクライアント上に存在するデータは、完全修飾ファイルまたは名前付きパイプの形式にすることができます。テープは OS/2 ではサポートされていません。ロード・ユーティリティーでは、階層レベルのデータのロードはサポートされていません。

効力範囲

このコマンドは、直接接続が存在している区画にのみ影響を及ぼします。ロード・ユーティリティーは単一のデータベース区画に対してのみ機能します。

リモートに接続されたクライアント上に存在するデータのロードは、以下の条件下ではサポートされません。

- クライアントが接続するデータベースが DB2 エンタープライズ拡張エディション環境にある場合。
- クライアントが接続するデータベースが、すでにカタログ化されたデータベースに対してカタログ化される場合。

許可

以下のどれかが必要です。

- *sysadm*
- *dbadm*
- データベースに対するロード権限、および以下の特権。
 - ロード・ユーティリティーが INSERT モード、 TERMINATE モード (それまでのロード挿入操作を終了する)、または RESTART モード (以前のロード挿入操作を再開する) で呼び出された場合には、その表に対する INSERT 特権。
 - ロード・ユーティリティーが REPLACE モード、 TERMINATE モード (それまでのロード置換操作を終了する)、または RESTART モード (以前のロード置換操作を再開する) で呼び出された場合には、その表に対する INSERT および DELETE 特権。
 - 例外表がロード操作の一部として使用される場合、その例外表に対する INSERT 特権。

注: すべてのロード・プロセス (および一般にすべての DB2 サーバー・プロセス) はインスタンス所有者によって所有されており、それらのプロセスすべてにおいて、必要なファイルにアクセスするためにそのインスタンス

LOAD

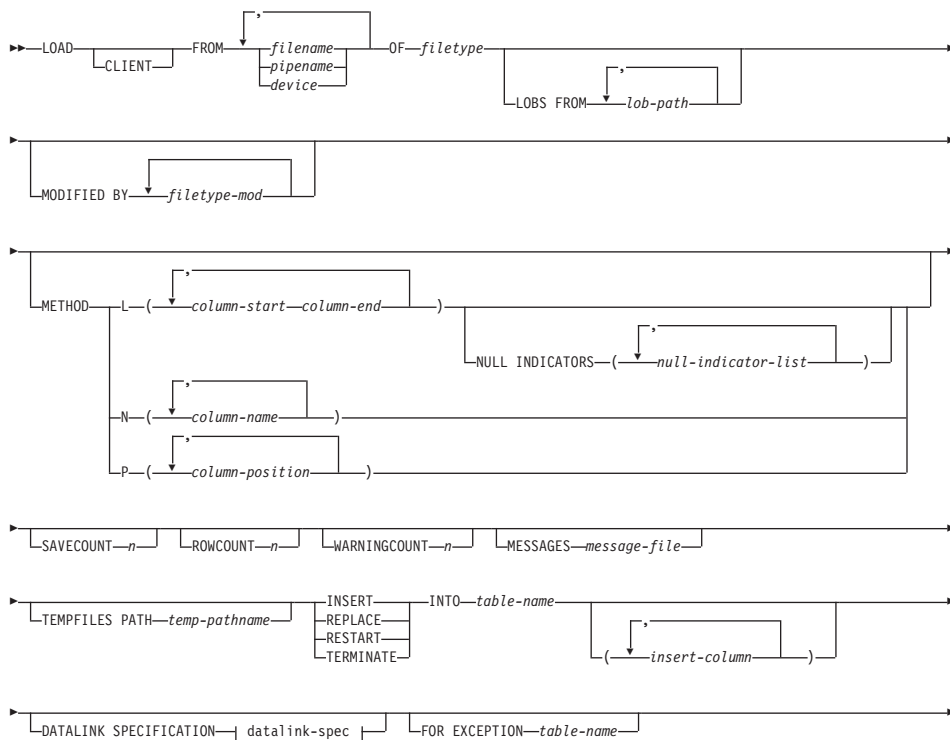
所有者の ID を使うため、インスタンス所有者には入力データ・ファイルに対する読み取りアクセス権が必要です。このコマンドをだれが呼び出すかには関係なく、それらの入力データ・ファイルをインスタンス所有者から読むことができなければなりません。

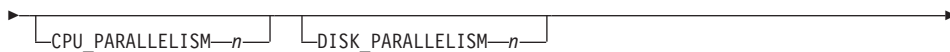
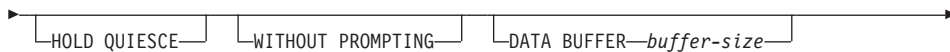
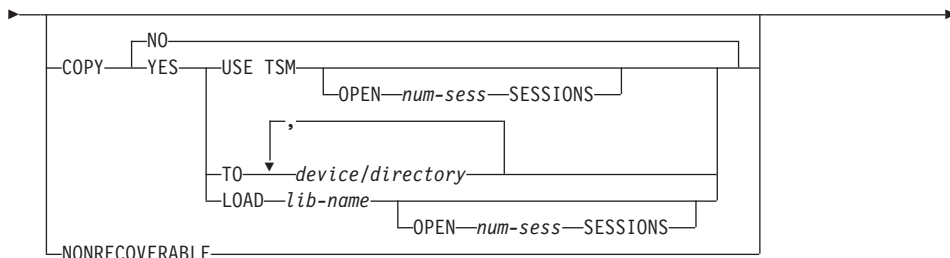
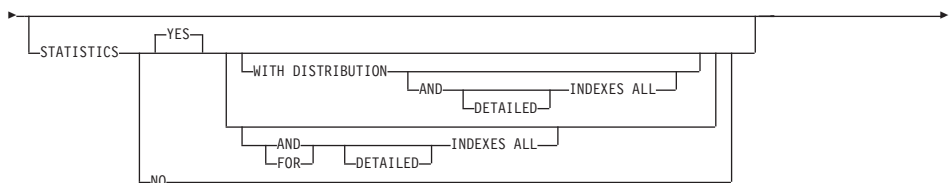
必須接続

データベース。暗黙接続が可能な場合には、省略時データベースへの接続が確立されます。

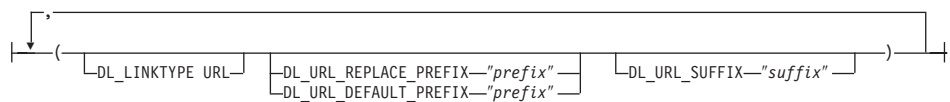
インスタンス。明示的な接続は必要ありません。データベースへの接続が確立している場合には、ローカル・インスタンスへの暗黙接続が試行されます。

コマンド構文





datalink-spec:



コマンド・パラメーター

CLIENT

ロードするデータが、リモートに接続するクライアントにあることを指定します。ロード操作がリモート・クライアントから呼び出されない場合、このオプションは無視されます。

LOAD

注:

1. リモートに接続されたクライアント上に存在するデータのロードは、以下の条件下ではサポートされません。
 - クライアントが接続するデータベースが DB2 エンタープライズ拡張エディション環境にある場合。
 - クライアントが接続するデータベースが、すでにカタログ化されたデータベースに対してカタログ化される場合。
2. DUMPFIL および LOBSINFIL 修飾子 (437ページの表7 を参照) は、CLIENT キーワードが指定されている場合でも、サーバー上のファイルを参照します。
3. コード・ページ変換は、リモートのロード操作時には実行されません。データのコード・ページがサーバーのコード・ページとは異なる場合、データのコード・ページは CODEPAGE 修飾子を使用して指定する必要があります (437ページの表7 を参照)。
4. リモート・クライアントのデータを (名前付きパイプではなく) ファイルからロードする場合、1 ファイルにつき 2GB の上限があります。

以下の例では、リモートに接続されたクライアント上に存在するデータ・ファイル (/u/user/data.del) は、サーバー・データベース上の MYTABLE にロードされます。

```
db2 load client from /u/user/data.del of del
      modified by codepage=850 insert into mytable
```

COPY NO

順方向回復が使用可能 (つまり、*logretain* または *userexit* がオン) になっていれば、表が存在している表スペースをバックアップ保留状態にするよう指定します。表スペースのバックアップまたはデータベースの完全バックアップを実行しない限り、表スペースのどの表のデータも更新または削除できません。ただし、SELECT ステートメントを使用すれば、どの表のデータにもアクセス可能です。

COPY YES

ロードするデータのコピーを保存することを指定します。順方向回復が使用禁止 (つまり *logretain* と *userexit* が両方ともオフ) であれば、このオプションは無効です。このオプションは DATALINK 列を含む表ではサポートされません。

USE TSM

Tivoli Storage Manager (TSM) を使ってコピーを保管することを指定します。

OPEN num-sess SESSIONS

TSM またはベンダー製品とともに使用する入出力セッションの数です。省略時値は 1 です。

TO device/directory

コピー・イメージを作成する先の装置またはディレクトリーを指定します。OS/2 では、テープはサポートされません。

SCO UnixWare 7 で実行中の DB2 サーバーの場合、テープへのコピーはサポートされません。

LOAD lib-name

使用するバックアップおよび復元 I/O 関数を含む共用ライブラリー (OS/2 または Windows オペレーティング・システムでは DLL) の名前。フルパスで指定することができます。フルパスを指定しない場合、省略時でユーザー・出口プログラムの存在するパスになります。

CPU_PARALLELISM n

表オブジェクトの作成時に、レコードの解析、変換、および形式設定のためにロード・ユーティリティーが spawn するプロセスまたはスレッドの数を指定します。このパラメーターは、区画内並列性を活用するために設計されています。これは、事前にソートされているデータのロード時に特に便利です。なぜなら、ソース・データ中のレコード順序が保存されるからです。このパラメーターの値が 0 の場合や、このパラメーターを指定しなかった場合、ロード・ユーティリティーは、実行時にインテリジェントな省略時値 (通常は使用可能な CPU の数に基づく) を使用します。

注:

1. LOB または LONG VARCHAR フィールドのどちらかを含む表でこのパラメーターを使用する場合、システムの CPU の数またはユーザーが指定した値には関係なく、値は 1 になります。
2. SAVECOUNT パラメーターに指定する値が小さいと、データと表のメタデータの両方をフラッシュするために、ローダーがさらに多くの入出力操作を実行することになります。CPU_PARALLELISM が 1 より大きいなら、フラッシュ操作は非同期になり、ローダーは CPU を活用できます。CPU_PARALLELISM が 1 に設定されているなら、ローダーは一貫性ポイントにおいて IO を待ちます。CPU_PARALLELISM を 2、SAVECOUNT を 10 000 に設定したロード操作は、CPU が 1 つしかなくても、同じ操作で CPU_PARALLELISM を 1 に設定した場合より速く完了します。

DATA BUFFER buffer-size

ユーティリティー内でデータを転送するためのバッファー・スペースとして使用する 4KB ページ数を設定します (並列性の程度には依存しません)。指定する値がアルゴリズム上の最小値より小さい場合、最小限必要なリソースが使用され、警告は戻されません。

このメモリーは、ユーティリティー・ヒープから直接に割り当てられ、そのサイズは `util_heap_sz` データベース構成パラメーターで修正可能です。

値を指定しない場合、ユーティリティーの実行時にインテリジェントな省略時値が計算されます。この省略時値は、表の特性だけでなく、ローダーのインスタンス生成時にユーティリティー・ヒープ中で使用可能な空きスペースの割合に基づいています。

DATALINK SPECIFICATION

各 DATALINK 列ごとに、それぞれ 1 つの列指定を括弧で囲んで指定できます。各列指定は、1 つまたは複数の `DL_LINKTYPE`、接頭部、および `DL_URL_SUFFIX` 指定で構成されます。接頭部指定は、`DL_URL_REPLACE_PREFIX` または `DL_URL_DEFAULT_PREFIX` のどちらかになります。

DATALINK 列指定の数は、表で定義されている DATALINK の数と同じだけ定義できます。指定の順序は、挿入列 リストの中での DATALINK 列の順序、または表定義内での順序 (挿入列 リストが指定されていない場合) に従います。

DISK_PARALLELISM n

表スペース・コンテナにデータを書き込むためにロード・ユーティリティーが生成するプロセスまたはスレッドの数を指定します。値を指定しない場合、ユーティリティーは表スペース・コンテナの数と表の特性に基づいて、インテリジェントな省略時値を選択します。

DL_LINKTYPE

指定した場合は、列定義の `LINKTYPE` に一致していなければなりません。そうすることによって、列定義に `LINKTYPE URL` が指定されている場合に `DL_LINKTYPE URL` が受け入れ可能になります。

DL_URL_DEFAULT_PREFIX "prefix"

これを指定すると、同じ列内のすべての DATALINK 値の省略時接頭部になります。ここでいう接頭部とは、URL 指定の「スキーム・ホスト・ポート」部分のことです。(分散ファイル・システム (DFS) の場合、接頭部とは URL 指定の「スキーム・セル名とファイル・スペースの接合」部分のことです。)

接頭部の例

```
"http://server"
"file://server"
"file:"
"http://server:80"
"dfs://.../cellname/fs"
```

列データの中に接頭部がない場合、DL_URL_DEFAULT_PREFIX で省略時の接頭部が指定されているなら、列の値の接頭部としてその省略時接頭部が付けられます (NULL でない場合)。

たとえば、DL_URL_DEFAULT_PREFIX で省略時接頭部が "http://toronto" として指定されている場合、

- 列入力値 "/x/y/z" は "http://toronto/x/y/z" として保管されます。
- 列入力値 "http://coyote/a/b/c" は "http://coyote/a/b/c" として保管されます。
- 列入力値 NULL は NULL として保管されます。

DL_URL_REPLACE_PREFIX "prefix"

この文節は、それ以前にエクスポート・ユーティリティによって生成されたデータをロードまたはインポートする際に、ユーザーがデータに含まれるホスト名を別のホスト名に一括置換したい場合に便利です。指定する場合には、それがすべての非 NULL 列値の接頭部になります。列値にすでに接頭部があるなら、それは置き換えられます。列値に接頭部がないなら、DL_URL_REPLACE_PREFIX で指定される接頭部がその列値の接頭部になります。分散ファイル・システム (DFS) の場合、接頭部とは URL 指定の「スキーム・セル名とファイル・スペースの接合」部分のことです。

たとえば、DL_URL_REPLACE_PREFIX で接頭部が "http://toronto" として指定されている場合、

- 列入力値 "/x/y/z" は "http://toronto/x/y/z" として保管されます。
- 列入力値 "http://coyote/a/b/c" は "http://toronto/a/b/c" として保管されます。"coyote" は "toronto" に置き換えられます。
- 列入力値 NULL は NULL として保管されます。

DL_URL_SUFFIX "suffix"

これを指定すると、それはその列のすべての非 NULL 列値に付加されます。これは実際には、DATALINK 値のデータ・ロケーション部分の『パス』構成要素に付加されます。

FOR EXCEPTION table-name

エラーが発生した行のコピー先となる例外表を指定します。固有索引または 1 次キー索引に違反した行はすべてコピーされます。

DATALINK 例外も例外表に取り込まれます。未修飾の表名を指定すると、その表は CURRENT SCHEMA で修飾されます。

例外表に書き込まれる情報は、ダンプ・ファイルには書き込まれません (dumpfile 修飾子については、437ページの表7を参照)。区分データベース環境では、例外表は、ロードする表が定義されたノードについて定義する必要があります。一方ダンプ・ファイルは、無効であるか構文エラーであるためにロードできない行が含まれます。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。

FROM filename/pipename/device

ロードするデータが含まれているファイル、パイプ、または装置を指定します。このファイル、パイプ、装置は、CLIENT オプションが指定されていなければ、データベースが存在するノードになければなりません。複数の名前を指定すると、それらは順番に処理されます。最後に指定した項目がテープ装置の場合は、別のテープを使うようユーザーに対してプロンプトが出ます。有効な応答オプションは、次のとおりです。

- c** 続行。警告メッセージを生成した装置の使用を続けます (たとえば新しいテープをマウントした場合など)。
- d** 装置の終了。警告メッセージを生成した装置の使用を停止します (たとえば、それ以上テープがない場合)。
- t** 終了。すべての装置を終了します。

注:

1. テープは OS/2 ではサポートされていません。
2. 可能なかぎり完全修飾ファイル名を使用してください。リモート・サーバーの場合は、常に完全修飾ファイル名を使う必要があります。呼び出し側と同じノードにデータベースが存在する場合には、相対パスを使用することもできます。
3. ファイルが物理的には分割されてはいるが論理的には 1 つのファイルである場合には、複数の IXF ファイルからのデータのロードがサポートされています。ファイルが論理的にも物理的にも分割されている場合は、サポートされていません。

4. OS/2 においてロードするデータとして予期される量より小さい値の *pipename* を指定した場合には、システム・リソースの終結処理をしてから (できれば IPL を実行)、LOAD コマンドをもう一度発行する必要があります。
5. クライアント・マシン上に存在するデータをロードする場合、そのデータは、完全修飾ファイルまたは名前付きパイプのいずれかの形式でなければなりません。

HOLD QUIESCE

ロード操作の後、ユーティリティーが表を静止排他状態のままにしておくことを指定します。表スペースの静止状態を終了するには、次のコマンドを発行します。

```
db2 quiesce tablespaces for table <tablename> reset
```

注: ファントム静止 が作成されないようにしてください (491ページの『QUIESCE TABLESPACES FOR TABLE』を参照)。

INDEXING MODE

ロード・ユーティリティーが索引を再作成するのか、それとも索引を増分で拡張するのかを指定します。有効な値は以下のとおりです。

AUTOSELECT

REBUILD モードと INCREMENTAL モードのどちらにするかを、ロード・ユーティリティーが自動的に決定します。

REBUILD

すべての索引が再作成されます。古い表データの索引キー部分も、追加される新しい表データの索引キー部分もすべてソートできるようにするため、ロード・ユーティリティーには十分なリソースが必要となります。

INCREMENTAL

索引に新しいデータが取り込まれて拡張します。このアプローチでは、索引の空きスペースが消費されます。このアプローチでは、新たに挿入されるレコードの索引キーを追加するためのソート・スペースがあれば十分です。この方式がサポートされるのは、索引オブジェクトが有効で、かつロード操作の開始時にアクセス可能な場合だけです。(たとえば、DEFERRED モードが指定されたロード操作の直後では、この方式は無効です)。このモードを指定したものの、索引の状態などの理由でサポートされない場合は、警告が戻され、REBUILD モードでロ

LOAD

ード操作が続行されます。同様に、ロード作成フェーズでロード再開操作を開始した場合も、INCREMENTAL モードはサポートされません。

以下の条件がすべて真の場合、増分索引の作成はサポートされません。

- LOAD COPY オプションが指定されている (*logretain* または *userexit* が使用可能である)。
- 表が DMS 表スペース内に存在している。
- 索引オブジェクトの存在している表スペースが、ロードしようとしている表に属する他の表オブジェクトによって共有されている。

この制限をう回するため、索引は別々の表スペースに置くようお勧めします。

DEFERRED

このモードが指定されている場合、ロード・ユーティリティーは索引の作成を試みません。最新表示が必要であることを示すマークが索引に付けられます。ロード操作とは関係のないこのような索引に最初にアクセスするときは、再作成が強制的に実行されたり (詳しくは、*管理の手引き* を参照)、データベースの再始動時に索引が再作成されたりする場合があります。このアプローチでは、最も大きい索引のキー部分をすべて処理できるだけのソート・スペースが必要です。索引を作成するためにその後かかる合計時間は、REBUILD モードの場合よりも長くなります。したがって、この索引作成据え置きモードで複数のロード操作を実行する場合、最初の非ロード・アクセス時に索引を再作成できるようにしておくよりも、順序列内の最後のロード操作で索引の再作成を実行できるようにした方が (パフォーマンスの観点から) 賢明であるといえます。

据え置き索引作成がサポートされるのは、非固有の索引があるために、ロード・フェーズで挿入される複写キーがロード操作後は永続的ではなくなる表だけです。

注: 据え置き索引作成は、DATALINK 列がある表ではサポートされません。

INSERT

ロード・ユーティリティを実行できる 4 つのモードのうちの 1 つ。既存の表データを変更することなく、ロードしたデータを表に追加します。

insert-column

データの挿入先となる表列を指定します。

ロード・ユーティリティは、1 つまたは複数のスペースを含んだ名前の列を解析できません。たとえば、次のように指定します。

```
db2 load from delfile1 of del modified by noeofchar noheader
method P (1, 2, 3, 4, 5, 6, 7, 8, 9)
insert into table1 (BLOB1, S2, I3, Int 4, I5, I6, DT7, I8, TM9)
```

は、Int 4 列があるためエラーになります。これは、次のようにして二重引用符で列名を囲むことによって解決できます。

```
db2 load from delfile1 of del modified by noeofchar noheader
method P (1, 2, 3, 4, 5, 6, 7, 8, 9)
insert into table1 (BLOB1, S2, I3, "Int 4", I5, I6, DT7, I8, TM9)
```

INTO table-name

データのロード先となるデータベース表を指定します。この表として、システム表または宣言一時表は指定できません。別名、完全修飾、または未修飾の表名は指定できます。修飾子付き表名は、*schema.tablename* の形式です。未修飾の表名を指定すると、その表は CURRENT SCHEMA で修飾されます。

LOBS FROM lob-path

ロードする LOB 値が含まれているデータ・ファイルへのパス。パスの最後は斜線 (/) でなければなりません。CLIENT オプションを指定した場合、パスは完全修飾しなければなりません。LOB データ・ファイルの名前は、メイン・データ・ファイル (ASC、DEL、または IXF) の、LOB 列にロードされる列内に保管されます。*filetype-mod* ストリング内に *lobsinfile* が指定されていない場合、このオプションは無視されます (437 ページの表 7 を参照)。

MESSAGES message-file

ロード操作中に生じ得る警告およびエラー・メッセージの宛先を指定します。メッセージ・ファイルを指定しなかった場合、メッセージは標準出力に書き込まれます。このファイルへの完全パスが指定されていない場合、ロード・ユーティリティは現行のディレクトリーおよび省略時のドライブを宛先として使用します。すでに存在するファイル名を指定すると、ロード時に情報が追加されます。

LOAD

通常、メッセージ・ファイルには、ロード操作の終了時にメッセージが入れられますが、それ自体は操作の進行状況のモニターには適していません。ロード操作のリアルタイムのモニターについては、453ページの『LOAD QUERY』を参照してください。

METHOD

L データのロードを開始する列および終了する列の番号を指定します。列の番号は、データの行の先頭からのバイト単位のオフセットです。この番号は 1 から始まります。

注: このメソッドは、ASC ファイルの場合にのみ使用することができ、そのファイル・タイプに対してのみ有効なオプションです。

N ロードするデータ・ファイルの中の列の名前を指定します。これらの列名の太文字小文字は、システム・カタログ中の対応する名前の太文字小文字と一致している必要があります。ヌル可能ではない各表列には、METHOD N リスト内に対応する項目が必要です。たとえば、データ・フィールドが F1、F2、F3、F4、F5、および F6 であり、表列が C1 INT、C2 INT NOT NULL、C3 INT NOT NULL、および C4 INT の場合、method N (F2, F1, F4, F3) は有効な要求ですが、method N (F2, F1) は無効です。

注: この方式は、IXF ファイルの場合にのみ使用することができます。

P ロードする入力データ・フィールドの索引 (1 から始まる) を指定します。ヌル可能ではない各表列には、METHOD P リスト内に対応する項目が必要です。たとえば、データ・フィールドが F1、F2、F3、F4、F5、および F6 であり、表列が C1 INT、C2 INT NOT NULL、C3 INT NOT NULL、および C4 INT の場合、method P (2, 1, 4, 3) は有効な要求ですが、method P (2, 1) は無効です。

注: この方式は、IXF または DEL ファイルの場合にのみ使用でき、DEL ファイル・タイプに対してのみ有効なオプションです。

MODIFIED BY filetype-mod

追加オプションを指定します (437ページの表7 を参照)。

NONRECOVERABLE

ロード・トランザクションが回復不能としてマークされており、それ以降のロールフォワード・アクションによってそれを回復させることは不可能であることを指定します。ロールフォワード・ユーティリティーは、そのトランザクションをスキップし、データのロード先の表に "invalid" (無効) としてマークします。ユーティリティーは、その表に対するそれ以降のどのトランザクションをも無視します。ロールフォワード操作が完了すると、そのような表は、除去するか、または回復不能なロード操作完了後のコミット・ポイントの後に取られたバックアップ (全バックアップまたは表スペースのバックアップ) から復元することしかできません。

このオプションを指定した場合、ロード操作の後、表スペースはバックアップ保留状態になりません。また、ロード操作時にロードされたデータのコピーを作成する必要はありません。

FILE LINK CONTROL が指定された DATALINK 列が表に存在している場合や、そのような列を表に追加しようとしている場合には、このオプションを使用しないでください。

NULL INDICATORS null-indicator-list

このオプションは、METHOD L パラメーターを指定した場合だけ使用できます (つまり、入力ファイルが ASC ファイルの場合)。ヌル標識リストは、コンマで区切られた正の整数のリストで、各ヌル標識フィールドの列の番号を指定します。列の番号は、データの行の先頭からのバイト単位の、各ヌル標識フィールドのオフセットです。ヌル標識リストには、METHOD L パラメーターで定義された各データ・フィールドに対する 1 つの項目がなければなりません。列の番号がゼロであることは、対応するデータ・フィールドが必ずデータを含んでいることを示します。

ヌル標識列中の Y の値は、その列データがヌル値であることを指定します。ヌル標識列に Y 以外の文字を指定した場合は、列データがヌル値ではなく、METHOD L オプションで指定された列データがロードされることを指定することになります。

ヌル標識文字は MODIFIED BY オプションを使用して変更できます (437ページの表7 の nullindchar 修飾子の説明を参照)。

OF filetype

入力ファイル内のデータの形式を指定します。

- ASC (区切りなし ASCII 形式)
- DEL (区切り付き ASCII 形式)

LOAD

- IXF (統合交換フォーマット、PC バージョン)。同一のあるいは別の DB2 表からエクスポートされたことを意味します。

ファイル形式の詳細については、データ移動ユーティリティー 手引きおよび解説書 の付録『Export/Import/Load ユーティリティー・ファイル形式』を参照してください。

REPLACE

ロード・ユーティリティーを実行できる 4 つのモードのうちの 1 つ。表内の既存のデータすべてを削除してから、ロードしたデータを挿入します。表定義および索引定義は変更されません。階層間でデータを移動する際にこのオプションを使用する場合は、階層全体に関係したデータだけが置き換えられます。副表は置き換えられません。

このオプションは DATALINK 列を含む表ではサポートされません。

RESTART

ロード・ユーティリティーを実行できる 4 つのモードのうちの 1 つ。以前に割り込みを受けたロード操作を再開します。ロード操作は、ロード、作成、または削除フェーズの最後の一貫性ポイントから自動的に続行されます。

RESTARTCOUNT

予約済み。

ROWCOUNT *n*

ロードするファイル内の物理レコードの数 *n* を指定します。ユーザーはファイル内の最初の *n* 個の行だけをロードできます。

SAVECOUNT *n*

ロード・ユーティリティーが *n* 行ごとに一貫性ポイントを取ることを指定します。この値はページ・カウントに変換され、エクステント・サイズの間隔に切り上げられます。メッセージが発行されるのは一貫性ポイントにおいてなので、453ページの『LOAD QUERY』を使用してロード操作をモニターする場合には、このオプションを選択する必要があります。*n* の値が十分な大きさでないなら、各一貫性ポイントで実行される活動の同期化によってパフォーマンスに影響してしまいます。

省略時値はゼロですが、それは、必要がなければ一貫性ポイントは確立されないことを意味します。

SORT BUFFER *buffer-size*

予約済み。

STATISTICS NO

統計データを収集せず、したがってカタログ内の統計データも変更しないことを指定します。これが省略時値です。

STATISTICS YES

表およびすべての既存索引の統計データを収集するよう指定します。このオプションがサポートされるのは、ロード操作が **REPLACE** モードの場合だけです。

WITH DISTRIBUTION

分散統計を収集するよう指定します。

AND INDEXES ALL

表統計と索引統計の両方を収集するよう指定します。

FOR INDEXES ALL

索引統計だけを収集するよう指定します。

DETAILED

拡張された索引統計を収集するよう指定します。

TEMPFILES PATH temp-pathname

ロード操作時に一時ファイルを作成する場合に使うパスの名前を指定します。これはサーバー・ノードに従って完全に修飾しなければなりません。

一時ファイルは、ファイル・システムのスペースを使用します。場合によっては、このスペースが相当必要になります。以下に示すのは、すべての一時ファイルにどの程度のファイル・システム・スペースを割り振るべきかの見積もりです。

- **DATALINK** 値を含む重複行または拒否行ごとに 4 バイト
- ロード・ユーティリティーが生成するメッセージごとに 136 バイト
- データ・ファイルに長フィールド・データまたは **LOB** が含まれている場合は、15KB のオーバーヘッド。 **INSERT** オプションを指定した場合で、表の中に多量の長フィールドまたは **LOB** データがすでに含まれている場合には、この数値はこれよりもかなり大きくなる場合があります。

一時ファイルの詳細については、 **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。

TERMINATE

ロード・ユーティリティーを実行できる 4 つのモードのうちの 1 つ。以前に割り込みを受けたロード操作を終了し、ロード操作が開始された時点まで操作をロールバックします。途中に一貫性ポイントがあっても

LOAD

通過します。操作に関係する表スペースの状態は正常に戻され、すべての表オブジェクトが一貫性のある状態に保たれます (索引の再作成が次のアクセスで自動的に行われることになっている場合は、索引オブジェクトは無効としてマークされます)。終了するロード操作がロード **REPLACE** の場合、その表はロード **TERMINATE** 操作完了後にヌルの表まで切り捨てられます。終了するロード操作がロード **INSERT** の場合、その表はロード **TERMINATE** 操作完了後も元のレコードをすべて保持します。

ロード終了オプションを使用しても、表スペースからバックアップ保留状態が取り除かれることはありません。

注: このオプションは **DATALINK** 列を含む表ではサポートされません。

USING directory

予約済み。

WARNINGCOUNT n

n 個の警告が出たらロード操作を停止します。このパラメーターは、警告が出ないことを期待しているが、正しいファイルと表が使われているかどうかを検査したい、という場合に設定してください。 n がゼロの場合、またはこのオプションが指定されていない場合、何度警告が出されてもロード操作は続行します。警告のしきい値に達したためにロード操作が停止された場合でも、あらためて **RESTART** モードでロード操作を開始できます。ロード操作は最後の一貫性ポイントから自動的に続行されます。または、入力ファイルの先頭から **REPLACE** モードであらためてロード操作を開始できます。

WITHOUT PROMPTING

データ・ファイルのリストにロードするすべてのファイルを含め、しかもリストに含まれる装置またはディレクトリーがロード操作全体で十分であるということ指定します。続きの入力ファイルが見つからなかったり、ロード操作が終了する前にコピー先がいっぱいになるとロード操作は失敗し、表はロード保留状態のままになります。

このオプションを指定しない場合に、テープ装置がコピー・イメージ用のテープの終わりに達した場合、またはリスト中の最後の項目がテープ装置であった場合は、ユーザーに対してその装置に新しいテープを装着するよう求めるプロンプトが出されます。テープは **OS/2** ではサポートされていません。

例

例 1

TABLE1 に次の 5 つの列があるとします。

- COL1 VARCHAR 20 NOT NULL WITH DEFAULT
- COL2 SMALLINT
- COL3 CHAR 4
- COL4 CHAR 2 NOT NULL WITH DEFAULT
- COL5 CHAR 2 NOT NULL

ASCFILE1 に次の 6 つの要素があるとします。

- ELE1 positions 01 to 20
- ELE2 positions 21 to 22
- ELE5 positions 23 to 23
- ELE3 positions 24 to 27
- ELE4 positions 28 to 31
- ELE6 positions 32 to 32
- ELE6 positions 33 to 40

データ・レコードは次のとおりです。

```

1...5....10...15...20...25...30...35...40
Test data 1          XXN 123abcdN
Test data 2 and 3   QQY   wxyzN
Test data 4,5 and 6 WWN6789   Y

```

次のコマンドは、ファイルから表をロードします。

```

db2 load from ascfile1 of asc modified by striptblanks reclen=40
method L (1 20, 21 22, 24 27, 28 31)
null indicators (0,0,23,32)
insert into table1 (col1, col5, col2, col3)

```

注:

1. MODIFIED BY パラメーターで striptblanks を指定すると、VARCHAR 列の中の空白が切り捨てられるようになります (たとえば行 1、2、および 3 の長さがそれぞれ 11、17、および 19 バイトである COL1)。
2. MODIFIED BY パラメーターで reclen=40 を指定すると、各入力レコードの最後が改行文字でなく、各レコードが 40 バイト長であることを指定することになります。最後の 8 バイトは、表のロードには使われません。

LOAD

3. COL4 は入力ファイルにはないので、その省略時値 (NOT NULL WITH DEFAULT と定義されています) を使って TABLE1 に挿入されます。
4. 位置 23 と 32 は、特定の行で TABLE1 の COL2 と COL3 が NULL としてロードされるかどうかを指示するために使われます。ある特定のレコードの、その列のヌル標識位置が Y である場合、その列は NULL になります。N なら、入力レコード中のその列のデータ位置のデータ値 (L(.....) で定義される) は、その行の列データのソースとして使用されます。この例では、行 1 のどの列も NULL ではなく、行 2 の COL2 は NULL であり、行 3 の COL3 は NULL です。
5. この例では、COL1 と COL5 の NULL INDICATORS は 0 (ゼロ) として指定されますが、それはそのデータをヌル値にできないことを示しています。
6. 特定の列の NULL INDICATOR は、入力レコード内のどこにでも指定できますが、その位置は指定する必要があり、かつ Y または N の値を指定しなければなりません。

例 2 (ファイルから LOB をロードする)

TABLE1 に次の 3 つの列があるとします。

- COL1 CHAR 4 NOT NULL WITH DEFAULT
- LOB1 LOB
- LOB2 LOB

ASCFILE1 には次の 3 つの要素があるとします。

- ELE1 positions 01 to 04
- ELE2 positions 06 to 13
- ELE3 positions 15 to 22

次に示すファイルは、 /u/user1 または /u/user1/bin のどちらかにあります。

- ASCFILE2 has LOB data
- ASCFILE3 has LOB data
- ASCFILE4 has LOB data
- ASCFILE5 has LOB data
- ASCFILE6 has LOB data
- ASCFILE7 has LOB data

ASCFILE1 内のデータ・レコード

```
1...5...10...15...20...25...30.
REC1 ASCFILE2 ASCFILE3
REC2 ASCFILE4 ASCFILE5
REC3 ASCFILE6 ASCFILE7
```

次のコマンドは、ファイルから表をロードします。

```
db2 load from ascfile1 of asc
      lobs from /u/user1, /u/user1/bin
      modified by lobsinfile reclen=22
      method L (1 4, 6 13, 15 22)
      insert into table1
```

注:

1. **MODIFIED BY** パラメーターの中で **lobsinfile** を指定すると、ファイルからすべての **LOB** データをロードすることをローダーに対して指定することになります。
2. **MODIFIED BY** パラメーターで **reclen=22** を指定すると、各入力レコードの最後が改行文字でなく、各レコードが 22 バイト長であることを指定することになります。
3. **LOB** データは、**ASCFILE2** から **ASCFILE7** までの 6 つのファイルに入っています。各ファイルには、特定の行の **LOB** 列をロードするのに使用されるデータが入れています。LOB と他のデータの関係は、**ASCFILE1** に指定します。このファイルの最初のレコードは、**REC1** を行 1 の **COL1** にするようローダーに指示します。行 1 の **LOB1** をロードするには **ASCFILE2** の内容が使われ、**ASCFILE3** の内容は行 1 の **LOB2** をロードするのに使われます。同じように、行 2 の **LOB1** および **LOB2** をロードするには **ASCFILE4** と **ASCFILE5** が使われ、行 3 の **LOB** をロードするには **ASCFILE6** と **ASCFILE7** が使われます。
4. これらのファイルがローダーで必要になった場合には、名前の指定された **LOB** ファイルを探索するのに使われる 2 つのパスが、**LOBS FROM** パラメーターに含まれています。
5. **lobsinfile** 修飾子を指定しないで **ASCFILE1** (区切りなしの ASCII ファイル) から直接 **LOB** をロードする場合は、次の規則を守ってください。
 - **LOB** を含めたレコードの全長は 32KB 以下でなければなりません。
 - 入力レコード内の **LOB** フィールドは固定長でなければならず、必要なら **LOB** データに空白を埋め込まなければなりません。
 - **LOB** をデータベースに挿入する際に、**LOB** の埋め込みに使われる後続空白を除去できるよう、**striptblanks** 修飾子を指定しなければなりません。

例 3 (ダンプ・ファイルの使用)

LOAD

表 FRIENDS は、次のように定義されています。

```
table friends "( c1 INT NOT NULL, c2 INT, c3 CHAR(8) )"
```

次のデータ・レコードをこの表にロードしようとすると、

```
23, 24, bobby
, 45, john
4,, mary
```

最初の INT が NULL で、列定義に NOT NULL が指定されているため、第 2 行は拒否されます。DEL 形式と互換でない開始文字を含む列は、エラーを生成し、レコードは拒否されます。そのようなレコードは、ダンプ・ファイルに書き込むことができます (437ページの表7 参照)。

文字区切り文字の外側にある列の DEL データは無視されますが、警告が生成されます。たとえば、次のようにします。

```
22,34,"bob"
24,55,"sam" sdf
```

ユーティリティーは、表の第 3 列に "sam" をロードし、警告の中で文字 "sdf" にフラグが付けられます。このレコードは拒否されません。別の例を考えましょう。

```
22 3, 34,"bob"
```

ユーティリティーは 22,34,"bob" をロードし、列 1 の中で 22 より後のデータは無視されたという警告を生成します。このレコードは拒否されません。

例 4 (DATALINK データのロード)

下記のコマンドは、DEL 形式のデータが含まれている入力ファイル delfile1 から、表 MOVIE TABLE をロードします。

```
db2 load from delfile1 of del
modified by dldel|
insert into movietable (actorname, description, url_making_of, url_movie)
datalink specification (dl_url_default_prefix "http://narang"),
(dl_url_replace_prefix "http://bomdel" dl_url_suffix ".mpeg")
for exception excptab
```

注:

1. この表には下記の 4 つの列が含まれています。

actorname	VARCHAR(n)
description	VARCHAR(m)
url_making_of	DATALINK (with LINKTYPE URL)
url_movie	DATALINK (with LINKTYPE URL)

2. 入力ファイルの中の DATALINK データのサブフィールド区切り文字は、縦線 (|) 文字です。
3. url_making_of の列値に接頭部文字列が含まれていないなら、"http://narang" が使用されます。
4. url_movie の非 NULL 列値には、接頭部として "http://bomdel" が付けられます。既存の値は置き換えられます。
5. url_movie の非 NULL 列値のパスには、".mpeg" が付加されます。たとえば、url_movie の列値が "http://server1/x/y/z" なら、それは "http://bomdel/x/y/z.mpeg" として保管されます。その値が "/x/y/z" なら "http://bomdel/x/y/z.mpeg" として保管されます。
6. 表のロード中に固有索引または DATALINK 例外が発生したなら、関係するレコードが表から削除され、例外表 excptab に入れられます。

例 5 (識別列がある表へのロード)

TABLE1 には以下の 4 つの列があります。

- C1 VARCHAR(30)
- C2 INT GENERATED BY DEFAULT AS IDENTITY
- C3 DECIMAL(7,2)
- C4 CHAR(1)

TABLE2 は TABLE1 と同じですが、C2 が GENERATED ALWAYS 識別列である点が異なります。

DATAFILE1 のデータ・レコード (DEL 形式):

```
"Liszt"
"Hummel",,187.43, H
"Grieg",100, 66.34, G
"Satie",101, 818.23, I
```

DATAFILE2 のデータ・レコード (DEL 形式):

```
"Liszt", 74.49, A
"Hummel", 0.01, H
"Grieg", 66.34, G
"Satie", 818.23, I
```

注:

1. 以下のコマンドは、DATAFILE1 で行 1 および 2 への識別値の提供がないので、それらの行のための識別値を生成します。ただし、行 3 および 4 は、それぞれユーザー提供の識別値 100 と 101 が割り当てられます。

```
db2 load from datafile1.del of del replace into table1
```

LOAD

2. DATAFILE1 を TABLE1 にロードしてすべての行に対する識別値を生成するには、以下のコマンドのいずれかを発行します。

```
db2 load from datafile1.del of del method P(1, 3, 4) replace into table1 (c1, c3, c4)
db2load from datafile1.del of del modified by identityignore replace into table1
```

3. DATAFILE2 を TABLE1 にロードして各行に対する識別値を生成するには、以下のコマンドのいずれかを発行します。

```
db2 load from datafile2.del of del replace into table1 (c1, c3, c4)
db2 load from datafile2.del of del modified by identitymissing replace into table1
```

4. 識別値 100 と 101 を行 3 および 4 に割り当てるために DATAFILE1 を TABLE2 にロードするには、以下のコマンドを発行します。

```
db2 load from datafile1.del of del modified by identityoverride replace into table2
```

この場合、ユーティリティには、ユーザー提供の値を優先して、システム生成の識別値に上書きするように指示しているため、行 1 および 2 は拒否されます。ユーザー提供の値が存在しない場合でも、識別列が暗黙的に非ヌルであるため、この行は拒否しなければなりません。

5. DATAFILE1 を TABLE2 に、識別関連のファイル・タイプ修飾子を使用せずにロードした場合、行 1 と 2 はロードされますが、行 3 と 4 は拒否されます。その理由は、それらが固有の非ヌル値を提供し、識別列が GENERATED ALWAYS であるからです。

使用上の注意

データは、入力ファイル内に並んでいる順序でロードされます。特定の順序にしたいなら、ロードの前にデータをソートしておく必要があります。

ロード・ユーティリティは、既存の定義に基づいて索引を作成します。固有キーの重複を処理するには、例外表が使用されます。ユーティリティは、参照保全を保証したり、制約検査を実行したり、ロード中の表に依存する要約表を更新したりしません。表に参照制約や検査制約が含まれている場合、その表は検査保留状態になります。REFRESH IMMEDIATE として定義されている要約表、およびロードする表に依存する要約表は、検査保留状態になります。表の検査保留状態を終了させるには、SET INTEGRITY ステートメントを発行してください。コピーされた要約表に対してロード操作を実行することはできません。

クラスター化が必要なら、ロード前にクラスター化索引でデータをソートしてください。

DB2 データ・リンク・マネージャーに関する考慮事項

各 DATALINK 列ごとに、それぞれ 1 つの列指定を括弧に入れて指定できます。各列指定は、1 つまたは複数の DL_LINKTYPE、接頭部、および DL_URL_SUFFIX 指定で構成されます。接頭部情報は、DL_URL_REPLACE_PREFIX かまたは DL_URL_DEFAULT_PREFIX 指定です。

DATALINK 列指定の数は、表で定義されている DATALINK の数と同じだけ定義できます。指定の順序は、挿入列リストの中での DATALINK 列の順序 (挿入列リストが INSERT INTO (insert-column, ...) で指定されている場合) か、または表定義内での順序 (insert-column が指定されていない場合) に従います。

たとえば、表の列が C1、C2、C3、C4、および C5 であり、その中で C2 と C5 だけが DATALINK タイプであって、挿入列 (insert-column) リストが (C1, C5, C3, C2) ならば、DATALINK 列指定は 2 つ必要です。第 1 の列指定は C5 のためのもの、第 2 の列指定は C2 のためのものです。挿入列リストが指定されていない場合には、第 1 の列指定は C2 のためのもの、第 2 の列指定は C5 のためのものとなります。

複数の DATALINK 列があり、いくつかの列には特定の指定が必要ないという場合、列指定には、指定順序の指定に関するあいまいさをなくすための括弧が少なくとも必要です。どの列についての指定もないなら、空括弧のリスト全体を除去することができます。したがって、省略時値でよいのであれば、DATALINK 指定は必ずしも必要ではありません。

FILE LINK CONTROL として定義されている DATALINK 列を含む表にデータをロードする場合は、以下のステップを実行してから、ロード・ユーティリティを呼び出してください。(すべての DATALINK 列が NO LINK CONTROL として定義されている場合は、これらのステップを実行する必要はありません)。

1. DATALINK 列の値によって参照されるデータ・リンク・サーバーに、DB2 データ・リンク・マネージャーがインストールされていることを確認します。
2. データベースを DB2 データ・リンク・マネージャーに登録します。
3. DATALINK 値として挿入されるすべてのファイルを、適切なデータ・リンク・サーバーにコピーします。
4. 接頭部名 (複数可) をデータ・リンク・サーバー上の DB2 データ・リンク・マネージャーに定義します。

LOAD

5. (ロードする) DATALINK データによって参照されるデータ・リンク・サーバーを、DB2 データ・リンク・マネージャー構成ファイルに登録します。

ロード・ユーティリティーの実行中に、DB2 とデータ・リンク・サーバーの間の通信に障害が発生することがあります。そのような場合、ロード操作は失敗します。その場合には、次のようにしてください。

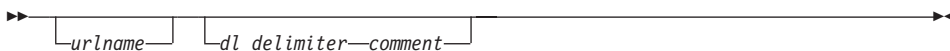
1. データ・リンク・サーバーと DB2 データ・リンク・マネージャーを開始します。
2. ロード再開操作を起動します。

ロード操作中に失敗したリンクはデータ保全性違反とみなされ、固有索引違反の場合とほとんど同じ仕方で処理されます。そのため、1 つ以上の DATALINK 列を含む表をロードする場合の特別な例外が定義されています。追加情報については、SQL 解説書の例外の説明を参照してください。

入力ファイル内での DATALINK 情報の表示

LINKTYPE (現在のところ URL のみサポート) は、DATALINK 情報の一部としては指定されません。LINKTYPE は、LOAD または IMPORT コマンドの中で指定されます。PC/IXF タイプの入力ファイルの場合は、データ移動ユーティリティー 手引きおよび解説書の中の付録『Export/Import/Load ユーティリティー・ファイル形式』で説明されているように、該当する列記述レコードの中で指定されます。

URL LINKTYPE のための DATALINK 情報の構文は、次のとおりです。



urlname と *comment* はいずれも任意選択です。どちらも省略した場合、ヌル値が代入されます。

urlname

この URL 名は有効な URL 構文に従うものでなければなりません。

注:

1. スキーム名として使えるのは "http" と "file" だけです。
2. URL 名の接頭部 (スキーム、ホスト、およびポート) はオプションです。接頭部がなければ、それはロード・ユーティリティーまたはインポート・ユーティリティーの DL_URL_DEFAULT_PREFIX または DL_URL_REPLACE_PREFIX の指定から取られます。そのどちらも指定されていないなら、省略時の接頭部として

"file://localhost" が使われます。このように、ローカル・ファイルの場合、LOAD または IMPORT コマンドの中で DATALINK 列指定なしで、フルパス名で指定したファイル名を URL 名として入力できます。

3. URL 名の中に接頭部が含まれていても、ロード操作またはインポート操作の DL_URL_REPLACE_PREFIX 指定に別の接頭部名が指定されているなら、それによってオーバーライドされることとなります。
4. 『パス』 (DL_URL_SUFFIX が指定されていればその後のパス) は、リモート・サーバーの中のリモート・ファイルのフルパス名です。相対パス名は使えません。http サーバーの省略時のパス接頭部は、それに含まれません。

dl_delimiter

区切り付き ASCII (DEL) ファイル形式の場合、LOAD または IMPORT コマンドの dlde1 修飾子によって指定される文字、または省略時の文字。区切りなし ASCII (ASC) ファイル形式の場合は、これを文字順序 ¥ (円記号とそれに続くセミコロン) に対応させる必要があります。ヌル文字 (ブランクやタブなど) も、このパラメーターに指定した値の前後に入れることができます。

comment

DATALINK 値の注釈部分。区切り付き ASCII (DEL) ファイル形式でこれを指定する場合、comment テキストは文字ストリング区切り文字で囲む必要があります。文字ストリング区切り文字は、省略時では二重引用符 (") です。この文字ストリング区切り文字は、LOAD または IMPORT コマンドの MODIFIED BY filetype-mod 指定によってオーバーライドできます。

注釈を指定しないなら、省略時として注釈は長さ 0 の文字列になります。

次に示すのは、区切り付き ASCII (DEL) ファイル形式の場合の DATALINK データの例です。

- `http://www.almaden.ibm.com:80/mrep/intro.mpeg; "Intro Movie"`
これは、次のような部分から構成されています。
 - スキーム = http
 - サーバー = www.almaden.ibm.com
 - パス = /mrep/intro.mpeg
 - 注釈 = "Intro Movie"

LOAD

- `file://narang/u/narang; "InderPal's Home Page"`
これは、次のような部分から構成されています。
 - スキーム = `file`
 - サーバー = `narang`
 - パス = `/u/narang`
 - 注釈 = `"InderPal's Home Page"`
- `file:/home/ff.gg; "hi there"`
これは、次のような部分から構成されています。
 - スキーム = `file`
 - サーバー = `localhost`
 - パス = `/home/ff.gg`
 - 注釈 = `"hi there"`

次に示すのは、区切りなし ASCII (ASC) ファイル形式の場合の DATALINK データの例です。

- `http://www.almaden.ibm.com:80/mrep/intro.mpeg;Intro Movie`
これは、次のような部分から構成されています。
 - スキーム = `http`
 - サーバー = `www.almaden.ibm.com`
 - パス = `/mrep/intro.mpeg`
 - 注釈 = `"Intro Movie"`
- `file://narang/u/narang; InderPal's Home Page`
これは、次のような部分から構成されています。
 - スキーム = `file`
 - サーバー = `narang`
 - パス = `/u/narang`
 - 注釈 = `"InderPal's Home Page"`
- `file:/home/ff.gg; hi there`
これは、次のような部分から構成されています。
 - スキーム = `file`
 - サーバー = `localhost`
 - パス = `/home/ff.gg`
 - 注釈 = `"hi there"`

以下に、DATALINK データの例を示します。列のロードまたはインポート指定が `DL_URL_DEFAULT_PREFIX` ("`http://qso`") であるとしています。

- `file://narang/pics/xxx.jpeg?search_pat`
これは、次のような部分から構成されています。
 - スキーム = `file`
 - サーバー = `narang`
 - パス = `/pics/xxx.jpeg`
 - 注釈 = `NULL` ストリング
- `/u/me/myfile.ps`
これは、次のような部分から構成されています。
 - スキーム = `http`
 - サーバー = `qso`
 - パス = `/u/me/myfile.ps`
 - 注釈 = `NULL` ストリング

表7. 有効なファイル・タイプ修飾子 (ロード)

修飾子	説明
すべてのファイル形式	
<code>anyorder</code>	この修飾子は、 <code>cpu_parallelism</code> パラメーターと共に使用されます。ソース・データ順序を保つことが必要でないことを指定し、それによって SMP システムでさらにパフォーマンスを高めます。 <code>cpu_parallelism</code> の値が 1 の場合、このオプションは無視されます。 <code>SAVECOUNT > 0</code> の場合、このオプションはサポートされません。一貫性ポイント後の破損回復では、データが順番にロードされることが必要になるからです。

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
fastparse	<p>ユーザー提供の列値に対して簡略化された構文検査が実行され、パフォーマンスが向上します。このオプション下でロードされた表は、アーキテクチャ的に正しいものが保証されています。また、このユーティリティーが十分なデータ検査を実行して、セグメント違反またはトラップが発生しないようにすることも保証されています。正しい形式のデータは、正しくロードされます。</p> <p>たとえば、ASC ファイルの整数列用のフィールド項目として値 123qwr4 が検出された場合、値が有効な数値を示していないので、ロード・ユーティリティーは通常それを構文エラーとします。fastparse を指定した場合、構文エラーは検出されず、任意の数値を整数フィールドがロードされます。この修飾子は、完全なデータだけに使用するよう注意する必要があります。ASCII データにおいてこのオプションを使うと、かなりパフォーマンスが上がる場合がありますが、PC/IXF データの場合は fastparse を指定してもそれほどパフォーマンスは上がりません。IXF はバイナリー・フォーマットであり、fastparse は解析操作と ASCII から内部形式への変換に影響するからです。</p>
generatedignore	<p>この修飾子は、ロード・ユーティリティーに、すべての生成列のデータはデータ・ファイルに存在するが、それらを無視すべきことを知らせます。ヌル値可の生成列の場合、これによってその列にヌルがロードされます。ヌル値不可の生成列の場合、生成列のデータ・タイプに応じた省略時値がロードされます。ロード操作の終了時に、SET INTEGRITY ステートメントを呼び出して、ロードした値を、生成列定義に応じて計算した値で置き換えるように強制することができます。この修飾子は、generatedmissing または generatedoverride 修飾子と共に使用することはできません。</p>
generatedmissing	<p>この修飾子を指定すると、ユーティリティーは、入力データ・ファイルには生成列のデータが全く入っていない (ヌルさえない) とみなし、その列にヌルをロードします。ロード操作の終了時に、SET INTEGRITY ステートメントを使用して、ヌルを、生成列定義に応じて計算した値で置き換えることができます。この修飾子は、generatedignore または generatedoverride 修飾子と共に使用することはできません。</p>

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
generatedoverride	<p>この修飾子は、(こうした列のタイプの通常の規則に反して) 表内のすべての生成列で、ヌル以外の明示的データを受け入れるようにロード・ユーティリティーに指示します。これは、他のデータベース・システムからデータを移行する場合や、 DROPPED TABLE RECOVERY オプションを指定した ROLLFORWARD DATABASE コマンドを使用して復元したデータから表をロードする場合に役立ちます。この修飾子を使用すると、データが入っていない行やヌル不可の生成列に対するヌルデータはすべて拒否されます (SQL3116W)。 注: このオプションを使用している場合、ロード・ユーティリティーは、生成列の値の妥当性検査を試行しません。</p> <p>この修飾子は、generatedmissing または generatedignore 修飾子と共に使用することはできません。</p>
identityignore	<p>この修飾子は、ロード・ユーティリティーに、識別列のデータはデータ・ファイルに存在するが、それらを見捨てるべきことを知らせます。この結果として、すべて識別値はこのユーティリティーによって生成されます。この動作は、GENERATED ALWAYS および GENERATED BY DEFAULT 識別列のどちらの場合も同じです。つまり、GENERATED ALWAYS 列の場合、拒否される行はないという意味です。この修飾子は、identitymissing または identityoverride 修飾子と共に使用することはできません。</p>
identitymissing	<p>この修飾子を指定すると、ユーティリティーは、入力データ・ファイルには識別列のデータが (NULL さえも) なく、したがって各行の値が生成されると想定します。この動作は、GENERATED ALWAYS および GENERATED BY DEFAULT 識別列のどちらの場合も同じです。この修飾子は、 identityignore または identityoverride 修飾子と共に使用することはできません。</p>
identityoverride	<p>この修飾子は、GENERATED ALWAYS として定義した識別列が、ロードする表に存在している場合にのみ使用するべきです。この修飾子は、(識別列のそれらのタイプの通常の規則に反して) そうした列で、ヌル以外の明示的データを受け入れるようにユーティリティーに指示します。これは、表を GENERATED ALWAYS と定義することが必要なときに他のデータベース・システムからデータを移行する場合や、 DROPPED TABLE RECOVERY オプションを指定した ROLLFORWARD DATABASE コマンドを使用して復元したデータから表をロードする場合に役立ちます。この修飾子を使用すると、データが入っていない行や識別列に対するヌルデータはすべて拒否されます (SQL3116W)。この修飾子は、identitymissing または identityignore 修飾子と共に使用することはできません。 注: このオプションを使用している場合、ロード・ユーティリティーは、表の識別列の値が固有であるように保守したり、その固有性を検査することはありません。</p>

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
indexfreespace= <i>x</i>	<p><i>x</i> は 0~99 の整数です。その値は、各索引ページの中で索引ロード時の空きスペースとして残しておく部分の割合を示すパーセントとして解釈されます。ページの最初の項目は、制限なしで追加されます。それより後の項目は、空きスペースのパーセントしきい値内である場合に追加されます。省略時値は、CREATE INDEX time で使用されている値です。</p> <p>この値は、CREATE INDEX ステートメントに指定された PCTFREE 値よりも優先して使用され、索引の葉ページだけが対象です。</p>
lobsinfile	<p><i>lob-path</i> には、LOB 値を含むファイルへのパスを指定します。ASC、DEL、または IXF ロード入力ファイルには、LOB 列に LOB データを含むファイルの名前が入っています。</p>
noheader	<p>ヘッダー検査コードをスキップします (単一ノードのノードグループに存在する表へのロード操作にのみ適用します)。</p> <p>オートローダー・ユーティリティ (データ移動ユーティリティ 手引きおよび解説書 中の『AutoLoader』を参照) は、マルチノードのノード・グループで表のデータの提供元となるファイルごとにヘッダーを書き込みます。そのヘッダーには、ノード番号、区分化マップ、および区分化キー指定が含まれています。ロード・ユーティリティでは、データが正しいノードでロードされていることを確認するためにこの情報が必要になります。単一ノードのノード・グループに存在する表にファイルをロードしている場合、このオプションを指定するとロード・ユーティリティはヘッダーの検査コードをスキップします。</p>
norowwarnings	拒否された行についての警告をすべて抑制します。
pagefreespace= <i>x</i>	<p><i>x</i> は 0~100 の整数です。その値は、各データ・ページの中で空きスペースとして残しておく部分の割合を示すパーセントとして解釈されます。</p> <p>指定された値が最小行サイズという点で無効な場合 (たとえばある行に 3000 バイト以上の長さがあり、<i>x</i> 値は 50 の場合)、その行は新しいページに入れられます。値 100 が指定された場合には、各行ごとにそれぞれ新しいページに入れられます。</p> <p>注: 表の PCTFREE 値は、ページごとの空きスペース量を決定するものとなります。ロード操作の pagefreespace 値も表の PCTFREE 値も設定されていない場合、ページごとに可能な限りたくさんのスペースが入れられます。pagefreespace によって設定された値は、表に対して指定されている PCTFREE 値をオーバーライドします。</p>
totalfreespace= <i>x</i>	<p><i>x</i> は 0~100 の整数です。その値は、表示の合計ページの中で空きスペースとして表示の最後に付加する部分の割合を示すパーセントとして解釈されます。たとえば、<i>x</i> が 20 で、表の中に 100 個のデータ・ページがある場合、20 個のヌル・ページが追加されます。その表のデータ・ページの合計数は 120 になります。</p>

表 7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
usedefaults	<p>ターゲット表の列に対応するソース列を指定していても、1 つまたは複数の行インスタンスにデータが入っていない場合、省略時値がロードされます。欠落データの例は次のとおりです。</p> <ul style="list-style-type: none"> • DEL ファイルの場合、列に ".,," が指定された場合 • DEL/ASC/WSF ファイルの場合、列が不足している行、または元の指定では十分な長さのない行。 <p>このオプションを指定せず、行インスタンスのソース列にデータがない場合、以下のいずれかになります。</p> <ul style="list-style-type: none"> • 列がヌル値可なら、ヌル値がロードされます。 • 列がヌル値不可の場合、ユーティリティは行を拒否します。
ASCII ファイル形式 (ASC/DEL)	
codepage=x	<p>x は ASCII 文字ストリングです。その値は、入力データ・セット中のデータのコード・ページとして解釈されます。ロード操作時に、このコード・ページからデータベースのコード・ページへ文字データ (および文字によって指定される数値データ) を変換します。</p> <p>以下の規則が適用されます。</p> <ul style="list-style-type: none"> • 純 DBCS (グラフィック)、混合 DBCS、および EUC では、区切り文字は x00~x3F の範囲に制限されます。 • EBCDIC コード・ページで指定された DEL データの場合、区切り文字はシフトインおよびシフトアウト DBCS 文字と一致しない場合があります。 • nullindchar には、コード・ポイントが x20~x7F の標準 ASCII セットに含まれる記号を指定する必要があります。これは、ASCII 記号およびコード・ポイントの場合です。EBCDIC データの場合は、コード・ポイントが異なるとしても対応する記号を使用できます。

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
dateformat="x"	<p>x は、ソース・ファイルの日付の形式です。^a 有効な日付要素は次のとおりです。</p> <p>YYYY - 年 (0000~9999 の範囲の 4 桁の数字) M - 月 (1~12 の範囲の 1 桁または 2 桁の数) MM - 月 (1~12 の範囲の 2 桁の数; M と同時には使用できない) D - 日 (1~31 の範囲の 1 桁または 2 桁の数) DD - 日 (1~31 の範囲の 2 桁の数; D と同時には使用できない) DDD - Day of the year (three digits ranging from 001 - 366; mutually exclusive with other day or month elements)</p> <p>省略時値の 1 が、指定されない各要素に割り当てられます。日付形式のいくつかの例を以下に示します。</p> <p>"D-M-YYYY" "MM.DD.YYYY" "YYYYDDD"</p>

表 7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
dumpfile = x	<p>x は、拒否された行を書き込む例外ファイルの (サーバー・ノードによる) 完全修飾名です。1 レコードにつき、最大で 32KB のデータが書き込まれます。ダンプ・ファイルの指定方法の例を下記に示します。</p> <pre data-bbox="619 343 1123 418">db2 load from data of del modified by dumpfile = /u/user/filename insert into table_name</pre> <p>注:</p> <ol data-bbox="585 479 1232 869" style="list-style-type: none"> 1. 区分データベース環境の場合、パスはロードを実行するノードにローカルなものでなければなりません。それによって、並行して実行される複数のロード操作が同じファイルに書き込むことを防ぐことができます。 2. ファイルの内容は、非同期バッファ・モードでディスクに書き込まれます。ロード操作が失敗したり割り込まれたりした場合、ディスクにコミットされたレコード数を決定できなくなり、LOAD RESTART 後の一貫性が保証されなくなります。ファイルが完全であるとされるのは、1 回のパスの中で開始して完了するロード操作の場合だけです。 3. この修飾子では、ファイル拡張子が複数のファイル名はサポートされません。たとえば、 <pre data-bbox="653 890 1059 913">dumpfile = /home/svtdbm6/DUMP.FILE</pre> はロード・ユーティリティーに対して指定できますが、 <pre data-bbox="653 991 1119 1013">dumpfile = /home/svtdbm6/DUMP.LOAD.FILE</pre> は指定できません。
implieddecimal	<p>暗黙の 10 進小数点の位置は、列定義によって指定されます。その位置が値の最後にあると想定されることはありません。たとえば、値 12345 が DECIMAL(8,2) 列にロードされる場合、123.45 としてであり、12345.00 ではありません。</p>

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
timeformat="x"	<p>x は、ソース・ファイルの時刻の形式です。^a 有効な時刻要素は次のとおりです。</p> <p>H - Hour (one or two digits ranging from 0 - 12 for a 12 hour system, and 0 - 24 for a 24 hour system)</p> <p>HH - Hour (two digits ranging from 0 - 12 for a 12 hour system, and 0 - 24 for a 24 hour system; mutually exclusive with H)</p> <p>M - 分 (0~59 の範囲の 1 桁または 2 桁の数)</p> <p>MM - 分 (0~59 の範囲の 2 桁の数; M と同時には使用できない)</p> <p>S - 秒 (0~59 の範囲の 1 桁または 2 桁の数)</p> <p>SS - 秒 (0~59 の範囲の 2 桁の数; S と同時には使用できない)</p> <p>SSSSS - Second of the day after midnight (5 digits ranging from 00000 - 86399; mutually exclusive with other time elements)</p> <p>TT - 午前午後の標識 (AM または PM)</p> <p>省略時値の 0 が、指定されない各要素に割り当てられます。時刻形式のいくつかの例を以下に示します。</p> <p>"HH:MM:SS" "HH.MM TT" "SSSSS"</p>

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
timestampformat="x"	<p>x は、ソース・ファイルのタイム・スタンプの形式です。^a 有効なタイム・スタンプ要素は次のとおりです。</p> <p>YYYY - 年 (0000~9999 の範囲の 4 桁の数字) M - 月 (1~12 の範囲の 1 桁または 2 桁の数) MM - 月 (1~12 の範囲の 2 桁の数; M (月) と同時には使用できない) D - 日 (1~31 の範囲の 1 桁または 2 桁の数) DD - 日 (1~31 の範囲の 2 桁の数; D と同時には使用できない) DDD - Day of the year (three digits ranging from 001 - 366; mutually exclusive with other day or month elements) H - Hour (one or two digits ranging from 0 - 12 for a 12 hour system, and 0 - 24 for a 24 hour system) HH - Hour (two digits ranging from 0 - 12 for a 12 hour system, and 0 - 24 for a 24 hour system; mutually exclusive with H) M - 分 (0~59 の範囲の 1 桁または 2 桁の数) MM - 分 (0~59 の範囲の 2 桁の数; M (分) と同時には使用できない) S - 秒 (0~59 の範囲の 1 桁または 2 桁の数) SS - 秒 (0~59 の範囲の 2 桁の数; S と同時には使用できない) SSSSS - Second of the day after midnight (5 digits ranging from 00000 - 86399; mutually exclusive with other time elements) UUUUUU - マイクロ秒 (000000 - 999999 の範囲の 6 桁 の数) TT - 午前午後の標識 (AM または PM)</p> <p>YYYY、M、MM、D、DD、または DDD が指定されていない場合、省略時値として 1 が割り当てられます。他の要素が指定されていない場合には、省略時値として 0 が割り当てられます。以下に、タイム・スタンプ形式の例を示します。</p> <p>"YYYY/MM/DD HH:MM:SS.UUUUUU"</p> <p>次の例では、ユーザー定義の日時形式を含むデータを、schedule という表にインポートする方法を示します。</p> <pre>db2 import from delfile2 of del modified by timestampformat="yyyy.mm.dd hh:mm tt" insert into schedule</pre>
noeofchar	<p>オプションのファイル終了文字 x'1A' は、ファイル終了として識別されません。それが普通の文字であるかのようにして処理は継続されます。</p>
ASC (区切りなし ASCII) ファイル形式	

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
binarynumerics	<p>数値データ (DECIMAL 以外) は、バイナリー形式でなければならず、文字表示であってはなりません。これによって、コストの大きい変換操作を避けることができます。</p> <p>このオプションがサポートされるのは、定位置 ASC において、reclen オプションによって固定長レコードが指定されている場合だけです。noeofchar オプションが前提です。</p> <p>以下の規則が適用されます。</p> <ul style="list-style-type: none"> データ型の変換は実行されません。ただし、BIGINT、INTEGER、および SMALLINT については実行されます。 データ長は、ターゲット列定義と一致している必要があります。 FLOAT は、IEEE 浮動小数点形式でなければなりません。 ロードのソース・ファイルに含まれるバイナリー・データは、ロード操作がどのプラットフォームで実行されるとしても、ビッグ・エンディアンであることが前提です。 <p>注: この修飾子の影響を受ける列のデータにヌル値があってはなりません。この修飾子を使用されると、ブランク (通常はヌル値として解釈される) は 1 個のバイナリー値として解釈されます。</p>
nochecklengths	<p>nochecklengths を指定した場合は、ソース・データの中にターゲット表列のサイズを超える列定義が含まれている場合であっても、各行のロードが試みられます。コード・ページ変換によってソース・データが縮小されれば、そのような行であったとしても正常にロードすることができます。たとえば、ソースに 4 バイトの EUC データがある場合、それがターゲットで 2 バイトの DBCS データに縮小されれば、必要なスペースは半分で済みます。列定義が一致しなくてもソース・データがきちんと収まるのが明らかな場合に、このオプションは特に便利です。</p>
nullindchar=x	<p>x は単一文字です。ヌル値を示す文字を x に変更します。 x の省略時値は Y です。^b</p> <p>EBCDIC データ・ファイルの場合、この修飾子は大文字小文字を区別しますが、英字の場合は区別しません。たとえば、ヌル標識文字を文字 N に指定した場合、 n もヌル標識として認識されます。</p>

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
packeddecimal	<p>binarynumerics 修飾子には DECIMAL フィールド・タイプが含まれないので、パック 10 進データを直接ロードします。</p> <p>このオプションがサポートされるのは、定位置 ASC において、reclen オプションによって固定長レコードが指定されている場合だけです。noeofchar オプションが前提です。</p> <p>ニブル符号用にサポートされる値は以下のとおりです。</p> <pre> + = 0xC 0xA 0xE 0xF - = 0xD 0xB </pre> <p>注: この修飾子の影響を受ける列のデータにヌル値があってはなりません。この修飾子を使用されると、ブランク (通常はヌル値として解釈される) は 1 個のバイナリー値として解釈されます。</p> <p>サーバーのプラットフォームには関係なく、ロードのソース・ファイルに含まれるバイナリー・データのバイト順はビッグ・エンディアンであることが前提となっています。つまり、この修飾子を OS/2 または Windows オペレーティング・システムで使用する場合も、バイト順を逆にしてはなりません。</p>
reclen= <i>x</i>	<p><i>x</i> は、32 767 以下の整数です。1 行につき <i>x</i> 文字ずつ読まれます。改行文字は行の終了にはなりません。</p>
striptblanks	<p>可変長フィールドにデータをインポートする場合に、後書きブランクをすべて切り捨てます。このオプションを指定しない場合、ブランク・スペースはそのまま保持されます。</p> <p>このオプションは、striptnulls と一緒には指定できません。それらは相互に排他的なオプションです。</p> <p>注: このオプションは、廃止された t オプション (後方互換性のためだけにサポートされる) に代わるものです。</p>
striptnulls	<p>可変長フィールドにデータをインポートする場合に、後書きヌル値 (0x00 文字) をすべて切り捨てます。このオプションを指定しない場合、ヌル値はそのまま保持されます。</p> <p>このオプションは、striptblanks と一緒には指定できません。それらは相互に排他的なオプションです。</p> <p>注: このオプションは、廃止された padwithzero オプション (後方互換性のためだけにサポートされる) に代わるものです。</p>

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
zoneddecimal	<p>BINARYNUMERICs 修飾子には DECIMAL フィールド・タイプが含まれないので、ゾーン 10 進データをロードします。このオプションがサポートされるのは、定位置 ASC において、RECLEN オプションによって固定長レコードが指定されている場合だけです。NOEOFCHAR オプションが前提です。</p> <p>ハーフバイト符号値は、以下のいずれかになります。</p> <p>+ = 0xC 0xA 0xE 0xF - = 0xD 0xB</p> <p>サポートされている数値は、0x0 ~ 0x9 です。</p> <p>サポートされているゾーン値は、0x3 と 0xF です。</p>
DEL (区切り付き ASCII) ファイル形式	
chardelx	<p>x は単一文字ストリング区切り文字です。省略時値は二重引用符 (") です。指定した文字は、文字ストリングを囲むために、二重引用符の代わりに使用されます。^{bc}</p> <p>単一引用符 (') も次のように文字ストリング区切り文字として指定できます。</p> <p style="text-align: center;">modified by chardel''</p>
coldelx	<p>x は単一文字列区切り文字です。省略時値はコンマ (,) です。指定した文字は、列の終わりを表すために、コンマの代わりに使用されます。^{bc}</p>
datesiso	<p>日付形式。すべての日付データ値を ISO 形式でロードします。</p>
decplusblank	<p>正符号文字。これによって正の 10 進値の先頭に正符号 (+) ではなく、ブランク・スペースが置かれます。省略時のアクションでは、正の 10 進数の前に正符号 (+) が付けられます。</p>
decptx	<p>x は、小数点としてピリオドと置換される単一文字です。省略時値はピリオド (.) です。指定した文字は、小数点文字としてピリオドの代わりに使用されます。^{bc}</p>

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
delprioritychar	<p>区切り文字の現在の省略時優先順位は、(1) レコード区切り文字、(2) 文字区切り文字、(3) 列区切り文字です。この修飾子を使用すると、区切り文字の優先順位が(1) 文字区切り文字、(2) レコード区切り文字、(3) 列区切り文字に戻り、以前の優先順位に依存している既存のアプリケーションが保護されます。構文は次のとおりです。</p> <pre>db2 load ... modified by delprioritychar ...</pre> <p>たとえば、次のような DEL データ・ファイルがあるとします。</p> <pre>"Smith, Joshua",4000,34.98<row delimiter> "Vincent,<row delimiter>, is a manager", 4005,44.37<row delimiter></pre> <p>delprioritychar 修飾子を指定すれば、このデータ・ファイルは 2 行だけになります。2 番目の <row delimiter> は 2 番目の行の最初のデータ列の一部として解釈されるのに対し、1 番目と 3 番目の <row delimiter> は実際のレコード区切り文字として解釈されます。この修飾子を指定しなかった場合は、このデータ・ファイルは 3 行のまま、各行は <row delimiter> によって区切られます。</p>
dldelx	<p>x は単一文字の DATALINK 区切り文字です。省略時値はセミコロン (;) です。指定した文字はセミコロンの代わりに、DATALINK 値のフィールド間区切り文字として使用されます。DATALINK 値には副値が複数個含まれる場合があるため、これが必要になります。^{bed}</p> <p>注: 行、列、または文字ストリング区切り文字と同じ文字を x に指定することはできません。</p>
keepblanks	<p>タイプが CHAR、VARCHAR、LONG VARCHAR、または CLOB である各フィールドの前後のブランクを保存します。このオプションを指定しないと、文字区切り文字で囲まれていないすべての前後のブランクは除去され、表のすべてのブランク・フィールドにヌルが挿入されます。</p> <p>以下の例では、データ・ファイルにある前後のブランクを保存しながら、TABLE1 という表にデータをロードする方法を示します。</p> <pre>db2 load from delfile3 of del modified by keepblanks insert into table1</pre>
nodoubledel	<p>二重文字区切り文字の認識を抑制します。詳細については、265ページの『区切り文字の制限』を参照してください。</p>
IXF ファイル形式	

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
forcein	<p>コード・ページが一致していなくてもデータを受け入れ、コード・ページの変換を抑制するようにユーティリティーに指示します。</p> <p>固定長ターゲット・フィールドは、データを入れるだけの十分な大きさがあるかどうかを検査されます。 <code>nochecklengths</code> を指定した場合、そのような検査は実行されず、各行のロードが試みられます。</p>
nochecklengths	<p><code>nochecklengths</code> を指定した場合は、ソース・データの中にターゲット表列のサイズを超える列定義が含まれている場合であっても、各行のロードが試みられます。コード・ページ変換によってソース・データが縮小されれば、そのような行であったとしても正常にロードすることができます。たとえば、ソースに 4 バイトの EUC データがある場合、それがターゲットで 2 バイトの DBCS データに縮小されれば、必要なスペースは半分で済みます。列定義が一致しなくてもソース・データがきちんと収まるのが明らかな場合に、このオプションは特に便利です。</p>

表7. 有効なファイル・タイプ修飾子 (ロード) (続き)

修飾子	説明
	<p>注:</p> <ol style="list-style-type: none"> 1. MODIFIED BY オプションでサポートされていないファイル・タイプが指定されても、ロード・ユーティリティーが警告を発することはありません。サポートされていないファイル・タイプを使おうとすると、ロード操作は失敗し、エラー・コードが戻されます。 2. ^a 日付形式ストリングを囲む二重引用符は必須です。フィールド区切り文字には、a~z、A~Z、および 0~9 を含めることはできません。フィールド区切り文字は、文字区切り文字、または DEL ファイル形式のフィールド区切り文字と同じであってはなりません。フィールド区切り文字は、要素の開始および終了位置が明確な場合は任意指定です。(修飾子によって) D、H、M、または S などの要素が使用される場合、項目が可変長であるためにあいまいさが存在することがあります。 タイム・スタンプ形式の場合、月と分の記述子の両方に文字 M が使用されるので、あいまいさを避けるように注意する必要があります。月フィールドは、他の日付フィールドと隣接してなければなりません。分フィールドは、他の時刻フィールドと隣接してなければなりません。以下に、いくつかのあいまいなタイム・スタンプ形式を示します。 " M " (月または分のどちらにもとれる) " M : M " (月と分が区別がつかない) " M : Y Y Y Y : M " (両方とも月と解釈される) " S : M : Y Y Y Y " (時刻値と日付値の両方に隣接している) あいまいな場合、ユーティリティーはエラー・メッセージを報告し、操作は失敗します。 以下に示すのは、明確なタイム・スタンプ形式です。 " M : Y Y Y Y " (M (月)) " S : M " (M (分)) " M : Y Y Y Y : S : M " (M (月) . . . M (分)) " M : H : Y Y Y Y : M : D " (M (分) . . . M (月)) <p>注: 二重引用符や円記号などの文字は、エスケープ文字が先行する必要があります (たとえば ¥)。</p> <ol style="list-style-type: none"> 3. ^b この文字は、ソース・データのコード・ページで指定されている必要があります。 (文字記号ではなく) 文字コード・ポイントは、 xJJ または 0xJJ という構文で指定することができます (JJ はコード・ポイントの 16 進表示)。たとえば、列区切りとして # 文字を指定するには、以下のうちの 1 つを使用します。 . . . modified by coldel# modified by coldel0x23 modified by coldelX23 . . . 4. ^c 265ページの『区切り文字の制限』に、区切り文字の指定変更として使用できる文字に適用される制限のリストが示されています。 5. ^d DATALINK 区切り文字は URL 構文の中では有効な文字ですが、ロード操作の範囲内ではその特別な意味がなくなります。

LOAD

以下も参照

453ページの『LOAD QUERY』

491ページの『QUIESCE TABLESPACES FOR TABLE』

LOAD QUERY

処理中にロード操作の状況を調べます。このコマンドを正常に呼び出すためには、同じデータベースへの接続と、別の CLP セッションも必要になります。このコマンドは、ローカル・ユーザーでもリモート・ユーザーでも使用できます。

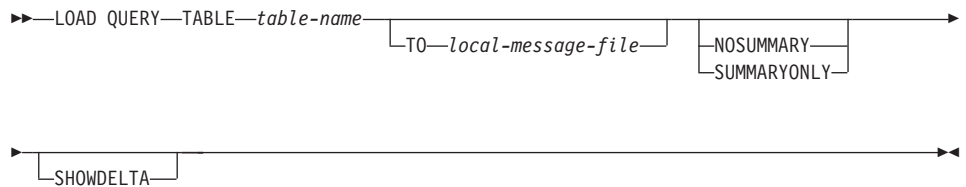
許可

なし

必須接続

データベース

コマンド構文



コマンド・パラメーター

NOSUMMARY

ロード要約情報 (読み取られた行、スキップされた行、ロードされた行、拒否された行、削除された行、コミットされた行、警告の数) のレポートを生成しないよう指定します。

SHOWDELTA

新しい情報 (最後の LOAD QUERY コマンド以後に発生したロード・イベントに関する) のレポートだけを生成するよう指定します。

SUMMARYONLY

ロード要約情報のレポートだけを生成するよう指定します。

TABLE *table-name*

データが現在ロード中の表の名前を指定します。未修飾の表名を指定すると、その表は CURRENT SCHEMA で修飾されます。

TO *local-message-file*

ロード操作中に生じ得る警告およびエラー・メッセージの宛先を指定します。このファイルは、LOAD コマンド用に指定された *message-file*

LOAD QUERY

であってはなりません。ファイルがすでに存在する場合、ロード・ユーティリティーが生成するメッセージはすべてそのファイルに追加されません。

例

大量のデータを STAFF 表にロードしている場合、ロード操作の状況を検査することが必要になるかもしれません。ユーザーは次のように指定することができます。

```
db2 connect to <database>
db2 load query table staff to /u/mydir/staff.tempsmsg
```

出力ファイル /u/mydir/staff.tempsmsg は、次のようになります。

```
SQL3500W The utility is beginning the "LOAD" phase at time
"02-13-1997 19:40:29.645353".
```

```
SQL3519W Begin Load Consistency Point. Input record count = "0".
```

```
SQL3520W Load Consistency Point was successful.
```

```
SQL3109N The utility is beginning to load data from file
"/u/mydir/data/staffbig.ixf".
```

```
SQL3150N The H record in the PC/IXF file has product "DB2 01.00",
date "19970111", and time "194554".
```

```
SQL3153N The T record in the PC/IXF file has name
"data/staffbig.ixf", qualifier " ", and source " ".
```

```
SQL3519W Begin Load Consistency Point. Input record count =
"111152".
```

```
SQL3520W Load Consistency Point was successful.
```

```
SQL3519W Begin Load Consistency Point. Input record count =
"222304".
```

```
SQL3520W Load Consistency Point was successful.
```

以下も参照

411ページの『LOAD』

MIGRATE DATABASE

旧バージョン (バージョン 2.x 以降) の DB2 データベースを現行の形式へ変換します。

考慮事項: DB2 バージョン 6 のインストール (OS/2 または Windows オペレーティング・システムの場合)、またはインスタンスの移行 (UNIX 系システム) の前には、データベース事前移行ツール (34ページの『db2ckmig - データベース事前移行ツール』) を実行しておく必要があります。それは DB2 バージョン 6 では実行できないためです。移行作業、および OS/2 または Windows オペレーティング・システムへの DB2 バージョン 6 のインストール作業に先立って、すべてのデータベースをバックアップしておいてください。

データベースの移行については、概説およびインストール の資料を参照してください。

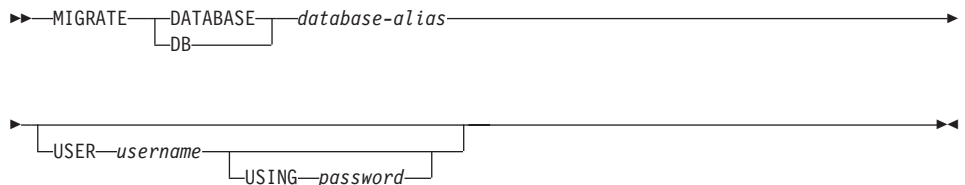
許可

sysadm

必須接続

このコマンドは、データベース接続を確立します。

コマンド構文



コマンド・パラメーター

DATABASE *database-alias*

現在インストールされているデータベース・マネージャーのバージョンに移行するデータベースの別名を指定します。

USER *username*

データベースを移行するのに使うユーザー名を指定します。

MIGRATE DATABASE

USING password

ユーザー名を承認するために使用するパスワード。ユーザー名を指定してパスワードを省略すると、ユーザーに入力を求めるプロンプトが出ます。

例

次の例は、データベースの別名 `sales` を使ってカタログ化されたデータベースを移行します。

```
db2 migrate database sales
```

使用上の注意

このコマンドは、データベースを新しいバージョンに移行するだけであり、移行済みのデータベースを元のバージョンに戻すために使用することはできません。

移行の前にデータベースをカタログ化しておく必要があります。

移行の途中でエラーが発生する場合、提案されているユーザー応答を試みる前に、586ページの『**TERMINATE**』を発行することが必要になる場合があります。たとえば、移行中にログが満ぱいになるというエラーが生じる場合 (SQL1704: データベースの移行は失敗しました。理由コード "3")、データベース構成パラメーター `LOGPRIMARY` および `LOGFILSIZ` の値を増やす前に、**TERMINATE** コマンドを発行する必要があります。データベースがすでに再配置された後に移行が失敗した場合、`CLP` はそのデータベース・ディレクトリー・キャッシュをリフレッシュする必要があります (「ログが満ぱい」エラーが戻される場合にこのようになる可能性があります)。

PRECOMPILE PROGRAM

組み込み SQL ステートメントの含まれているアプリケーション・プログラム・ソース・ファイル処理します。SQL ステートメントに対するホスト言語呼び出しを含む変更後のソース・ファイルが作成されます。また、省略時値として、データベース内にパッケージが作成されます。

効力範囲

このコマンドは、db2nodes.cfg 中のどのノードからでも発行できます。実行すると、カタログ・ノードのデータベース・カタログが更新されます。その影響はすべてのノードから見えます。

許可

以下のどれかが必要です。

- *sysadm* または *dbadm* の権限
- パッケージが存在しない場合は、BINDADD 特権および以下のどちらかが必要です。
 - パッケージのスキーマ名が存在しない場合は、データベースに対する IMPLICIT_SCHEMA 権限
 - パッケージのスキーマ名が存在する場合は、スキーマに対する CREATEIN 特権
- パッケージが存在する場合は、スキーマに対する ALTERIN 特権
- パッケージに対する BIND 特権 (パッケージが存在する場合)

ユーザーには、アプリケーション内の静的 SQL ステートメントをコンパイルするのに必要な特権もすべて必要になります。グループに付与された特権は、静的ステートメントの許可検査では使用されません。ユーザーに *sysadm* 権限があつてバインドを完成させる明示特権がない場合、データベース・マネージャーは、明示的な *dbadm* 権限を自動的に付与します。

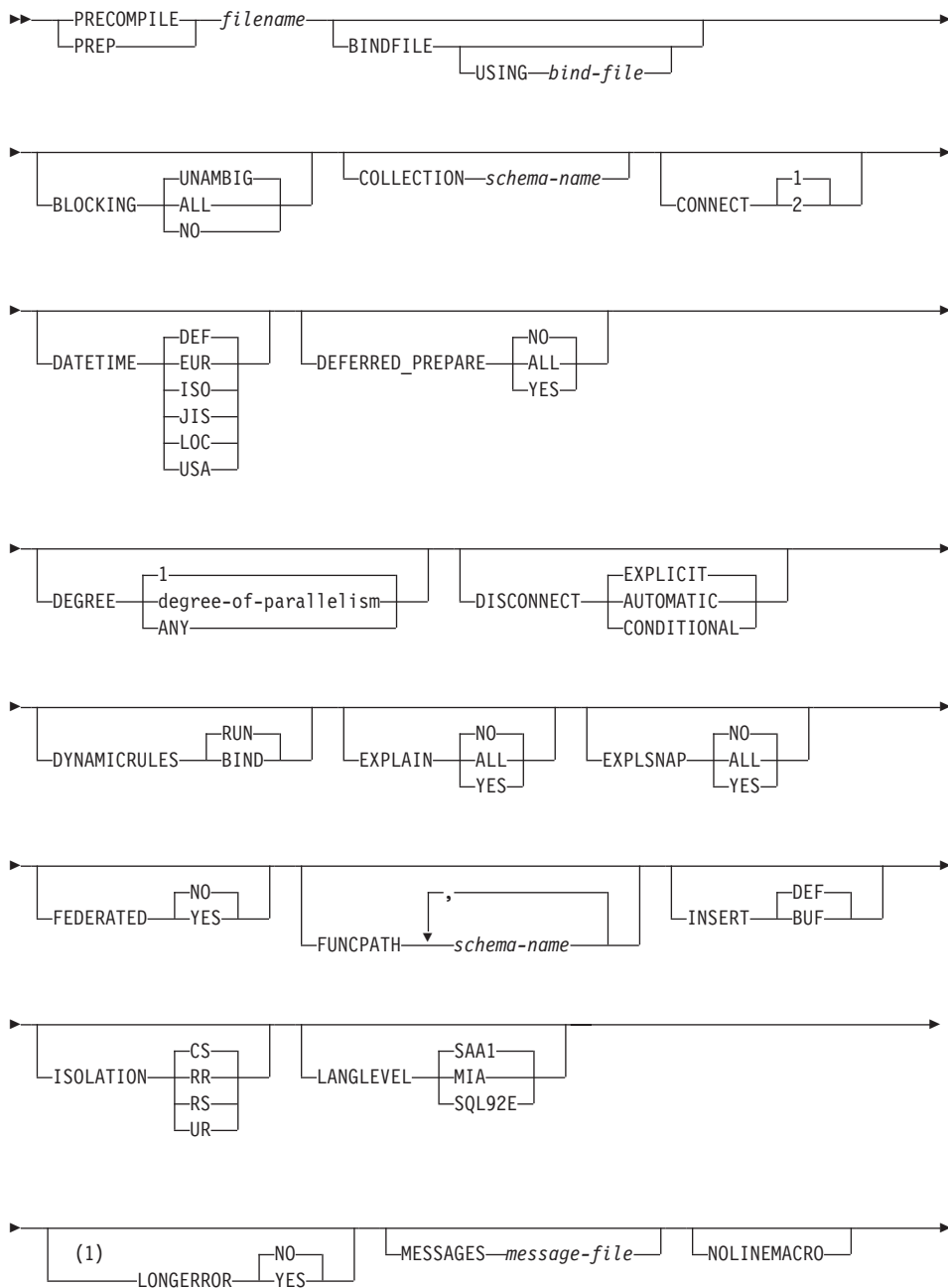
必須接続

データベース。暗黙接続が可能な場合には、省略時データベースへの接続が確立されます。

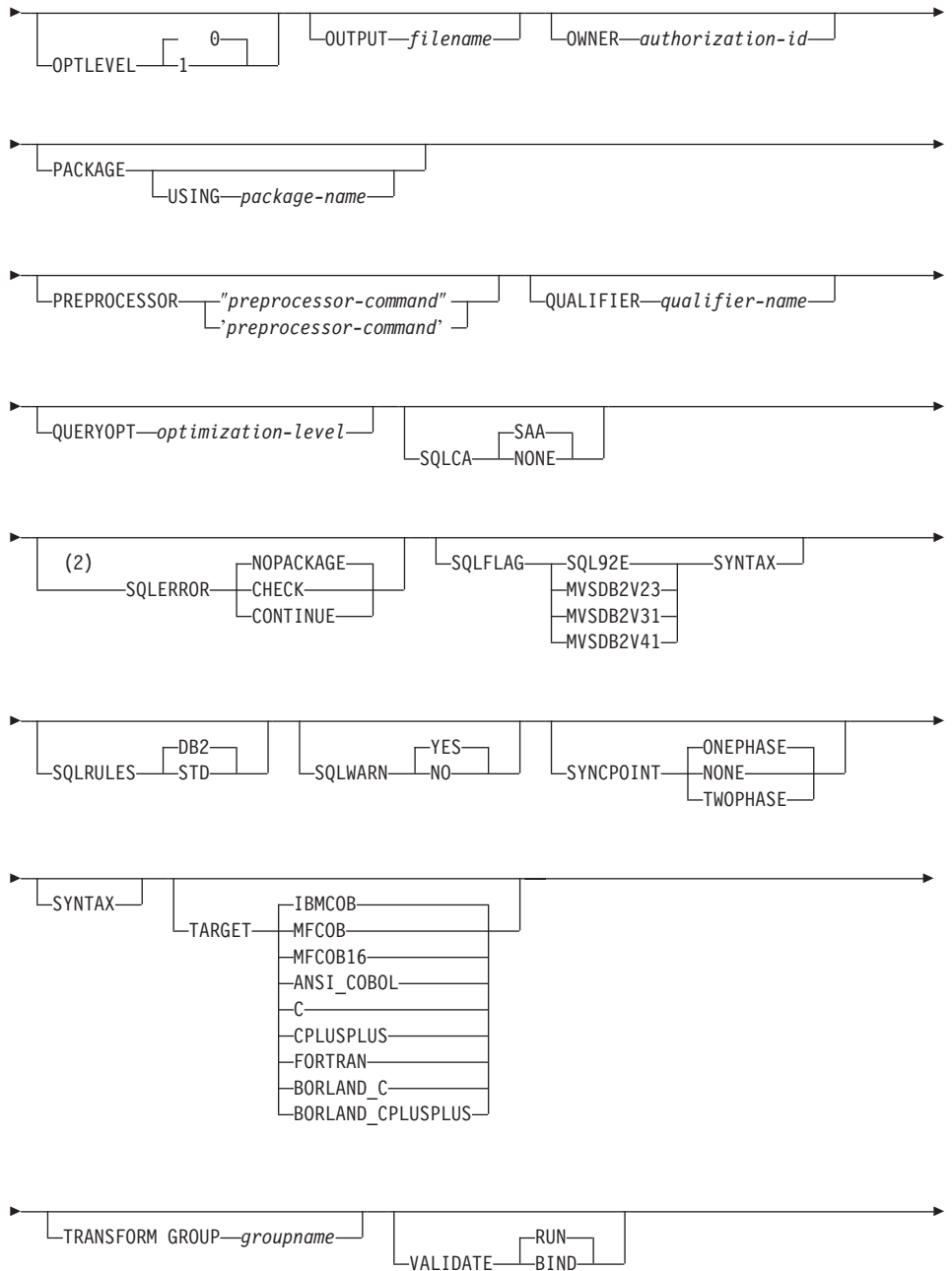
コマンド構文

DB2 の場合

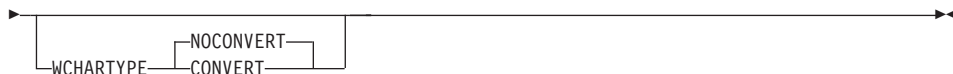
PRECOMPILE PROGRAM



PRECOMPILE PROGRAM



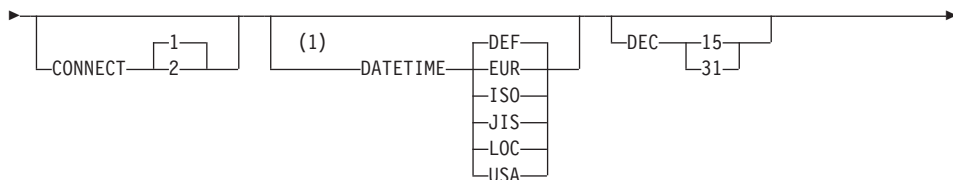
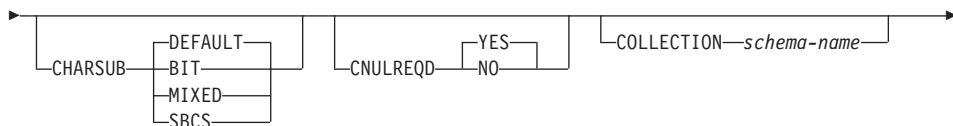
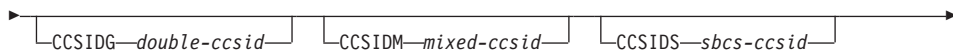
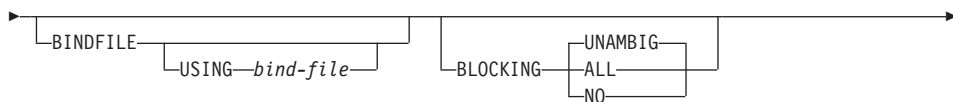
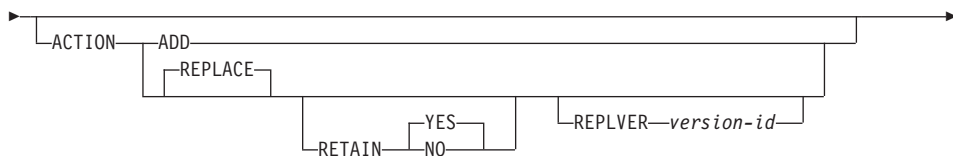
PRECOMPILE PROGRAM



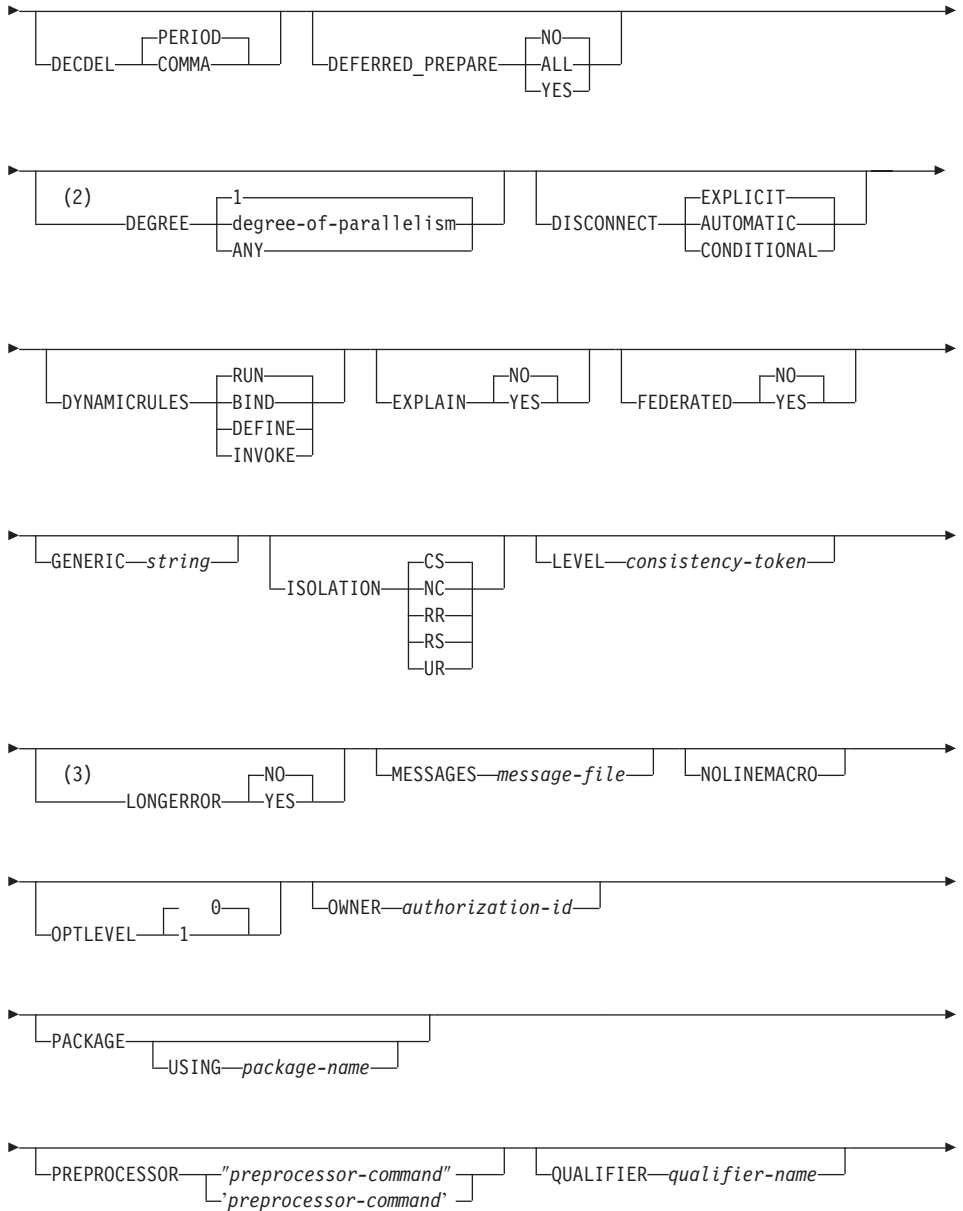
注:

- 1 NO は、32 ビット・システムと 64 ビット NT システムの省略時値です。それらのシステムでは、長いホスト変数を INTEGER 列の宣言として使用できます。YES は、64 ビット UNIX システムの省略時値です。
- 2 SYNTAX は SQLERROR(CHECK) の同義語です。

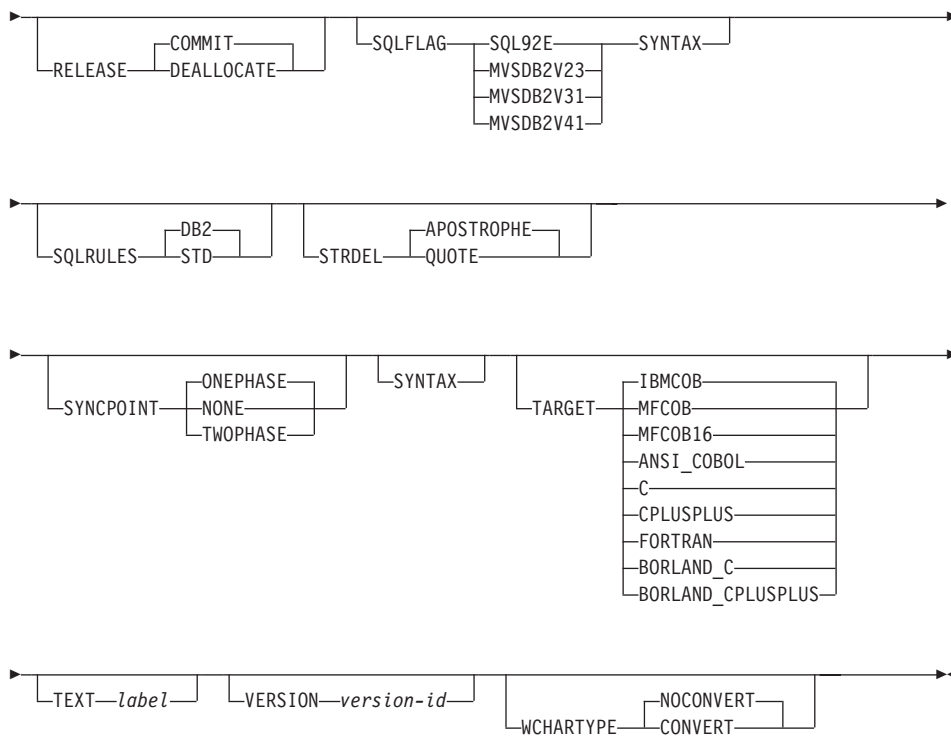
DRDA の場合



PRECOMPILE PROGRAM



PRECOMPILE PROGRAM



注:

- 1 DATETIME DEF オプションは DRDA ではサポートされておらず、DB2 コネクトを経て ISO にマップされます。
- 2 DEGREE オプションは DRDA レベル 2 のアプリケーション・サーバーでしかサポートされていません。
- 3 NO は、32 ビット・システムと 64 ビット NT システムの省略時値です。それらのシステムでは、長いホスト変数を INTEGER 列の宣言として使用できます。YES は、64 ビット UNIX システムの省略時値です。

コマンド・パラメーター

filename

プリコンパイルするソース・ファイルを指定します。拡張子の指定は、以下のようになります。

- C アプリケーションの場合、.sqc を指定します (.c ファイルが生成されます)。

- C++ アプリケーションの場合、.sqx (OS/2 または Windows オペレーティング・システム)、または .sqc (UNIX 系システム) を指定します (OS/2 または Windows オペレーティング・システムの場合は .cxx ファイル、UNIX 系システムの場合は .c ファイルが作成されます)。
- COBOL アプリケーションの場合、.sqb を指定します (.cbl ファイルが生成されます)。
- FORTRAN アプリケーションの場合、.sqf を指定します (OS/2 または Windows オペレーティング・システムの場合は .for ファイル、UNIX 系システムの場合は .f ファイルが生成されます)。

UNIX 系システムにおいて、組み込み SQL を含む C++ アプリケーションの場合に望ましい拡張子は sqc です。しかし、UNIX 系システムでは、大文字小文字を区別しないシステムのための sqx 規則も通用します。

ACTION

パッケージを追加または置換できるかどうかを示します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。

ADD 名前付きパッケージが存在せず、新規パッケージを作成するということを指示します。すでにパッケージがある場合は、実行停止状態となり、診断エラー・メッセージが戻されます。

REPLACE

既存のパッケージを、ロケーション、コレクション、およびパッケージ名が同じ新規パッケージと置き換えることを指示します。

RETAIN

パッケージを置き換えたときに EXECUTE 権限が保持されるかどうかを指示します。パッケージの所有権を変更した場合、新規所有者は前のパッケージ所有者に BIND 権限と EXECUTE 権限を付与します。

NO パッケージを置き換えたとき、EXECUTE 権限を保持しません。

YES パッケージを置き換えたとき、EXECUTE 権限を保持します。

REPLVER version-id

特定のバージョンのパッケージを置き換えます。パー

PRECOMPILE PROGRAM

ジョン識別子は、どのバージョンのパッケージを置き換えるのかを指定するものです。最大長は 254 文字です。

BINDFILE

バインド・ファイルが作成されます。 **package** オプションが共に指定されていない場合、パッケージは作成されません。次に示す例のように、バインド・ファイルを要求したのにパッケージが作成されないなら、

```
db2 prep sample.sqc bindfile
```

オブジェクトの存在と認証 **SQLCODE** はエラーではなく警告として扱われます。それで、プリコンパイルに使用されるデータベースに、アプリケーション内の静的 **SQL** ステートメントで参照されているすべてのオブジェクトが入っているわけではない場合でも、バインド・ファイルは正常に作成されます。必要なオブジェクトが作成されたなら、バインド・ファイルは正常にバインドされ、パッケージが作成されます。

USING bind-file

プリコンパイラーが生成するバインド・ファイルの名前。ファイル名には、**.bnd** 拡張子が付いていなければなりません。ファイル名を入力しないなら、プリコンパイラーは、**filename** パラメーターとして入力されているプログラムの名前を使用し、それに **.bnd** 拡張子を付けてファイル名とします。パスを指定しないと、バインド・ファイルは現行ディレクトリーに作成されます。

BLOCKING

行のブロック化については、 [管理の手引き](#) または [アプリケーション開発の手引き](#) を参照してください。

ALL 次のカーソルをブロック化することを指定します。

- 読み取り専用カーソル。
- **FOR UPDATE OF** と指定されていないカーソル。

未確定のカーソルは、読み取り専用として扱われます。

NO どのカーソルもブロック化しないことを指定します。未確定のカーソルは、更新可能として扱われます。

UNAMBIG

次のカーソルをブロック化することを指定します。

- 読み取り専用カーソル。

- FOR UPDATE OF と指定されていないカーソル。
未確定のカーソルは、更新可能として扱われます。

CCSIDG double-ccsid

CREATE および ALTER TABLE SQL ステートメントの文字列定義で、2 バイト文字用のコード化文字セット識別子 (CCSID) (特定の CCSID 文節は使用しない) を指定する整数。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

CCSIDM mixed-ccsid

CREATE および ALTER TABLE SQL ステートメントの文字列定義で、混合バイト文字用のコード化文字セット識別子 (CCSID) (特定の CCSID 文節は使用しない) を指定する整数。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

CCSIDS sbcs-ccsid

CREATE および ALTER TABLE SQL ステートメントの文字列定義で、1 バイト文字用のコード化文字セット識別子 (CCSID) (特定の CCSID 文節は使用しない) を指定する整数。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

CHARSUB

CREATE および ALTER TABLE SQL ステートメントの列定義に使用する、省略時の文字サブタイプを指定します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。

BIT 明示的にサブタイプを指定しなかった場合、すべての新規文字列に FOR BIT DATA SQL 文字サブタイプが使用されます。

DEFAULT

明示的にサブタイプを指定しなかった場合、すべての新規文字列にターゲット・システムが定義した省略時サブタイプが使用されます。

MIXED

明示的にサブタイプを指定しなかった場合、すべての新規文字列に FOR MIXED DATA SQL 文字サブタイプが使用されます。

PRECOMPILE PROGRAM

SBCS 明示的にサブタイプを指定しなかった場合、すべての新規文字列に FOR SBCS DATA SQL 文字サブタイプが使用されません。

CNULREQD

このオプションは DRDA でサポートされていない **langlevel** プリコンパイル・オプションと関連します。これは、C または C++ アプリケーションで作成されたバインド・ファイルの場合のみ有効です。この DRDA バインド・オプションは、DB2 ではサポートされていません。

NO C ストリング・ホスト変数中の NULL 終止符に関して、**langlevel SAA1** プリコンパイル・オプションに基づいてアプリケーションがコード化された場合です。

YES C ストリング・ホスト変数中の NULL 終止符に関して、**langlevel MIA** プリコンパイル・オプションに基づいてアプリケーションがコード化された場合です。

COLLECTION schema-name

パッケージ用の 30 文字の収集識別子を指定します。これを指定しなかった場合、パッケージを処理する際には、ユーザーの許可識別子が使用されます。

CONNECT

1 CONNECT ステートメントをタイプ 1 の CONNECT として処理するよう指定します。

2 CONNECT ステートメントをタイプ 2 の CONNECT として処理するよう指定します。

DATETIME

使用する日時形式を指定します。日時形式の詳細については、*SQL 解説書* を参照してください。

DEF データベースの国別コードと対応する日時形式を使用します。

EUR IBM 欧州規格の日時形式を使用します。

ISO 国際標準化機構規格の日時形式を使用します。

JIS 日本工業規格の日時形式を使用します。

LOC データベースの国別コードと対応する地域別日時形式を使用します。

USA IBM 米国規格の日時形式を使用します。

DEC 10 進算術演算に使用する最大精度を指定します。なお、DB2 はこの

DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

15 10 進算術演算に 15 桁精度が使用されます。

31 10 進算術演算に 31 桁精度が使用されます。

DECDEL

10 進数および浮動小数点リテラル中で 10 進小数点標識としてピリオド (.) またはコンマ (,) のどちらかを指定します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

COMMA

10 進小数点標識としてコンマ (,) を使用します。

PERIOD

10 進小数点標識としてピリオド (.) を使用します。

DEFERRED_PREPARE

DB2 共通サーバー・データベースまたは DRDA データベースにアクセスする際のパフォーマンス機能を強化します。このオプションは、SQL PREPARE ステートメントを、それに関連した OPEN、DESCRIBE、または EXECUTE ステートメント・フローと結合して、プロセス間またはネットワーク・フローを最小にします。

NO このコマンドが発行されると同時に PREPARE ステートメントが実行されます。

YES PREPARE ステートメントの実行は、対応する OPEN、DESCRIBE、または EXECUTE オープンが発行されるまで据え置かれます。

SQLDA を即時に戻す必要がある INTO 文節を使用している場合、PREPARE ステートメントは据え置かれませんが、パラメーター・マーカを使用しないカーソルに対して PREPARE INTO ステートメントが発行される場合、その処理は PREPARE の実行時にカーソルを事前 OPEN することによって最適化されます。

ALL PREPARE INTO ステートメントも据え置かれること以外は、YES と同じです。PREPARE ステートメントで SQLDA を戻すために INTO 文節を使用している場合、OPEN、

PRECOMPILE PROGRAM

DESCRIBE、または EXECUTE ステートメントが発行されて戻されるまで、アプリケーションでこの SQLDA の内容を参照してはなりません。

DEGREE

SMP システムで静的 SQL ステートメントを実行するための並列処理の程度を指定します。このオプションは、CREATE INDEX 並列処理には影響を与えません。

1 ステートメントの実行に並列処理を使用しません。

degree-of-parallelism

ステートメントを実行する際の並列処理の程度を指定します。値の範囲は 2 ~ 32 767 です。

ANY ステートメントの実行時にデータベース・マネージャーで判別した程度で並列処理を行うよう指定します。

DISCONNECT

AUTOMATIC

コミット時にすべてのデータベース接続を切断するよう指定します。

CONDITIONAL

RELEASE をマークしたか、またはオープン状態の WITH HOLD カーソルをもたないデータベース接続を、コミット時に切断するよう指定します。

EXPLICIT

RELEASE ステートメントで明示的に解放をマークしたデータベース接続だけを、コミット時に切断するよう指定します。

DYNAMICRULES

許可 ID に使用される値の初期設定、および未修飾オブジェクト参照の暗黙的な修飾の、実行時の動的 SQL に適用される規則を定義します。

RUN パッケージを実行するユーザーの許可 ID が使用されるように指定します。これが省略時値です。

BIND 許可および修飾の静的 SQL に適用されるすべての規則が、実行時に使用されるように指定します。つまり、パッケージ所有者の許可 ID が動的 SQL の権限検査に使用され、省略時のパッケージ修飾子が動的 SQL ステートメント内の未修飾オブジェクト参照の暗黙的な修飾に使用されます。

このオプションを指定してパッケージをバインドする場合、パッケージのバインダーはパッケージのユーザーが受け取るべきでない許可を持つことはできません。動的 SQL ステートメントがパッケージ所有者の許可 ID を使用するためです。次の動的な準備済み SQL ステートメントは、このオプションを指定してバインドされているパッケージ内では使用できません。GRANT、REVOKE、ALTER、CREATE、DROP、COMMENT ON、RENAME、SET CONSTRAINTS、および SET EVENT MONITOR STATE です。

DEFINE

UDF またはストアド・プロシージャ内にある動的 SQL ステートメントの実行に使用する許可識別子が、UDF またはストアド・プロシージャの定義者であるということを指示します。このオプションは、DB2 ではサポートされません。

INVOKE

UDF またはストアド・プロシージャ内にある動的 SQL ステートメントの実行に使用する許可識別子が、UDF またはストアド・プロシージャの呼び出し元であることを指示します。このオプションは、DB2 ではサポートされません。

EXPLAIN

各 SQL ステートメント用に選択したアクセス・プランに関する Explain 表の情報を、パッケージに保管します。DRDA では、このオプションの ALL 値がサポートされていません。

NO Explain 情報はキャプチャーされません。

YES Explain 表には、静的ステートメントの場合は prep/bind 時間で、増分バインド・ステートメントの場合はランタイムで、選択されたアクセス・プランについての情報が取り込まれます。

ALL 適格な静的 SQL ステートメントの Explain 情報が、prep/bind 時間で各 Explain 表に入れられます。適格な増分バインド SQL ステートメントの Explain 情報が、ランタイムで各 Explain 表に入れられます。さらに、CURRENT EXPLAIN SNAPSHOT レジスターが NO に設定されていても、Explain 情報は実行時に適格な動的 SQL ステートメント用に集められます。特殊レジスターの詳細については、SQL 解説書を参照してください。

注: DRDA では、EXPLAIN のこの値 (ALL) はサポートされていません。

PRECOMPILE PROGRAM

EXPLSNAP

Explain 表に Explain スナップショットを保管します。この DB2 プリコンパイル / バインド・オプションは、DRDA ではサポートされていません。

NO Explain スナップショットはキャプチャーされません。

YES Explain 表には、静的ステートメントの場合は prep/bind 時間で、増分バインド・ステートメントの場合はランタイムで、適格な各静的 SQL ステートメントの Explain スナップショットが、Explain 表内に入れられます。

ALL 適格な各静的 SQL ステートメントの Explain スナップショットが、prep/bind 時間で Explain 表内に入れられます。適格な増分バインド SQL ステートメントの Explain スナップショット情報が、ランタイムで各 Explain 表に入れられます。さらに、CURRENT EXPLAIN SNAPSHOT レジスターが NO に設定されていても、Explain スナップショット情報は実行時に適格な動的 SQL ステートメント用に集められます。特殊レジスターの詳細については、SQL 解説書を参照してください。

FEDERATED

パッケージ内の静的 SQL ステートメントが、通称または統合視点を参照するかどうかを指定します。このオプションを指定しており、通称または統合視点が参照されない場合、エラーが戻され、パッケージは作成されません。このオプションが指定されず、通称または統合視点がパッケージ内の静的ステートメントによって参照される場合、エラーが戻され、パッケージは作成されません。

NO ユーティリティは DB2 ユニバーサル・データベースに接続します。これが省略時値です。

YES ユーティリティは DB2 統合システムにアクセスします。

FUNCPATH

静的 SQL で、ユーザー定義の個別タイプおよび機能を解析する際に使用する機能パスを指定します。このオプションを指定しなかった場合、省略時の機能パスは "SYSIBM"、"SYSFUN"、または USER になります。この DB2 プリコンパイル / バインド・オプションは、DRDA ではサポートされていません。

schema-name

SQL 識別子 (通常または区切り)。これは、アプリケーション・サーバーに存在するスキーマを識別します。スキーマが存在する場合、プリコンパイル時やバインド時に妥当性検査は行

われません。同一スキーマは、機能パス内に一度しか存在できません。指定できるスキーマ数は、処理結果の機能パスの長さによって限定され、254 バイトを超えることはできません。スキーマ **SYSIBM** は、明示的に指定する必要がありません。機能パス内に含まれていなければ、最初のスキーマに暗黙的に想定されます。詳細については、*SQL 解説書* を参照してください。

INSERT

DB2 エンタープライズ拡張エディション・サーバーヘプリコンパイルまたはバインドされているプログラムが、パフォーマンス向上のために挿入データをバッファリングすることを要求できるようにします。

BUF アプリケーションからの挿入データをバッファリングすることを指定します。

DEF アプリケーションからの挿入データをバッファリングしないことを指定します。

GENERIC string

宛先 **DRDA** データベースに新規バインド・オプションを渡す手段を提供します。宛先データベースで定義されていても、ローカル・コマンドで認識されない新規バインド・オプションをサポートします。165ページの『**BIND**』または457ページの『**PRECOMPILE PROGRAM**』で定義されているバインド・オプションを渡すようにするには、このオプションを使用しないでください。このオプションは、動的 **SQL** のパフォーマンスをかなり向上させることができます。構文は次のとおりです。

```
generic "option1 value1 option2 value2 ..."
```

各オプションと値は、1 つまたは複数のブランク・スペースで区切らなければなりません。たとえば、宛先 **DRDA** データベースが **DB2 MVS** バージョン 5 の場合、次のようにします。

```
generic "keepdynamic yes"
```

こうして、新規 **keepdynamic YES** オプションは **PRECOMPILE PROGRAM** または **BIND** コマンドでローカルに定義されてはませんが、バインドできます。

ストリングの最大長は 1023 バイトです。この **DRDA** バインド・オプションは、現在 **DB2 MVS** バージョン 5 のみでサポートされています。**DB2** ではサポートされていません。

PRECOMPILE PROGRAM

ISOLATION

このパッケージにバインドされるプログラムを、他の実行プログラムの影響からどの程度分離できるかを指定します。分離レベルの詳細については、*SQL 解説書* を参照してください。

- CS** 分離レベルとしてカーソル固定を指定します。
- NC** コミットしません。コミット制御が使用されないということを指定します。なお、DB2 はこの分離レベルをサポートしません。
- RR** 分離レベルとして反復可能読み取りを指定します。
- RS** 分離レベルとして読み取り固定を指定します。読み取り固定は、パッケージ内での SQL ステートメントの実行を、他のアプリケーションが読み取りおよび変更を行った行に対する処理から分離させます。
- UR** 分離レベルとして非コミット読み取りを指定します。

LANGLEVEL

アプリケーション内の静的および動的 SQL の構文および意味に適用される SQL 規則を指定します。このオプションは、DB2 コネクトではサポートされていません。このオプションについては、*アプリケーション開発の手引き* を参照してください。

- MIA** 以下のように ISO/ANS SQL92 規則を選択します。
- エラー SQLCODE または SQLSTATE の検査をサポートするには、アプリケーション・コードの中で SQLCA が宣言されていなければなりません。
 - C のヌル文字で終了するストリングにはブランクが埋め込まれ、切り捨てが実行された場合でも常にヌル終了文字が含まれます。
 - FOR UPDATE 文節は、定位置 UPDATE で更新されるすべての列において任意指定です。
 - UPDATE または DELETE ステートメントの対象となる表の列が、検索条件の中や代入文節の右辺で参照されているなら、検索 UPDATE または DELETE に、対象となる表に対する SELECT 特権が必要です。
 - 索引を使って解決可能な列関数 (たとえば MIN または MAX) は、ヌルも検査し、ヌルがあったなら警告 SQLSTATE 01003 を戻します。

- CREATE または ALTER TABLE ステートメントの中に重複した固有制約が含まれているなら、エラーが戻されます。
- 特権が何も付与されていない場合、付与者にそのオブジェクトに対する権限がないなら、エラーが戻されます (その権限があるなら警告が戻されます)。

SAA1 以下のように共通 IBM DB2 規則を選択します。

- エラー SQLCODE または SQLSTATE の検査をサポートするには、アプリケーション・コードの中で SQLCA が宣言されていなければなりません。
- C のヌル文字で終了するストリング、切り捨てが実行された場合には、ヌル文字が最後に付けられません。
- FOR UPDATE 文節は、定位置 UPDATE で更新されるすべての列において必須です。
- UPDATE または DELETE ステートメントの対象となる表が、そのステートメントの全選択から参照されるのでなければ、検索 UPDATE または DELETE に、対象となる表に対する SELECT 特権は不要です。
- 索引を使って解決可能な列関数 (たとえば MIN または MAX) は、ヌルを検査せず、警告 SQLSTATE 01003 は戻されません。
- 警告が戻され、固有制約が重複していても無視されます。
- 特権が付与されていないなら、エラーが戻されます。

SQL92E

以下のように ISO/ANS SQL92 規則を定義します。

- SQLCODE または SQLSTATE の検査をサポートするには、その名前の変数をホスト変数の宣言セクションで宣言できます (どちらも宣言されていないなら、事前コンパイル中には SQLCODE が指定されているものとされます)。
- C のヌル文字で終了するストリングには空白が埋め込まれ、切り捨てが実行された場合でも常にヌル終了文字が含まれます。
- FOR UPDATE 文節は、定位置 UPDATE で更新されるすべての列において任意指定です。
- UPDATE または DELETE ステートメントの対象となる表の列が、検索条件の中や代入文節の右辺で参照されているな

PRECOMPILE PROGRAM

ら、検索 UPDATE または DELETE に、対象となる表に対する SELECT 特権が必要です。

- 索引を使って解決可能な列関数 (たとえば MIN または MAX) は、ヌルも検査し、ヌルがあったなら警告 SQLSTATE 01003 を戻します。
- CREATE または ALTER TABLE ステートメントの中に重複した固有制約が含まれているなら、エラーが戻されます。
- 特権が何も付与されていない場合、付与者にそのオブジェクトに対する権限がないなら、エラーが戻されます (その権限があるなら警告が戻されます)。

LEVEL consistency-token

一貫性トークンを使用するモジュールのレベルを定義します。一貫性トークンとは、8 文字までの長さの任意の英数字値のことです。RDB パッケージの一貫性トークンは、リクエスターのアプリケーションと関係データベース・パッケージが同期化されているかどうかを検証します。この DRDA プリコンパイル・オプションは、DB2 ではサポートされていません。

注: このオプションは、通常は使用しないでください。

LONGERROR

長いホスト変数宣言をエラーとして扱うかどうかを示します。移行性のために、sqlint32 は、プリコンパイル C および C++ コードで INTEGER 列の宣言として使用できます。

NO 長いホスト変数宣言の使用に対してエラーを生成しません。これが 32 ビット・システムと 64 ビット NT システムの省略時値です。それらのシステムでは、長いホスト変数を INTEGER 列の宣言として使用できます。このオプションを 64 ビット UNIX プラットフォームで使用すると、長いホスト変数を、BIGINT 列の宣言として使用することができます。

YES 長いホスト変数宣言の使用に対してエラーを生成します。これが 64 ビット UNIX システムの省略時値です。

MESSAGES message-file

警告メッセージ、エラー・メッセージ、および完了状況メッセージの宛先を指定します。メッセージ・ファイルは、パインドが正常であるかどうかによって作成されます。メッセージ・ファイル名を指定しなかった場合、メッセージは標準出力に書き込まれます。ファイルへの完全パス

を指定しなかった場合は、現行ディレクトリーが使用されます。なお、既存ファイルの名前を指定すると、そのファイルの内容は上書きされません。

NOLINEMACRO

出力 .c ファイルでの # 行マクロの生成を抑制します。これは、プロファイル、相互参照ユーティリティー、およびデバッガーのようなソース行情報を必要とする作成ツールでファイルを使用する場合に役に立ちます。

注: このプリコンパイル・オプションは、C/C++ プログラム言語でのみ使用します。

OPTLEVEL

SQL ステートメントの中でホスト変数を使用されている場合に、内部 SQLDA の初期化を C/C++ プリコンパイラーで最適化するかどうかを指示します。そのように最適化すれば、密なループ内で単一の SQL ステートメント (FETCH など) を使う場合のパフォーマンスが向上します。

- 0 SQLDA の初期化を最適化しないようプリコンパイラーに指示します。
- 1 SQLDA の初期化を最適化するようプリコンパイラーに指示します。アプリケーションで以下のものを使っている場合は、この値を指定しないでください。

- 次の例に示すようなポインター・ホスト変数

```
exec sql begin declare section;
char (*name)[20];
short *id;
exec sql end declare section;
```

- 直接 SQL ステートメントに指定される C++ データ・メンバー

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。

OUTPUT filename

コンパイラーが生成した修正済みソース・ファイルの省略時名をオーバーライドします。この名前にはパスを含めることもできます。

OWNER authorization-id

パッケージ所有者の 30 文字の許可識別子を指定します。その所有者には、パッケージに含まれる SQL ステートメントを実行するための特権が必要です。SYSADM または DBADM 許可を持つユーザーのみが、

PRECOMPILE PROGRAM

ユーザー ID 以外の許可識別子を指定できます。省略時値は、プリコンパイル / バインド処理の 1 次許可 ID です。SYSIBM、SYSCAT、および SYSSTAT はこのオプションには無効な値です。

PACKAGE

パッケージを作成します。 **package**、**bindfile**、または **syntax** のどれも指定されていない場合は、省略時で、データベースの中にパッケージが作成されます。

USING package-name

プリコンパイラーが生成するパッケージの名前。名前を入力しないなら、アプリケーション・プログラムのソース・ファイルの名前 (括弧を取り去って大文字に変換したもの) が使われます。最大長は 8 文字です。

PREPROCESSOR "preprocessor-command"

組み込み SQL ステートメントを処理する前に、プリコンパイラーが実行できるプリプロセッサ・コマンドを指定します。プリプロセッサ・コマンド・ストリング (最大 1024 バイト) は、二重引用符または単一引用符で囲む必要があります。

このオプションは、宣言セクションでマクロを使用できるようにします。有効なプリプロセッサ・コマンドとは、コマンド行から発行でき、ソース・ファイルを指定しなくてもプリプロセッサを呼び出せるコマンドです。たとえば、次のように指定します。

```
x1c -P -DMYMACRO=0
```

QUALIFIER qualifier-name

パッケージに含まれる未修飾オブジェクトの 30 文字の暗黙修飾子を指定します。 **owner** が明示的に指定されていれば、その所有者の許可 ID が省略時 ID になります。

QUERYOPT optimization-level

パッケージに含まれるすべての静的 SQL ステートメントに必要な最適化レベルを指示します。省略時値は 5 です。使用可能な最適化レベルの範囲の詳細については、*SQL 解説書* の SET CURRENT QUERY OPTIMIZATION ステートメントを参照してください。この DB2 プリコンパイル / バインド・オプションは、DRDA ではサポートされていません。

RELEASE

リソースを、各 COMMIT ポイントで解放するか、アプリケーション

の終了時に解放するかどうかを指示します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。

COMMIT

各コミット点でリソースを解放します。これは、動的 SQL ステートメントに使用されます。

DEALLOCATE

アプリケーションの終了時にだけリソースを解放します。

SQLCA

FORTTRAN アプリケーションでのみ使用します。その他の言語で使用しても、このオプションは無視されます。

NONE 修正されたソース・コードは SAA 定義と整合性がないことを指定します。

SAA 修正されたソース・コードは SAA 定義と整合性があることを指定します。

SQLERROR

エラーを検出した場合に、パッケージまたはバインド・ファイルを作成するかどうかを指示します。

CHECK

ターゲット・システムがすべての構文を実行し、バインドしている SQL ステートメントの意味を調べるということを指定します。この処理の一部としてパッケージが作成されることはありません。バインド中に、名前とバージョンが同じ既存パッケージを検出した場合、その既存パッケージはドロップも置換 (**ACTION REPLACE** を指定した場合) もされません。

CONTINUE

SQL ステートメントのバインド時にエラーが発生しても、パッケージを作成します。許可または存在などの理由でバインドに失敗したこれらのステートメントは、**VALIDATE RUN** も指定されている場合は、実行時に増分でバインドすることができます。ランタイムでこれらのステートメントを実行しようとする、エラー (SQLCODE -525, SQLSTATE 51015) が生成されます。

NOPACKAGE

エラーを検出した場合、パッケージもバインド・ファイルも作成しません。

PRECOMPILE PROGRAM

SQLFLAG

このオプションに指定された SQL 言語構文との相違点を識別して報告します。

バインド・ファイルまたはパッケージが作成されるのは、**sqlflag** オプションに加えて **bindfile** または **package** オプションが指定されている場合だけです。

ローカル構文検査が実行されるのは、次に示すオプションのうちの 1 つが指定された場合だけです。

- **bindfile**
- **package**
- **sqlerror check**
- **syntax**

sqlflag が指定されていない場合、標識機能は呼び出されず、バインド・ファイルやパッケージが影響を受けることはありません。

SQL92E SYNTAX

SQL ステートメントは、データベース・カタログにアクセスするのに必要な構文規則を除いて、ANSI または ISO SQL92 のエントリー・レベル SQL 言語の書式および構文と比較して調べられます。相違があれば、プリコンパイラー・リストに報告されます。

MVSDB2V23 SYNTAX

SQL ステートメントは、MVS DB2 バージョン 2.3 SQL 言語構文と比較して調べられます。構文の相違があれば、プリコンパイラー・リストに報告されます。

MVSDB2V31 SYNTAX

SQL ステートメントは、MVS DB2 バージョン 3.1 SQL 言語構文と比較して調べられます。構文の相違があれば、プリコンパイラー・リストに報告されます。

MVSDB2V41 SYNTAX

SQL ステートメントは、MVS DB2 バージョン 4.1 SQL 言語構文と比較して調べられます。構文の相違があれば、プリコンパイラー・リストに報告されます。

SQLRULES

下記のことを指定します。

- タイプ 2 の CONNECT を DB2 規則に従って処理するか、それとも ISO/ANS SQL92 の標準 (STD) の規則に従って処理するか。

- ユーザーまたはアプリケーションが LOB 応答セット列の形式をどのように指定するか。

DB2

- SQL CONNECT ステートメントで、現在の接続と、確立されている (休止 状態の) 別の接続との間で切り換えることができるようにします。
- ユーザーまたはアプリケーションは、LOB 列の形式を最初のフェッチ要求においてしか指定できません。

STD

- SQL CONNECT ステートメントでは、新しい 接続を確立することしかできないようにします。休止接続へ切り替えるには、SQL SET CONNECTION ステートメントを使います。
- ユーザーまたはアプリケーションは、LOB 列の形式を各フェッチ要求ごとに指定できます。

SQLWARN

動的 SQL ステートメントの完了時 (PREPARE または EXECUTE IMMEDIATE を通して)、または記述処理時 (PREPARE...INTO または DESCRIBE を通して) に警告を戻すかどうかを指示します。この DB2 プリコンパイル / バインド・オプションは、DRDA ではサポートされていません。

NO SQL コンパイラから警告を戻しません。

YES SQL コンパイラから警告を戻します。

注: SQLCODE +238 は例外です。これは、**SQLWARN** オプションの値が何であろうと戻されます。

STRDEL

SQL ステートメントで使用するストリング区切り文字として、アポストロフィ (') または二重引用符 (") のどちらを使用するか指定します。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。このオプションを指定しないと、DRDA サーバーは、システムが定義した省略時値を使用します。

APOSTROPHE

ストリング区切り文字として、アポストロフィ (') を使用します。

QUOTE

ストリング区切り文字として、二重引用符 (") を使用します。

PRECOMPILE PROGRAM

SYNCPOINT

複数のデータベース接続にまたがってコミットまたはロールバックを調整する仕方を指定します。

NONE 2 フェーズ・コミットを実行するのにトランザクション管理機能 (TM) を使用しないことを指定し、単一更新機構、多重読み取り機構を適用しません。コミットは、関連する各データベースに送られます。コミットが失敗したときの回復は、アプリケーションが行います。

ONEPHASE

2 フェーズ・コミットを実行するのに TM を使用しないことを指定します。複数のデータベース・トランザクションの各データベースが行う作業をコミットするときは、1 フェーズ・コミットが使用されます。

TWOPHASE

このプロトコルをサポートする複数のデータベースにまたがって 2 フェーズ・コミットを調整するのに TM が必要であることを指定します。

SYNTAX

プリコンパイル時に、パッケージまたはバインド・ファイルの作成を抑制します。このオプションを使用すれば、既存のパッケージまたはバインド・ファイルを修正したり変更したりしないで、ソース・ファイルの妥当性を検査できます。 **Syntax** は **sqlerror check** の同義語です。

syntax を **package** オプションと一緒に使うと、**package** は無視されます。

TARGET

現行のプラットフォームでサポートされているコンパイラーの 1 つに合わせて調整した修正コードを生成するように、プリコンパイラーに指示します。

IBMCOB

AIX では、IBM COBOL Set (AIX 版) のコンパイラーのためのコードが生成されます。 OS/2 では、IBM VisualAge for COBOL コンパイラー用のコードが生成されます。

MFCOB

Micro Focus COBOL コンパイラー用のコードが生成されます。 OS/2 では、32 ビット Micro Focus COBOL コンパイラー用です。すべての UNIX プラットフォームおよび Windows

NT の COBOL プリコンパイラーにおいて、**target** 値が指定されないなら、この値が省略時値になります。

MFCOB16

16 ビット Micro Focus COBOL コンパイラー用のコードが生成されます。この値は OS/2 でのみ有効であり、COBOL プリコンパイラーで **target** 値を指定しない場合の省略時値です。

ANSI_COBOL

ANS X3.23-1985 標準規格と互換性のあるコードが生成されます。

C 現行プラットフォーム上の DB2 によりサポートされる C コンパイラーと互換性のあるコードが生成されます。

CPLUSPLUS

現行プラットフォーム上の DB2 によりサポートされる C++ コンパイラーと互換性のあるコードが生成されます。

BORLAND_C

Borland C/C++ コンパイラー用の C コードが生成されます。この値は、OS/2 でのみ有効です。

BORLAND_CPLUSPLUS

Borland C/C++ コンパイラー用の C++ コードが生成されます。この値は、OS/2 でのみ有効です。

FORTTRAN

現行プラットフォーム上の DB2 によりサポートされる FORTRAN コンパイラーと互換性のあるコードが生成されません。

TEXT label

パッケージの記述。最大長は 255 文字です。また、省略時値は空白です。なお、DB2 はこの DRDA プリコンパイル / バインド・オプションをサポートしません。

TRANSFORM GROUP

静的 SQL ステートメントが、ユーザー定義の構造型の値をホスト・プログラムと交換するために使用する、変換グループ名を指定します。この変換グループは、動的 SQL ステートメントには使用されません。また、パラメーターの交換や外部関数またはメソッドの結果にも使用されません。

groupname

SQL 識別子。長さは最大で 18 文字です。グループ名には、修飾子接頭部を含めることはできません。また、接頭部 **SYS** はデータベースで使用するために予約されているので、その接頭部は使用できません。ホスト変数とやりとりする静的 SQL ステートメントでは、構造型の値の交換に使用する変換グループの名前は以下ようになります。

- **TRANSFORM GROUP** バインド・オプション内のグループ名 (もしあれば)
- **TRANSFORM GROUP** 準備オプション内のグループ名。最初の事前コンパイル時に指定したとおりのもの (もしあれば)
- **DB2_PROGRAM** グループ。グループ名が **DB2_PROGRAM** の、特定のタイプに対する変換がある場合。
- 上記のいずれの条件もない場合には、変換グループは使用されません。

静的 SQL ステートメントのバインド時には、以下のエラーが発生する可能性があります。

- **SQLCODE yyy, SQLSTATE xxxxx**: 変換が必要ですが、静的変換グループが選択されていません。
- **SQLCODE yyy, SQLSTATE xxxxx**: 選択された変換グループには、交換するデータ・タイプに必要な変換が含まれていません (入力変数用の **TO SQL**、出力変数用の **FROM SQL**)。
- **SQLCODE yyy, SQLSTATE xxxxx**: **FROM SQL** 変換の結果タイプは、出力変数のタイプと互換性がありません。または、**TO SQL** 変換のパラメーター・タイプは、入力変数のタイプと互換性がありません。

VALIDATE

データベース・マネージャーが、許可エラーとエラー未検出のオブジェクトをいつ検査するかを判別します。この妥当性検査には、パッケージ所有者の許可 **ID** が使用されます。

BIND プリコンパイル / バインド時に妥当性検査が実行されます。オブジェクトが 1 つもない場合、または権限がまったく保持されていない場合、エラー・メッセージが作成されます。 **SQLERROR CONTINUE** を指定した場合、エラー・メッセージにかかわらずパッケージ / バインド・ファイルは作成されますが、エラーのあるステートメントは実行できません。

RUN バインド時に妥当性検査が行われます。すべてのオブジェクトが存在し、全権限が保持されていれば、それ以上実行しても検査は行われません。

プリコンパイル / バインド時にオブジェクトが 1 つもない場合、または権限がまったく保持されていない場合、**SQLERROR CONTINUE** オプションの設定とは無関係に警告メッセージが作成されて、パッケージは正常にバインドされます。ただし、プリコンパイル / バインド処理時に SQL ステートメントの権限検査と存在検査に障害が生じた場合、実行時に再実行される可能性があります。

VERSION version-id

パッケージのバージョン識別子を定義します。バージョン識別子は、任意の英数字値、\$、#、@、_、-、または . で、長さは 254 バイト以下です。この DRDA プリコンパイル・オプションは、DB2 ではサポートされていません。

WCHARTYPE

wchartype の使用と適用についての詳細および制約事項は、[アプリケーション開発の手引き](#) を参照してください。

CONVERT

wchar_t 基本タイプを使って宣言されたホスト変数には、**wchar_t** 形式のデータが入れられるものとして扱われます。この形式は、データベースに格納されるグラフィック・データの形式 (DBCS 形式) と直接に互換性はないので、**wchar_t** ホスト変数の入力データは、ANSI C 関数 **wcstombs()** を使って DBCS 形式に暗黙のうちに変換されます。同じように出力 DBCS データは、ホスト変数を保管する前に、**mbstowcs()** を使用して **wchar_t** 形式に暗黙のうちに変換されます。

NOCONVERT

wchar_t 基本タイプを使って宣言されたホスト変数には、DBCS 形式のデータが入れられるものとして扱われます。これは、データベースの中でグラフィック・データ用に使われる形式ですが、C 言語で採用されている固有の **wchar_t** 形式とは違うものです。

noconvert を使用する場合、グラフィック・データはアプリケーションとデータベースの間で変換されないことになり、それによって効率が改善されます。しか

し、アプリケーション側では、データベース・マネージャに `wchar_t` 形式のデータが渡されることがないようにしなければなりません。このオプションを使用する場合は、`wchar_t` ホスト変数を C ワイド文字列関数で処理しないようにし、ワイド文字リテラル (*L-literals*) で初期化しないようにしなければなりません。

使用上の注意

修正されたソース・ファイルが作成されます。それには SQL ステートメントに相当するホスト言語ステートメントが入っています。パッケージは、省略時には、接続が確立されているデータベース内に作成されます。パッケージの名前は、ファイル名と同じ (拡張子を取り去って大文字に変換したもの) であり、最大 8 文字までです。

データベースへの接続が終わると、開始されているトランザクションの下で PREP が実行されます。次に PREP は COMMIT または ROLLBACK を実行し、現行トランザクションを終了して別のトランザクションを開始します。

すでに存在していないスキーマ名を指定してパッケージを作成すると、そのスキーマが暗黙のうちに作成されます。スキーマの所有者は SYSIBM になります。スキーマに対する CREATEIN 特権が PUBLIC に付与されます。

プリコンパイル時には、パッケージが作成されて **explsnap** が指定されているのでない限り、**Explain** スナップショットは取られません。スナップショットは、パッケージを作成するユーザーの **Explain** 表に入れられます。同じように、**Explain** 表情報が取得されるのは、**explain** が指定されていて、パッケージが作成される場合だけです。

致命的エラーが発生するか、100 以上のエラーが発生すると、プリコンパイルは停止します。致命的エラーが発生すると、ユーティリティはプリコンパイルを停止し、すべてのファイルをクローズしてからパッケージを廃棄します。

パッケージが **dynamicrules** バインドでバインドされる場合、バインド・オプション **owner** の暗黙または明示的な値は、動的 SQL ステートメントの許可検査に使用され、バインド・オプション **qualifier** の暗黙または明示的な値は、動的 SQL ステートメント内の未修飾オブジェクトの暗黙修飾子として使用されます。複数のパッケージが単一の接続中に参照される場合、特定のパッケージにより準備された動的 SQL は、そのパッケージへのバインド・オプシ

ョンにしたがって動作します。特殊レジスター CURRENT SCHEMA の値は、**dynamicrules** バインドを持つパッケージ結合にある修飾子には影響しません。

クライアント / サーバー環境で、PRECOMPILE コマンドがバージョン 6 のサーバーに接続されたバージョン 6 より前のクライアントに発行される場合、エラーを示す SQL0020W がクライアントに戻される場合があります。この状態は、コマンドが DYNAMICRULES または OWNER パラメーターの一方または両方を使用する場合に起きます。エラー・メッセージが表示されても、それらのパラメーターは引き続き有効であり、使用されます。SQL0020W メッセージに表示される他のパラメーターは正しくエラーと示され、メッセージが示すとおりは無視されます。これらの他のパラメーターには、以下のものが含まれます。ACTION、CCSIDG、CCSIDM、CCSIDS、CHARSUB、CNULREQD、DEC、DECDEL、GENERIC、RELEASE、REPLVER、RETAIN、STRDEL、TEXT、VALIDATE、VERSION、および値 CONTINUE が指定された SQLERROR。

同じように、バージョン 6 以前のクライアントがバージョン 7 のサーバーに接続されると、DYNAMICRULES、OWNER、または VALIDATE パラメーターのいずれかを組み合わせて使用すると、SQL0020W エラーが示されます。エラー・メッセージが表示されても、それらのパラメーターは引き続き有効であり、使用されます。SQL0020W メッセージに表示される他のパラメーターは正しくエラーと示され、メッセージが示すとおりは無視されます。これらの他のパラメーターには、以下のものが含まれます。ACTION、CCSIDG、CCSIDM、CCSIDS、CHARSUB、CNULREQD、DEC、DECDEL、GENERIC、RELEASE、REPLVER、RETAIN、STRDEL、TEXT、VERSION、および値 CONTINUE が指定された SQLERROR。

注: バージョン 7 またはそれ以降のサーバーに接続されており、VALIDATE RUN が指定されており、エラーを生成する原因となるがこの BIND オプションによってバイパスされるステートメントがある間は、バージョン 6 以前のクライアントはバインド・ファイルを生成しません。ただし、パッケージは作成されます。

SQL ステートメントがエラーであることが検出され、PRECOMPILE オプション SQLERROR CONTINUE が指定されている場合、このステートメントは無効とマークされます。この SQL ステートメントの状態を変えるには、さらに別の PRECOMPILE を発行する必要があります。暗黙的および明示的な再バインドでは、無効なステートメントの状態は変わりません。VALIDATE RUN でバインドされたパッケージでは、ステートメントは、再バインド時にオブジェクトが存在するかまたは権限の問題があるかどうかに応じて、暗黙的と明示

PRECOMPILE PROGRAM

的な再バインドとの両方で、静的バインドから増分バインドに変更したり、増分バインドを静的バインドに変更することができます。

以下も参照

165ページの『BIND』

PRUNE HISTORY/LOGFILE

回復活動記録ファイルから項目を削除したり、アクティブ・ログ・ファイル・パスからログ・ファイルを削除したりするのに使用します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

必須接続

データベース

コマンド構文



コマンド・パラメーター

HISTORY *timestamp*

削除される、回復活動記録ファイルにある項目範囲を識別します。完全なタイム・スタンプ (書式 *yyyymmddhhmmss*)、または最初の接頭部 (最低限、*yyyy*) を指定できます。提供されているそのタイム・スタンプ以下のタイム・スタンプ付きのすべての項目は、回復活動記録ファイルから削除されます。

WITH FORCE OPTION

最新の復元セットのいくつかの項目がファイルから削除されるとしても、指定したタイム・スタンプに従って項目を枝取りすることを指定します。復元セットは、バックアップ・イメージのすべての復元を含む、最新の全データベース・バックアップです。このパラメーターを指定しない場合、バックアップ・イメージ転送からのすべての項目は活動記録で保守されます。

LOGFILE PRIOR TO *log-file-name*

ログ・ファイル名を表すストリング (例: *S0000100.LOG*) を指定します。指定したログ・ファイルより前のすべてのログ・ファイルは削除さ

PRUNE HISTORY/LOGFILE

れます。指定したログ・ファイルそのものは削除されません。
LOGRETAIN データベース構成パラメーターは、RECOVERY または
CAPTURE に設定する必要があります。

例

前に行われた、すべての復元、ロード、表スペース、バックアップ、および全部のデータベース・バックアップのための項目を除去するには、回復活動記録ファイルから 1994 12.1 を含んで、次のように入力してください。

```
db2 prune history 199412
```

注: 199412 は 19941201000000 と解釈されます。

使用上の注意

活動記録ファイルからバックアップ項目を枝取りすると、DB2 データ・リンク・マネージャー・サーバー上にある関連ファイルのバックアップが削除されます。

QUERY CLIENT

アプリケーション処理用の現行接続設定に戻ります。

許可

なし

必須接続

なし

コマンド構文

▶—QUERY CLIENT—▶

コマンド・パラメーター

なし

例

以下に示すのは、QUERY CLIENT の出力例です。

The current connection settings of the application process are:

```

CONNECT      = 1
DISCONNECT   = EXPLICIT
MAX_NETBIOS_CONNECTIONS = 1
SQLRULES     = DB2
SYNCPOINT    = ONEPHASE
CONNECT_NODE  = CATALOG_NODE
ATTACH_NODE   = -1

```

CONNECT_NODE および ATTACH_NODE が SET CLIENT コマンドを使用して設定されていない場合、これらのパラメーターの値は環境変数 DB2NODE のものと同じになります。CONNECT_NODE または ATTACH_NODE パラメーターの表示値が -1 である場合、パラメーターは設定されていません。つまり、環境変数 DB2NODE が設定されなかったか、あるいは以前に発行された SET CLIENT コマンドでパラメーターが指定されませんでした。

使用上の注意

アプリケーション処理用の接続設定は、実行中にいつでも照会できます。

分散作業単位 (DUOW) の詳細については、[管理の手引き](#) を参照してください。

QUERY CLIENT

以下も参照

566ページの『SET CLIENT』

QUIESCE TABLESPACES FOR TABLE

表の表スペースを静止させます。共用、更新意図、排他の 3 つの有効な静止モードがあります。静止機能の結果として生じる状態には、次の 3 つの状態、QUIESCED SHARE、QUIESCED UPDATE、および QUIESCED EXCLUSIVE があります。

効力範囲

単一区画環境では、ロード操作中に排他モードのロード操作を起動すると、このコマンドは表スペースをすべて静止します。区分データベース環境では、このコマンドはノードでローカルに活動します。このコマンドは、ロード操作を実行しているノードに属する表スペースの部分のみを静止します。

許可

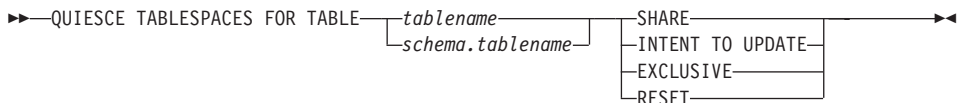
以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- *load*

必須接続

データベース

コマンド構文



コマンド・パラメーター

TABLE

tablename

非修飾表名を指定します。システム・カタログ表を指定することはできません。

QUIESCE TABLESPACES FOR TABLE

schema.tablename

修飾表名を指定します。 *schema* が指定されない場合には、CURRENT SCHEMA が使用されます。システム・カタログ表を指定することはできません。

SHARE

静止が共用モードであることを指定します。

『静止モードでの共用』 要求を行うと、トランザクションは、表スペースに対して専用共用ロックを、および表に対して共用ロックを要求します。トランザクションがロックを獲得すると、表スペースの状態が QUIESCED SHARE に変更されます。他のユーザーによって保留されているような対立状態がない場合には、その状態は、そのユーザーにのみ付与されます。表スペースの状態は、その状態が持続されるように、許可 ID およびそのユーザーのデータベース・エージェント ID とともに、表スペースにある表に記録されます。表の表スペースが QUIESCED SHARE 状態である間は、その表を変更できません。表および表スペースに要求するその他の共用モードは、認められます。トランザクションがコミットまたはロールバックされる際、ロックは解放されますが、その表の表スペースはその状態が明示的にリセットされるまで、QUIESCED SHARE 状態のまま残ります。

INTENT TO UPDATE

静止モードが更新意図モードであることを指定します。

『静止モードでの更新意図』 要求を行うと、表スペースは意図排他 (IX) モードでロックされ、表は更新 (U) モードでロックされます。表スペースの状態は、表スペースの表に記録されます。

EXCLUSIVE

静止が排他モードであることを指定します。

『静止モードでの排他』 要求を行うと、トランザクションは、表スペースに対する特別な排他ロックと、表に対する特別な排他ロックを要求します。トランザクションがロックを獲得すると、表スペースの状態が QUIESCED EXCLUSIVE に変更されます。表スペースの状態は、許可 ID およびそのユーザーのデータベース・エージェント ID とともに、表スペースにある表に記録されます。表スペースは、特別な排他モードで保護されているため、その表スペースへのアクセスが認められているその他のアクセスはありません。静止機能呼び出すユーザー (静止者) は、その表と表スペースへの排他的アクセスを行うことができます。

RESET

表スペースの状態が、正常にリセットされることを指定します。

例

```
db2 quiesce tablespaces for table staff share
```

```
db2 quiesce tablespaces for table boss.org intent to update
```

使用上の注意

このコマンドは、宣言一時表に対してはサポートされていません。

静止は持続ロックです。その利点は、それがトランザクション障害、接続障害、およびシステム障害（電源障害や、リブートなど）が生じても持続することです。

静止は接続によって所有されます。接続が失われた場合、静止は残りますが、非所有の状態に移り、ファントム静止 と呼ばれます。ファントム静止は、同じ表スペースまたは表に対して、**QUIESCE TABLESPACES FOR TABLE** コマンドを発行した次の接続によって『所有』されるようになります。たとえば、削除フェーズ中に停電によってロード操作が割り込まれると、ロードされていた表の表スペースは削除保留、静止排他状態で残ります。データベースの再始動時に、この静止は非所有（ファントム）の状態になります。

ファントム静止を取り除くには、次のようにします。

1. データベースに接続する。
2. **LIST TABLESPACES** コマンドを使用して、静止させる表スペースを決定する。
3. 現行の静止状態を使用して、表スペースを再静止させる。たとえば、次のようにします。

```
db2 quiesce tablespaces for table mytable exclusive
```

完了すると、新しい接続が静止を所有するようになり、ロード操作を再開できるようになります。

いつでも、表スペース上での静止者の限度は 5 つです。

静止者は表スペースの状態を、制限の少ない状態から、より制限のある状態（たとえば、S から U へ、または U から X へ）へアップグレードすることができます。ユーザーがすでに保持されている状態より低い状態を要求すると、元の状態に戻されます。状態は、ダウングレードされません。

QUIESCE TABLESPACES FOR TABLE

以下も参照

411ページの『LOAD』

QUIT

コマンド行プロセッサの対話式入力モードを終了し、オペレーティング・システムのコマンド・プロンプトに戻ります。バッチ・ファイルが、コマンド行プロセッサにコマンドを入力するのに使用されている場合には、QUIT、TERMINATE、またはファイルの終わりが検出されるまで、コマンドは処理されます。

許可

なし

必須接続

なし

コマンド構文

▶▶—QUIT—▶▶

コマンド・パラメーター

なし

使用上の注意

QUIT はコマンド行プロセッサ・バックエンド・プロセスを終了しないか、データベース接続を中断します。CONNECT RESET は、接続を中断しますが、バック・エンド・プロセスを終了しません。586ページの『TERMINATE』は両方を行います。

REBIND

ユーザーが、バインド・ファイルを必要とせずに、データベースに格納されているパッケージを、再作成することができます。

許可

以下のどれかが必要です。

- *sysadm* または *dbadm* の権限
- スキーマに対する ALTERIN 特権
- パッケージに対する BIND 特権

SYSCAT.PACKAGES システム・カタログ表の BOUNDBY 列にログに記録した許可 ID は、パッケージの最新のバインダーの ID であり、再バインド用のバインダー許可 ID として使用されます。また、パッケージの表参照のための省略時スキーマ としても使用されます。この省略時修飾子は、ユーザーが実行する再バインド要求の許可 ID と、異なる可能性があることに注意してください。REBIND は、パッケージが作成された時に指定されたのと同じバインド・オプションを使用します。

必須接続

データベース。データベース接続が存在しない場合で、暗黙の接続が使用可能な場合には、省略時データベースへの接続が行われます。

コマンド構文

```
REBIND [PACKAGE] package-name [/RESOLVE {ANY | CONSERVATIVE}]
```

コマンド・パラメーター

PACKAGE package-name

再バインドされるパッケージを指定する修飾されている、または修飾されていない名前。修飾されていないパッケージ名は、暗黙的に現在の許可 ID によって修飾されます。

RESOLVE

パッケージの再バインドの実行に、従来のバインド・セマンティクスを使用するかどうかを指定します。これは、パッケージの静的 DML ステートメントの関数およびタイプの解決時に、新しい関数およびデータ・タイプを対象にするかどうかに影響します。このオプションは DRDA ではサポートされていません。有効な値は以下のとおりです。

ANY 関数およびタイプの解決時に、SQL パスにあるすべての関数とタイプを対象にします。従来のバインド・セマンティクスは使用されません。これが省略時値です。

CONSERVATIVE

関数およびタイプの解決時に、最後の明示的バインドのタイム・スタンプより前に定義された SQL パスにある関数とタイプだけを対象にします。従来のバインド・セマンティクスを使用します。

使用上の注意

REBIND は、正常な再バインドに続いて、トランザクションを自動的にコミットしません。ユーザーは、トランザクションを明示的にコミットする必要があります。これにより、ユーザーがある統計を更新する場合の、「what if」分析を使用可能にして、変更した内容を見るために、パッケージを再バインドしようとしています。また、作業単位内の複数の再バインドも許可しています。

注: REBIND コマンドは、自動コミットが使用可能な場合には、トランザクションをコミットします。

このコマンドは以下の事柄を行います。

- パッケージを再作成する簡易的な方法を提供します。これによりユーザーは、元のバインド・ファイルを必要とせずに、システムにおける変更を利用することができます。たとえば、特定の SQL 言語が新しく作成された索引を利用できるような場合には、REBIND コマンドがパッケージを再作成するのに使用できます。REBIND は、561ページの『RUNSTATS』が実行された後に、新規統計を利用してパッケージを再作成するためにも使用することができます。
- 作動不能パッケージを再作成する方法を提供します。作動不能パッケージは、バインド・ユーティリティまたは再バインド・ユーティリティのどちらかを呼び出して、明示的に再バインドする必要があります。パッケージが依存する機能インスタンスがドロップされると、パッケージは作動不能とマークされます (SYSCAT.PACKAGES システム・カタログの VALID 列は、X と設定されます。)
- ユーザーに無効パッケージの再バインド以上の制御を与えます。無効パッケージは、実行される際にデータベース・マネージャーによって、自動的に (または暗黙的に) 再バインドされます。これは、その結果無効パッケージの最初の SQL 要求の実行を遅らせる可能性があります。暗黙の再バインドが失敗した場合に、最初の遅延を無くし、戻される予期していない SQL エラー・メッセージを防ぐためには、システムが自動的に再バインドできるよう

にする以外に、無効なパッケージを明示的に再バインドすることが必要とされます。たとえば、移行に続いて、データベースに格納されているすべてのパッケージが、DB2 バージョン 5 の移行作業によって無効にされます。これは多数のパッケージを含んでいる場合には、一度に無効パッケージのすべてを明示的に再バインドする必要があります。この明示的な再バインドは、**BIND**、**REBIND**、または **db2rbind** ツール (98ページの『db2rbind - すべてのパッケージの再バインド』を参照) を使用して行うことができます。

パッケージを明示的に再バインドするのに **BIND** と **REBIND** のどちらを使用するかは、その環境によります。**REBIND** のパフォーマンスが **BIND** のパフォーマンスよりかなり良いため、**BIND** の使用を明示的に必要としない場合はいつでも、**REBIND** を使用するように推奨されています。しかしながら、次の場合には、**BIND** は使用されなければなりません。

- プログラムを修正した場合 (たとえば、SQL 言語が追加または削除された場合、またはパッケージがそのプログラムの実行可能モジュールと一致しない場合)。
- ユーザーが再バインドの部分としてバインド・オプションのどれかを修正する場合。**REBIND** は、バインド・オプションをサポートしません。たとえば、ユーザーが、バインド・プロセスの部分として付与されたパッケージの特権を所有したい場合、**BIND** が、**grant** オプションを持っているために、それを使用しなければなりません。
- パッケージが現在データベースに存在しない場合。
- すべてのバインド・エラーの検出が必要な場合。**REBIND** は、検出される最初のエラーのみ戻しますが、**BIND** コマンドはバインド中に発生する、最初の 100 のエラーを戻します。

REBIND は、DB2 コネクトによってサポートされます。

REBIND が他のユーザーが使用中のパッケージで実行された場合、他のユーザーの論理装置の作業が終了するまで、再バインドは起こりません。再バインドの間、**SYSCAT.PACKAGES** システム・カタログ表中のパッケージのレコードで排他ロックが保留になっているからです。

REBIND を実行する際、データベース・マネージャーは、**SYSCAT.STATEMENTS** システム・カタログ表に保管されている SQL 文からパッケージを再作成します。

REBIND がエラーを検出すると、処理を停止してエラー・メッセージが戻されます。

REBIND は、**explsnap** バインド・オプションが YES または ALL に設定されて作成されたパッケージ (値の設定は、SYSCAT.PACKAGES カタログ表項目の EXPLAIN_SNAPSHOT 列で示されます)、あるいは **explain** バインド・オプションが YES または ALL に設定されて作成されたパッケージ (値の設定は、SYSCAT.PACKAGES カタログ表項目の EXPLAIN_MODE 列で示されます) に再び Explain を実行します。使用される Explain 表は、REBIND コマンド要求者のものであり、最初のバインド・コマンド発行者のものではありません。

SQL ステートメントがエラーであることが検出され、BIND オプションの SQLERROR CONTINUE を指定していた場合、問題が訂正されたとしても、そのステートメントには無効というマークが付けられます。REBIND によっても、ステートメントが無効の状態は変更できません。VALIDATE RUN でバインドされたパッケージでは、ステートメントは、REBIND 実行時にオブジェクトが存在するかまたは権限の問題があるかどうかに応じて、REBIND を通じて静的バインドから増分バインドに変更したり、増分バインドを静的バインドに変更することができます。

以下も参照

165ページの『BIND』

98ページの『db2rbind - すべてのパッケージの再バインド』

561ページの『RUNSTATS』

RECONCILE

RECONCILE

表の DATALINK データについてファイルへの参照を妥当性検査します。ファイルへの参照を確立できない行が例外表 (指定されている場合) にコピーされ、入力ファイルで変更されます。

RECONCILE を使用すると、UNIX ベース・システムのインスタンス・パスと Windows NT および OS/2 のインストール・パスに、メッセージ・ファイル (reconcil.msg) が生成されます。このファイルには、例外表の妥当性検査で生成された警告およびエラー・メッセージが入ります。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- 表に対する CONTROL 特権

必須接続

データベース

コマンド構文

```
▶▶ RECONCILE—table-name—DLREPORT—filename—┬──────────────────────────────────▶▶  
└──FOR EXCEPTION—table-name──┘
```

コマンド・パラメーター

RECONCILE *table-name*

調整が実行される表を指定します。別名、完全修飾、または未修飾の表名は指定できます。修飾子付き表名は、*schema.tablename* の形式です。未修飾の表名を指定すると、その表は現行許可 ID で修飾されます。

DLREPORT *filename*

調整時にリンク解除されたファイルに関する情報が入るファイルを指定します。名前は完全修飾する必要があります (例: /u/johnh/report)。調整ユーティリティーは、指定されたファイル名に *.ulk* 拡張子を付け加えます (例:report.ulk)。

FOR EXCEPTION table-name

DATALINK 値が関係したリンク障害が発生した行のコピー先となる例外表を指定します。

例

以下のコマンドを実行すると、表 DEPT の調整が行われ、ユーザーがすでに作成している例外表 EXCPTAB に例外が書き込まれます。調整時にリンク解除されたファイルについての情報は、 /u/johnh ディレクトリーに作成される report.ulk ファイルに書き込まれます。FOR EXCEPTION excptab を指定しなかった場合は、 /u/johnh ディレクトリーに作成される report.exp ファイルに例外情報が書き込まれます。

```
db2 reconcile dept dlreport /u/johnh/report for exception excptab
```

使用上の注意

調整時には、存在するファイルのリンクは表データに従って試みられますが、存在しないファイルのリンクはデータ・リンク・ファイル・マネージャーのメタデータに従って試みられます (他の矛盾が存在しなければ)。

調整は表内のすべての DATALINK データに関して実行されます。ファイル参照を再確立できない場合は、違反行が例外表 (指定されている場合) に挿入されます。これらの違反行は入力表から削除されます。ファイル参照の保全性を保証するため、問題となっている DATALINK 値はヌルにされます。列がヌル不可として定義されている場合には、DATALINK 値が長さゼロの URL によって置換されます。

例外表を指定しなかった場合、ファイル参照を再確立できなかった DATALINK 列値は、列 ID および注釈とともに、例外報告ファイル (<filename>.exp) にコピーされます。

調整処理の終了時に、表はデータ・リンク調整保留 (DRP) 状態から解放されます。

例外表 (指定されている場合) を作成してからでないと、調整ユーティリティーは実行できません。調整ユーティリティーで使用される例外表は、ロード・ユーティリティーで使用される例外表と同じです。

例外表は、調整する表の定義を模倣します。例外表では、データ列の後に 1 つか 2 つのオプション列を加えることもできます。最初のオプション列は、TIMESTAMP 列です。この列には、調整操作が開始された時刻のタイム・スタンプが含まれます。2 番目のオプション列は、タイプ CLOB (32KB 以上) に

RECONCILE

する必要があります。この列には、リンク障害の発生した列の ID と、その障害の理由が含まれます。例外表の DATALINK 列は NO LINK CONTROL を指定しなければなりません。そのように指定することにより、行 (DATALINK 列を含む) が挿入されてもファイルはリンクされず、例外表から行が選択されてもアクセス・トークンが生成されずに済みます。

MESSAGE 列の情報は、以下の構造に従って編成されます。

フィールド 番号	内容	サイズ	注釈
1	違反の数	5 文字	右寄せ '0' を埋め込む
2	違反のタイプ	1 文字	'L' - DATALINK 違反
3	違反の長さ	5 文字	右寄せ '0' を埋め込む
4	違反している DATALINK 列の数	4 文字	右寄せ '0' を埋め込む
5	DATALINK 列番号 (最初の違反列の)	4 文字	右寄せ '0' を埋め込む
6	違反の理由	5 文字	右寄せ '0' を埋め込む

フィールド 5 と
6 を、それぞれの
違反列ごとに繰り返す

以下に示すのは、可能性のある違反のリストです。

- 00001 - DB2 データ・リンク・マネージャーがファイルを見つけられない。
- 00002 - ファイルはすでにリンクされている。
- 00003 - ファイルは変更された状態にある。
- 00004 - 接頭部の名前が登録されていない。
- 00005 - ファイルが検索できなかった。
- 00006 - File entry missing. This will happen for
recovery = no, partial control 1 DATALINK
columns. Use update to relink the file.
- 00007 - ファイルがリンク解除状態にある。
- 00008 - 復元されたが変更されていないファイルは
<filename>.MOD にコピーされている。
- 00009 - ファイルはすでに別の表にリンクされている。
- 00999 - ファイルをリンクできなかった。

例:

- 00001L000120002000400002000500001
- 00001 - 違反の数が 1 であることを指定する。
- L - 違反のタイプが 'DATALINK 違反' であることを指定する。
- 00012 - 違反の長さが 12 バイトであることを指定する。
- 0002 - リンク違反を検出した行には 2 つの列があることを指定する。
- 0004,00002
- 0005,00001 - 列 ID と違反の理由を指定する。

RECONCILE

メッセージ列がある場合は、タイム・スタンプ列もあるはずです。例外表の詳細については、*SQL 解説書* を参照してください。

REDISTRIBUTE NODEGROUP

ノード・グループ内の全ノードにデータを再配布します。均一なあるいは偏った現行のデータ配分を指定できます。再配布アルゴリズムは、現行のデータ配分に基づいて移動する区分を選択します。

このコマンドは、カタログ・ノードからしか発行できません。369ページの『LIST DATABASE DIRECTORY』を使用して、各データベースのカタログ・ノードがどのノードかを判別してください。

効力範囲

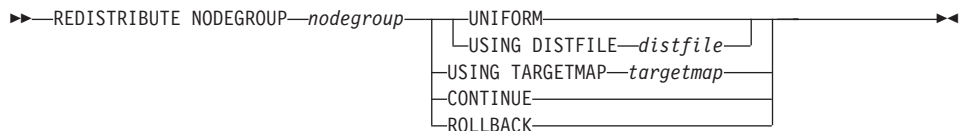
このコマンドは、ノード・グループ内のすべてのノードに影響を与えます。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *dbadm*

コマンド構文



コマンド・パラメーター

NODEGROUP nodegroup

ノード・グループの名前。この 1 部分名は、SYSNODEGROUPS カタログ表に記述されたノード・グループを識別します。ノード・グループは、現在再配布を受けることはできません。

注: IBMCATGROUP および IBMTEMPGROUP ノード・グループ内の表を再配布することはできません。

UNIFORM

データがハッシュ区分にわたって均等に配布されることを指定します (つまり、それぞれのハッシュ区分が同じ数の行を持つことが想定されます)。しかし、それぞれのノードに同じ数のハッシュ区分はマップさ

REDISTRIBUTE NODEGROUP

れません。再配布後、ノード・グループのすべてのノードは、ほぼ同じ数のハッシュ区分を持っています。

USING DISTFILE *distfile*

区分化キー値の配布がスキューである場合、このオプションを使用して、ノード・グループのノード全体にわたるデータに均一な再配布を行います。

distfile を使用して、4 096 個のハッシュ区分にわたる現行のデータの配布を指示します。

行カウント、バイト・ボリューム、または他の任意のメジャーを使用して、各ハッシュ区分で表示されたデータ量を示します。ユーティリティは、区分に関連する整数値をその区分の重みとして読み取ります。*distfile* を指定した場合、ユーティリティは宛先区分化マップを生成し、ノード・グループのノード全体にデータをできるだけ均一に再配布するために使用します。再配布の後、ノード・グループ中の各ノードの重みは、ほぼ同じになります (ノードの重みは、ノードにマップしたすべての区分の重みの合計です)。

たとえば、入力配布ファイルに以下の項目があるとします。

```
10223
1345
112000
0
100
...
```

例の中で、ハッシュ区分 2 は 112 000 の重みを持ち、区分 3 (重さは 0) には、マッピングするデータがまったくありません。

distfile には、4 096 の正整数値が文字形式で入っているとします。値の合計は、4 294 967 295 以下である必要があります。

distfile のパスが指定されていない場合、現行ディレクトリーが使用されます。

USING TARGETMAP *targetmap*

targetmap で指定されたファイルは、宛先区分化マップとして使用されます。データの再配布はこのファイルに従って行われます。パスを指定していない場合、現行ディレクトリーが使用されます。

宛先マップに含まれるノードがノード・グループ中に存在しない場合、エラーが戻されます。ALTER NODEGROUP ADD NODE を発行してから、REDISTRIBUTE NODEGROUP を実行してください。

宛先マップから実行したノードが、ノード・グループにある場合、そのノードは区分の中に含まれていません。そのようなノードは、REDISTRIBUTE NODEGROUP を発行する前であっても後であっても、ALTER NODEGROUP DROP NODE を使用して消去できます。

CONTINUE

直前に失敗した REDISTRIBUTE NODEGROUP 操作を継続します。何も起こらなければ、エラーが戻されます。

ROLLBACK

直前に失敗した REDISTRIBUTE NODEGROUP 操作をロールバックします。何も起こらなければ、エラーが戻されます。

使用上の注意

再配布操作が行われると、メッセージ・ファイルは次のディレクトリーに書き込まれます。

- UNIX ベースのシステムの場合、/sql1lib/redis ディレクトリー。サブディレクトリーとファイル名については、次の形式が使用されます。
database-name.nodegroup-name.timestamp。
- Windows オペレーティング・システムの場合、%sql1lib%redis% ディレクトリー。サブディレクトリーとファイル名については、次の形式が使用されます。
database-name%first-eight-characters-of-the-nodegroup-name%date%time。

タイム・スタンプ値は、コマンドが発行された時の時刻です。

このユーティリティは、処理の途中で断続的 COMMIT を実行します。

ALTER NODEGROUP ステートメントを使用して、ノードをノード・グループに追加してください。このステートメントは、ノード・グループに対応した表スペース用のコンテナを定義できるようにします。詳細については、SQL 解説書を参照してください。

注: DB2 パラレル・エディション (AIX 版) バージョン 1 構文では、ADD NODE および DROP NODE オプションを指定することは、sysadm 権限または sysctrl 権限のあるユーザーに限りサポートされます。ADD NODE の場合、コンテナはノード・グループ内の既存のノードの、ノード番号が最小のコンテナと同じように作成されます。

再配布を受けた表と従属関係があるすべてのパッケージは無効にされます。再配布ノード・グループ操作が完了した後に、そのようなパッケージを明示的に再バインドすることをお勧めします。明示の再バインドによって、無効パッケ

REDISTRIBUTE NODEGROUP

ージの最初の SQL 要求の実行を初期遅延させなくします。再配布メッセージ・ファイルには、再配布を受けたすべての表のリストが入っています。

再配布ノード・グループ操作が完了した後に、561ページの『RUNSTATS』を発行して統計を更新することもお勧めします。

複写された要約表や DATA CAPTURE CHANGES を用いて定義された表を含むノード・グループは、再配布することができません。

ユーザー一時表スペースと既存の宣言一時表とがノード・グループに存在する場合、再配布は許可されません。

以下も参照

496ページの『REBIND』

REFRESH LDAP

Lightweight Directory Access Protocol (LDAP) の情報が変更された場合に、更新されたその情報に合わせてローカル・マシンのキャッシュを最新表示します。

このコマンドは、Windows NT、Windows 98、Windows 95、AIX、および Solaris でのみ使用可能です。

許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

CLI CFG

CLI 構成を更新するよう指定します。

注: このパラメーターは AIX および Solaris ではサポートされていません。

DB DIR

データベース・ディレクトリーを更新するよう指定します。

NODE DIR

ノード・ディレクトリーを更新するよう指定します。

使用上の注意

最新表示中に LDAP のオブジェクトが除去されると、それに対応するローカル・マシンの LDAP 項目も除去されます。LDAP の情報が変更されると、それに対応する LDAP 項目もそれに応じて変更されます。DB2CLI.INI ファイルを手動で更新する場合は、REFRESH LDAP CLI CFG コマンドを実行して、現行ユーザーのキャッシュを更新してください。

REFRESH LDAP

REFRESH LDAP DB DIR および REFRESH LDAP NODE DIR コマンドは、ローカル・データベースまたはノード・ディレクトリーで検出される LDAP データベースまたはノード項目を削除します。データベースまたはノード項目は、ユーザーがデータベースや LDAP で検出されるインスタンスに接続し、DB2LDAPCACHE が未設定または YES に設定されている場合に、ローカル・データベースまたはノード・ディレクトリーに再度追加されます。

REGISTER

ネットワーク・ディレクトリー・サーバーに DB2 サーバーを登録します。

許可

なし

必須接続

なし

コマンド構文

```

▶ REGISTER [DB2 SERVER IN] [NWBINDERY path] [LDAP path]

```

NWBINDERY path:

```

| NWBINDERY USER username [PASSWORD password]

```

LDAP path:

```

| LDAP AS nodename

```

```

▶ PROTOCOL [TCP/IP] [HOSTNAME hostname] [SVCENAME svcename] [SECURITY SOCKS]
| [NETBIOS] [NNAME nname]
| [APPN path]
| [IPXSPX] [IPX_ADDRESS ipxaddr]
| [NPIPE]

```

```

▶ [REMOTE computer] [INSTANCE instance] [NODETYPE [SERVER] [MPP] [DCS]]

```

```

▶ [OSTYPE ostype] [WITH "comments"] [USER username] [PASSWORD password]

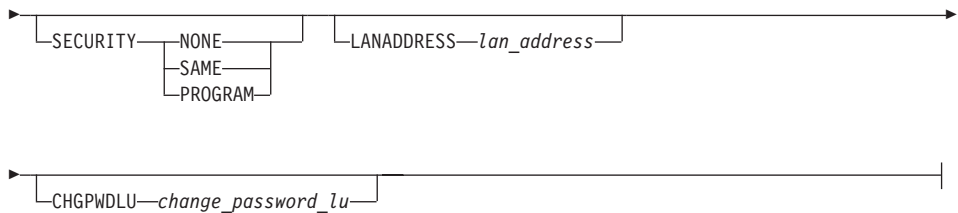
```

APPN path:

```

| APPN NETWORK net_id PARTNERLU partner_lu MODE mode [TPNAME tp_name]

```



コマンド・パラメーター

IN DB2 サーバーを登録するネットワーク・ディレクトリー・サーバーを指定します。有効な値は次のとおりです。 NetWare バインダリーの場合 NWBINDERY、LDAP (Lightweight Directory Access Protocol) ディレクトリー・サーバーの場合 LDAP。

USER username

NWBINDERY の場合、これはネットワーク・サーバーにログインするときに使用するユーザー ID です。ユーザー ID には、SUPERVISOR またはワークグループ管理者と同等のセキュリティーが必要です。ユーザー名は、NetWare ディレクトリー・サーバーに登録する場合には必ず提供しなければなりません。ユーザー名は、NETWARE サーバーにログインするときのユーザー ID となります。LDAP の場合、ユーザーの LDAP 識別名 (DN) です。LDAP ユーザー DN には、LDAP ディレクトリー内でオブジェクトを作成したり更新したりするための権限が必要です。LDAP に登録する場合には、ユーザー名はオプションです。ユーザーの LDAP DN が指定されない場合、現行ログオン・ユーザーの認証が使用されます。

PASSWORD password

アカウント・パスワード。

AS nodename

LDAP 内の DB2 サーバーを表す短い名前を指定します。ノード項目は、このノード名によって LDAP にカタログ化されます。クライアントは、このノード名によってサーバーに接続できます。この LDAP ノード項目に関連付けられるプロトコルは、PROTOCOL パラメーターで指定します。

PROTOCOL

LDAP ノード項目に関連付けるプロトコル・タイプを指定します。データベース・サーバーは複数のプロトコル・タイプをサポートできるため、この値にはクライアント・アプリケーションが実際に使用するプロ

トコル・タイプを指定します。DB2 サーバーはプロトコルごとに 1 つずつ指定しなければなりません。有効な値は、TCP/IP、NETBIOS、APPN、IPXSPX、および NPIPE です。Windows NT の名前付きパイプを使用する場合は、一番あとの値を指定してください。このプロトコル・タイプをサポートするのは、Windows NT オペレーティング・システムで稼働する DB2 サーバーだけです。

注: NETBIOS および NPIPE は、AIX および Solaris オペレーティング・システム上ではサポートされていません。ただし、これらのプロトコルは、Windows NT などのオペレーティング・システムを使用するリモート・サーバーに対しては登録できます。

HOSTNAME hostname

TCP/IP ホスト名 (または IP アドレス) を指定します。

SVCENAME svcename

TCP/IP サービス名またはポート番号を指定します。

SECURITY SOCKS

使用する TCP/IP ソケット・セキュリティーを指定します。

NNAME nname

NetBIOS ワークステーション名を指定します。

NETWORK net_id

APPN ネットワーク ID を指定します。

PARTNERLU partner_lu

DB2 サーバー・マシン用の APPN パートナー LU 名を指定します。

MODE mode

APPN モード名を指定します。

TPNAME tpname

APPN トランザクション・プログラム名を指定します。省略時値は DB2DRDA です。

SECURITY

APPN セキュリティー・レベルを指定します。有効な値は以下のとおりです。

NONE サーバーに送信する割り振り要求に、安全保護情報が含まれないということを指定します。これは、DB2 UDB サーバーの場合の省略時セキュリティーです。

SAME サーバーに送信する割り振り要求に、ユーザー名が含まれないということを指定します。これは、ユーザー名が「すでに検査

済み」という標識で指定されます。サーバーは、「すでに検査済み」という保証を受け入れられるように構成されていなければなりません。

PROGRAM

サーバーに送信する割り振り要求に、ユーザー名とパスワードの両方が含まれるということを指定します。これは、DB2 (MVS 版)、DB2 (AS/400 版)、DB2 (VM 版) などのホスト・データベース・サーバーの場合の省略時セキュリティです。

LANADDRESS lan_address

APPN ネットワーク・アダプター・アドレスを指定します。

CHGPWDLU change_password_lu

ホスト・データベース・サーバーのパスワード変更時に使用される、パートナー LU の名前を指定します。

IPX_ADDRESS ipxaddr

完全な IPX アドレスを指定します。DB2 UDB がインストールされているシステムの IPX アドレスは、**db2ipxad** コマンドを呼び出せば分かります。IPX アドレスは、12 桁のネットワーク・アドレス、8 桁のノード・アドレス、4 桁のソケット番号によって、*<NetworkAddress>.<NodeAddress>.<socket>* の形式で構成されます。

REMOTE computer

DB2 サーバーが常駐するマシンのコンピューター名を指定します。このパラメーターを指定する必要があるのは、リモート DB2 サーバーを LDAP に登録する場合だけです。この値は、サーバー・マシンを LDAP に追加したときに指定した値と同じでなければなりません。Windows NT オペレーティング・システムの場合、これは NT コンピューター名です。UNIX ベースのシステムの場合、これは TCP/IP ホスト名です。OS/2 の場合、これは **DB2SYSTEM** レジストリー変数に指定された値です。

INSTANCE instance

DB2 サーバーのインスタンス名を指定します。リモート・インスタンスの場合 (つまり、REMOTE パラメーターの値が指定されている場合)、インスタンス名は必ず指定しなければなりません。

NODETYPE

データベース・サーバーのノード・タイプを指定します。有効な値は以下のとおりです。

SERVER

DB2 UDB エンタープライズ・エディション・サーバーの場合に、**SERVER** ノード・タイプを指定します。これが省略時値です。

MPP

DB2 UDB エンタープライズ拡張エディション (区分データベース) サーバーの場合に、**MPP** ノード・タイプを指定します。

DCS

ホスト・データベース・サーバーを登録する場合に、**DCS** ノード・タイプを指定します。これを指定すると、クライアントまたはゲートウェイはデータベース・プロトコルとして **DRDA** を使用します。

OSTYPE ostype

サーバー・マシンのオペレーティング・システムのタイプを指定します。有効な値は次のとおりです。OS2、AIX、WIN95、NT、HPUX、SUN、MVS、OS400、VM、VSE、SNI、SCO、SGI、LINUX、および DYNIX/ptx。オペレーティング・システムのタイプが指定されない場合、ローカル・サーバーに対してはローカルのオペレーティング・システムのタイプが使用され、リモート・サーバーに対してはオペレーティング・システムのタイプは使用されません。

WITH "comments"

DB2 サーバーを記述します。ネットワーク・ディレクトリーで登録されるサーバーについての記述を補足する、任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは必ず二重引用符で囲んでください。

使用上の注意

DB2 サーバーを **NWBINDERY** に登録することは、ファイル・サーバー・アドレス指定を使用して、クライアントやサーバーに接続する場合にのみ用いられます。

DB2 サーバーは、サーバーがサポートするプロトコルごとに 1 つずつ登録します。たとえば、DB2 サーバーが **NetBIOS** と **TCP/IP** の両方をサポートする場合、**REGISTER** コマンドは次のように 2 回呼び出す必要があります。

```
db2 register db2 server in ldap as tcpnode protocol tcpip
db2 register db2 server in ldap as nbnode protocol netbios
```

DB2 サーバー・インスタンスごとに **REGISTER** コマンドを 1 回ずつ発行して、サーバーをディレクトリー・サーバーに登録してください。通信パラメー

REGISTER

ター・フィールドを再構成する場合や、サーバー・ネットワーク・アドレスを変更する場合には、ネットワーク・ディレクトリー・サーバーで DB2 サーバーを更新してください。

LDAP の DB2 サーバーを更新するには、変更が実施された後に UPDATE LDAP NODE コマンドを使用します。

NWBINDERY の DB2 サーバーを更新するには、変更が実施される前に DB2 サーバーを一度登録解除し、変更が実施された後に再度登録します。

DB2 サーバーをローカルで登録するときに何らかのプロトコル構成パラメーターを指定すると、データベース・マネージャー構成ファイルに指定された値はオーバーライドされます。

APPN の場合、データベース・マネージャー構成ファイルには TPNAME だけがあります。APPN を正しく登録するためには、必須パラメーター (NETWORK、PARTNERLU、MODE、TPNAME、および SECURITY) に値を指定しなければなりません。オプション・パラメーター (LANADDRESS および CHGPWDLU) に値を指定することもできます。

REGISTER コマンドを、ローカル DB2 インスタンスを LDAP に登録するために使用し、NODETYPE および OSTYPE の 1 つまたは両方を指定する場合、それらはローカル・システムから検索された値で置き換えられます。

REGISTER コマンドを、リモート DB2 インスタンスを LDAP に登録するために使用し、NODETYPE および OSTYPE の 1 つまたは両方を指定しない場合、省略時値の SERVER と Unknown がそれぞれ使用されます。

REGISTER コマンドを使用してリモートの DB2 サーバーを LDAP に登録する場合、リモート・サーバーが使用する通信プロトコルとともに、リモート・サーバーのコンピューター名とインスタンス名も指定する必要があります。

ホスト・データベース・サーバーを登録する場合には、NODETYPE パラメーターに値 DCS を指定しなければなりません。

以下も参照

64ページの『db2ipxad - IPX/SPX インターネットワーク・アドレスの獲得』

245ページの『DEREGISTER』

610ページの『UPDATE LDAP NODE』

REORGANIZE TABLE

断片化されたデータを消去するために行を再作成、および情報を縮小化することによって、表を再編成します。

効力範囲

このコマンドは、ノード・グループ内のすべてのノードに影響を与えます。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- 表に対する CONTROL 特権

必須接続

データベース

コマンド構文

```

▶▶ REORG TABLE table-name [INDEX index-name] [USE tablespace-name]

```

コマンド・パラメーター

TABLE *table-name*

再編成する表を指定します。表は、ローカルまたはリモート・データベースにあるものです。完全修飾の名前または別名の書式は、*schema.table-name* が使用されなければなりません。 *schema* には、表作成時のユーザー名が入ります。

注: 型既定表の場合、指定するテーブル名は階層のルート表の名前でなければなりません。

INDEX *index-name*

表を再編成する際に使用する索引を指定します。完全修飾の名前の書式は、*schema.index-name* が使用されなければなりません。 *schema* は、その索引が作成された時のユーザー名です。データベース・マネージャ

REORGANIZE TABLE

ーは、再編成している表のレコードを物理的にリオーダーする索引を使用します。索引名が指定されない場合には、そのレコードは、順番に関係なく再編成されます。

USE tablespace-name

データベース・マネージャーが再構成される表を一時的に保管できるシステム一時表スペースの名前を指定します。表スペースの名前が入力されなかった場合、データベース・マネージャーは、再編成しようとする表が存在する表スペースにその表の作業コピーを保管します。

8KB、16KB、または 32KB の表オブジェクトの場合、ユーザーによって明示的に指定されるシステム一時表スペースのページ・サイズは、表データ (LONG または LOB 列データを含む) が存在する表スペースのページ・サイズと一致しなければなりません。

例

作業域としてシステム一時表スペース TEMPSPACE1 を使用して EMPLOYEE 表を再編成するには、次のように入力してください。

```
db2 reorg table homer.employee using tempspace1
```

使用上の注意

このコマンドは、宣言一時表に対してはサポートされていません。

何回も修正されてデータが断片化して、アクセス・パフォーマンスが大幅に低下した表も再編成の対象になります。520ページの『REORGCHK』を使用して、表の再編成が必要かどうか判別してください。すべてのデータベース操作が完了し、すべてのロックが解放されていることを確かめてから、REORGANIZE TABLE を呼び出してください。これは、WITH HOLD でオープンされた、すべてのカーソルをクローズした後で COMMIT または ROLLBACK を発行することによって行われます。表の再編成の後で、561ページの『RUNSTATS』を使用して表統計を更新し、496ページの『REBIND』を使用してこの表を使用するパッケージを再バインドします。再編成ユーティリティーは、暗黙的にすべてのカーソルをクローズします。

表がいくつかのノードに区分されている場合、影響を受けるノードのいずれかで再編成が失敗すると、失敗したノードでのみ表の再編成がロールバックされます。

注: 再編成が成功しなかった場合には、一時ファイルを削除すべきではありません。データベース・マネージャーは、これらのファイルを使用し、データベースを回復します。

索引の名前が指定されると、データベース・マネージャーはその索引の順番に従って、データを再編成します。パフォーマンスを最善なものにするため、SQL 照会で頻繁に使用される索引を指定してください。索引の名前が指定されない場合、およびクラスター化索引が存在する場合、データはクラスター化索引に従って順序付けられます。

表の PCTFREE 値は、ページごとの空きスペース量を決定するものとなります。値が設定されていない場合、ユーティリティーはそれぞれのページで可能な限り多くのスペースを満たします。

このユーティリティーは、通称の使用はサポートしていません。

REORGANIZE TABLE は、視点に対しては使用できません。

DMS 表がある表スペースのオンライン・バックアップが実行されている間は、REORGANIZE TABLE を DMS 表に使用できません。

表の再編成の後に表スペースのロールフォワード回復を完了させるには、データと長形式表スペースの両方で、ロールフォワードが使用可能でなければなりません。

その表が、COMPACT オプションを使用しない LOB 列を含む場合、LOB DATA 記憶域オブジェクトは、表の再編成に従いかなり大きくなる可能性があります。これは、行が再編成された順序、および使用される (SMS/DMS) 表スペースのタイプの結果になります。

REORGANIZE TABLE では、索引拡張に基づく索引を使用することはできません。

以下も参照

496ページの『REBIND』

520ページの『REORGCHK』

561ページの『RUNSTATS』

REORGCHK

REORGCHK

データベースでの統計を計算し、表が再編成される必要があるか判別します。

効力範囲

このコマンドは、db2nodes.cfg ファイル中の任意のノードから発行できます。これを使用して、カタログ中の表および索引統計を更新できます。

許可

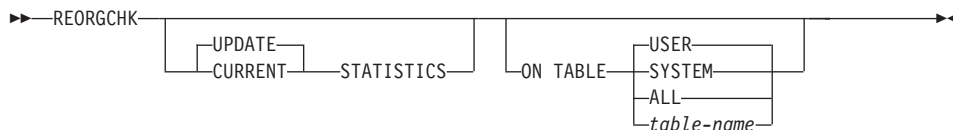
以下のどれかが必要です。

- *sysadm* または *dbadm* の権限
- 表に対する CONTROL 特権

必須接続

データベース

コマンド構文



コマンド・パラメーター

UPDATE STATISTICS

RUNSTATS ルーチン呼び出して、表の統計を更新してから、更新された統計を使用して、表の再編成が必要であるか判別します。

REORGCHK を発行したノードに表区画が存在する場合、このノードで RUNSTATS を実行します。このノードに表区画が存在しない場合、その要求は表の区画を保留しているノード・グループ中の最初のノードに送信されます。次にそのノードで RUNSTATS を実行します。

CURRENT STATISTICS

現在の表の統計を使用して、表の再編成が必要であるか判別します。

ON TABLE

USER 実行時許可 ID が所有する表を検査します。

SYSTEM

システムの表を検査します。

ALL すべてのユーザーおよびシステムの表を検査します。

table-name

検査する表を指定します。完全修飾の名前または別名の書式は、*schema.table-name* が使用されなければなりません。*schema* には、表作成時のユーザー名が入ります。指定された表がシステム・カタログの表である場合には、*schema* は **SYSIBM** です。

注: 型既定表の場合、指定するテーブル名は階層のルート表の名前でなければなりません。

例

以下に示すのは、次のコマンドの出力例です。

```
db2 reorgchk update statistics on table system
```

SAMPLE データベースに対して実行すると、

```
Doing RUNSTATS ....
```

```
Table statistics:
```

```
F1: 100*OVERFLOW/CARD < 5
```

```
F2: 100*TSIZE / ((FPAGES-1) * (TABLEPAGESIZE-76)) > 70
```

```
F3: 100*NPAGES/FPAGES > 80
```

CREATOR	NAME	CARD	OV	NP	FP	TSIZE	F1	F2	F3	REORG
SYSIBM	SYSCHECKS	-	-	-	-	-	-	-	-	---
SYSIBM	SYSCOLAUTH	-	-	-	-	-	-	-	-	---
SYSIBM	SYSCOLCHECKS	-	-	-	-	-	-	-	-	---
SYSIBM	SYSCOLDIST	-	-	-	-	-	-	-	-	---
SYSIBM	SYSCOLUMNS	735	0	25	25	92610	0	95	100	---
SYSIBM	SYSCONSTDEP	-	-	-	-	-	-	-	-	---
SYSIBM	SYSDATATYPES	13	0	1	1	1027	0	-	100	---
SYSIBM	SYSDBAUTH	3	0	1	1	90	0	-	100	---
SYSIBM	SYSEVENTMONITORS	-	-	-	-	-	-	-	-	---
SYSIBM	SYSEVENTS	-	-	-	-	-	-	-	-	---
SYSIBM	SYSFUNCPARMS	254	0	6	6	21590	0	100	100	---
SYSIBM	SYSFUNCTIONS	104	0	8	8	728	0	2	100	*-
SYSIBM	SYSINDEXAUTH	2	0	1	1	112	0	-	100	---
SYSIBM	SYSINDEXES	57	17	3	5	9063	29	56	60	***
SYSIBM	SYSKEYCOLUSE	4	0	1	1	268	0	-	100	---
SYSIBM	SYSPLAN	22	0	2	2	154	0	3	100	*-
SYSIBM	SYSPLANAUTH	41	0	1	1	1804	0	-	100	---
SYSIBM	SYSPLANDEP	-	-	-	-	-	-	-	-	---
SYSIBM	SYSRELS	-	-	-	-	-	-	-	-	---
SYSIBM	SYSSECTION	4	0	1	1	260	0	-	100	---
SYSIBM	SYSSTMT	4	0	1	1	268	0	-	100	---

REORGCHK

SYSIBM	SYSTABAUTH	68	0	2	2	3944	0	98	100	---
SYSIBM	SYSTABCONST	2	0	1	1	132	0	-	100	---
SYSIBM	SYSTABLES	69	0	6	6	483	0	2	100	--*
SYSIBM	SYSTABLESPACES	3	0	1	1	225	0	-	100	---
SYSIBM	SYSTRIGDEP	-	-	-	-	-	-	-	-	----
SYSIBM	SYSTRIGGERS	-	-	-	-	-	-	-	-	----
SYSIBM	SYSVIEWDEP	42	0	1	1	2646	0	-	100	---
SYSIBM	SYSVIEWS	32	0	5	5	3168	0	19	100	--*

Index statistics:

F4: CLUSTERRATIO or normalized CLUSTERFACTOR > 80

F5: $100 * (\text{KEYS} * (\text{ISIZE} + 8) + (\text{CARD} - \text{KEYS}) * 4) / (\text{NLEAF} * \text{INDEXPAGESIZE}) > 50$

F6: $(100 - \text{PCTFREE}) * (\text{INDEXPAGESIZE} - 96) / (\text{ISIZE} + 12) ** (\text{NLEVELS} - 2) * (\text{INDEXPAGESIZE} - 96) / (\text{KEYS} * (\text{ISIZE} + 8) + (\text{CARD} - \text{KEYS}) * 4) < 100$

CREATOR	NAME	CARD	LEAF	LVLS	ISIZE	KEYS	F4	F5	F6	REORG

Table: SYSIBM.SYSCHECKS										
SYSIBM	IBM37	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSCOLAUTH										
SYSIBM	IBM42	-	-	-	-	-	-	-	-	----
SYSIBM	IBM43	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSCOLCHECKS										
SYSIBM	IBM38	-	-	-	-	-	-	-	-	----
SYSIBM	IBM39	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSCOLDIST										
SYSIBM	IBM46	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSCOLUMNS										
SYSIBM	IBM01	735	12	2	33	735	97	64	11	---
SYSIBM	IBM24	735	1	1	20	10	85	-	-	----
Table: SYSIBM.SYSCONSTDEP										
SYSIBM	IBM44	-	-	-	-	-	-	-	-	----
SYSIBM	IBM45	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSDATATYPES										
SYSIBM	IBM40	13	1	1	20	13	100	-	-	----
SYSIBM	IBM41	13	1	1	2	13	100	-	-	----
Table: SYSIBM.SYSDBAUTH										
SYSIBM	IBM12	3	1	1	17	3	100	-	-	----
Table: SYSIBM.SYSEVENTMONITORS										
SYSIBM	IBM47	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSEVENTS										
SYSIBM	IBM48	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSFUNCPARMS										
SYSIBM	IBM31	254	2	2	30	104	100	58	77	---
SYSIBM	IBM32	254	3	2	51	154	96	79	37	---
SYSIBM	IBM33	254	1	1	6	1	100	-	-	----
Table: SYSIBM.SYSFUNCTIONS										
SYSIBM	IBM25	104	1	1	30	104	100	-	-	----
SYSIBM	IBM26	104	1	1	27	104	86	-	-	----
SYSIBM	IBM27	104	1	1	18	50	86	-	-	----
SYSIBM	IBM28	104	1	1	16	2	99	-	-	----
SYSIBM	IBM29	104	1	1	4	104	100	-	-	----
SYSIBM	IBM30	104	2	2	53	104	86	79	56	---
Table: SYSIBM.SYSINDEXAUTH										
SYSIBM	IBM17	2	1	1	47	2	100	-	-	----

SYSIBM	IBM18	2	1	1	30	2	100	-	-	----
Table: SYSIBM.SYSINDEXES										
SYSIBM	IBM02	57	1	1	17	57	100	-	-	----
SYSIBM	IBM03	57	1	1	25	57	100	-	-	----
Table: SYSIBM.SYSKEYCOLUSE										
SYSIBM	IBM35	4	1	1	57	4	100	-	-	----
SYSIBM	IBM36	4	1	1	44	2	100	-	-	----
Table: SYSIBM.SYSPLAN										
SYSIBM	IBM07	22	1	1	16	22	100	-	-	----
SYSIBM	IBM19	22	1	1	8	1	100	-	-	----
Table: SYSIBM.SYSPLANAUTH										
SYSIBM	IBM13	41	1	1	33	41	100	-	-	----
SYSIBM	IBM14	41	1	1	16	22	100	-	-	----
Table: SYSIBM.SYSPLANDEP										
SYSIBM	IBM08	-	-	-	-	-	-	-	-	----
SYSIBM	IBM09	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSRELS										
SYSIBM	IBM20	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSSECTION										
SYSIBM	IBM10	4	1	1	20	4	100	-	-	----
Table: SYSIBM.SYSSTMT										
SYSIBM	IBM11	4	1	1	20	4	100	-	-	----
Table: SYSIBM.SYSTABAUTH										
SYSIBM	IBM15	68	1	1	38	68	100	-	-	----
SYSIBM	IBM16	68	1	1	21	68	100	-	-	----
Table: SYSIBM.SYSTABCONST										
SYSIBM	IBM34	2	1	1	44	2	100	-	-	----
Table: SYSIBM.SYSTABLES										
SYSIBM	IBM00	69	1	1	21	69	95	-	-	----
SYSIBM	IBM21	69	1	1	12	3	100	-	-	----
SYSIBM	IBM22	69	1	1	6	1	100	-	-	----
SYSIBM	IBM23	69	1	1	6	1	100	-	-	----
Table: SYSIBM.SYSTABLESPACES										
SYSIBM	IBM49	3	1	1	14	3	100	-	-	----
SYSIBM	IBM50	3	1	1	8	1	100	-	-	----
Table: SYSIBM.SYSTRIGDEP										
SYSIBM	IBM51	-	-	-	-	-	-	-	-	----
SYSIBM	IBM52	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSTRIGGERS										
SYSIBM	IBM53	-	-	-	-	-	-	-	-	----
SYSIBM	IBM54	-	-	-	-	-	-	-	-	----
Table: SYSIBM.SYSVIEWDEP										
SYSIBM	IBM05	42	1	1	42	42	100	-	-	----
SYSIBM	IBM06	42	1	1	20	32	100	-	-	----
Table: SYSIBM.SYSVIEWS										
SYSIBM	IBM04	32	1	1	20	32	100	-	-	----

CLUSTERRATIO or normalized CLUSTERFACTOR (F4) will indicate REORG is necessary for indexes that are not in the same sequence as the base table. When multiple indexes are defined on a table, one or more indexes may be flagged as needing REORG. Specify the most important index for REORG sequencing.

表統計の用語 (式 1~3) の意味は以下のとおりです。

CARD 基本表の行数。

OV (OVERFLOW) オーバーフローした行数。

NP (NPAGES) データを含むページ数。

FP (FPAGES) ページの合計数。

TSIZE 表サイズ (バイト数)。表 (CARD) 内の行数と行の長さの平均を基にして計算されます。行の長さの平均は、列の長さの平均 (SYSCOLUMNS の AVGCOLLEN) の合計に、行のオーバーヘッドとして 10 バイトを加えたものを基にして計算されます。長フィールドと LOB の場合には、記述子のおおよその長さだけが使用されます。実際の長フィールドまたは LOB データは、TSIZE にはカウントされません。

TABLEPAGESIZE

表データが存在する表スペースのページ・サイズ。

F1 式 1 の結果。

F2 式 2 の結果。

F3 式 3 の結果。

REORG

この列に表示されている各ハイフン (-) は、計算結果が、対応する式の設定範囲内であったことを示しています。各アスタリスク (*) は、計算結果が、その対応する式の設定範囲を超えたことを示しています。

- 列の左側の - または * は F1 (公式 1) に対応しています。
- 列の中央の - または * は F2 (公式 2) に対応しています。
- 列の右側の - または * は F3 (公式 3) に対応しています。

表の再編成は、その計算結果が式によって設定された範囲を超える場合に、提案されます。

たとえば、--- は、F1、F2、および F3 の式の結果がその式の設定範囲内であるために、再編成が提案される表はないことを示しています。表記 ** は、F2 の結果がその設定範囲内であっても、F1 と F3 の結果が表の再編成を提案していることを示しています。表記 *-- は、F1 の式のみが、その範囲を超えていることを示しています。

注: 表名は 30 文字で切り捨てられ、31 列目の ">" 記号が表名の切り捨て位置を表します。

索引統計の用語 (式 4~6) の意味は、次のとおりです。

CARD 基本表の行数。

LEAF 索引リーフ (ページ) の合計数。

LVLS (LEVELS) 索引レベルの数。

ISIZE 索引サイズ、索引に関係するすべての列による列の平均の長さから計算される。

KEYS (FULLKEYCARD) 固有索引項目の数。

INDEXPAGESIZE

表索引が存在する表スペースのページ・サイズ。表の作成時に指定します。指定しなかった場合、INDEXPAGESIZE の値は TABLEPAGESIZE の値と同じになります。

PCTFREE

各索引ページで空きスペースのままにしておくパーセントを指定します。値は索引の定義時に割り当てられます。値の範囲は 0~99 です。省略時値は 10 です。

F4 式 4 の結果。

F5 式 5 の結果。表記法 +++ は、結果が 999 を超えて無効であることを示しています。UPDATE STATISTICS オプションを指定して REORGCHK を戻すか、REORGCHK コマンドに続けて 561ページの『RUNSTATS』を発行してください。

F6 式 6 の結果。表記法 +++ は、結果が 999 を超えて無効であることを示しています。UPDATE STATISTICS オプションを指定して REORGCHK を戻すか、REORGCHK コマンドに続けて 561ページの『RUNSTATS』を発行してください。

REORG

この列に表示されている各ハイフン (-) は、計算結果が、対応する式の設定範囲内であったことを示しています。各アスタリスク (*) は、計算結果が、その対応する式の設定範囲を超えたことを示しています。

- 列の左側の - または * は F4 (式 4) に対応しています。
- 列の中央の - または * は F5 (式 5) に対応しています。
- 列の右側の - または * は F6 (式 6) に対応しています。

表の再編成は、その計算結果が式によって設定された範囲を超える場合に、提案されます。

使用上の注意

このコマンドは、宣言一時表の統計情報は表示しません。

このユーティリティーは、通称の使用はサポートしていません。

REORGCHK は、6 つの異なる公式から得た統計を計算し、表の再編成によってパフォーマンスが低下するか、または改善できるのかを判別します。

考慮事項: ヌルの表 (TSIZE=0) が再編成を必要としている場合には、これらの統計を参考にせず、次の規準を当てはめてください。つまり、TSIZE=0 で FPAGE>0 の場合は、表の再編成が必要です。TSIZE=0 で FPAGE=0 の場合は、再編成は必要ありません。

REORGCHK は、次の式を使用して、行の物理的な場所および表のサイズを分析します。

- 式 F1:

$$100 * \text{OVERFLOW} / \text{CARD} < 5$$

表のオーバーフロー行の合計数は、行の合計数の 5 パーセント以下でなければなりません。オーバーフロー行は、行が更新されて、新しい行のバイト数が古い行 (VARCHAR フィールド) のそれより大きくなる場合、または列が既存の表に追加される場合に作成されます。

- 式 F2:

$$100 * \text{TSIZE} / ((\text{FPAGES} - 1) * (\text{TABLEPAGE} \text{SIZE} - 76)) > 70$$

バイトで表された表のサイズ (TSIZE) は、表に割り当てられた合計スペースの 70 パーセントを超えていなければなりません (空きスペースを 30% 以下の値にします)。表に割り当てられる合計スペースは、表データが存在する表スペースのページ・サイズによって決まります (オーバーヘッド分の 76 バイトを差し引きます)。割り当てられている最終ページは、通常埋められていないため、FPAGES から 1 を引きます。

- 式 F3:

$$100 * \text{NPAGES} / \text{FPAGES} > 80$$

全く行を含まないページ数は、ページ合計数の 20 パーセントより少ない値にします (行が削除された後では、ページはヌルになります)。

REORGCHK は、次の式を使用して、表データに対する索引の関係を分析します。

- 式 F4:

CLUSTERRATIO or normalized CLUSTERFACTOR > 80

クラスター化索引比率は、80 パーセントより大きくします。複数の索引が 1 つの表に定義される場合は、これらの索引のいくつかは、低いクラスター比率を持っています (索引順序は、表の順序と同じではありません)。これを避けることはできません。表を再編成する際に、必ず最も重要な索引を指定してください。そのクラスター比率は、通常、数の多い複写キーおよび数の多い項目を含む索引には最適ではありません。

- 式 F5:

$$100 * (\text{KEYS} * (\text{ISIZE} + 8) + (\text{CARD} - \text{KEYS}) * 4) / (\text{NLEAF} * \text{INDEXPAGESIZE}) > 50$$

索引項目に予約されたスペースの 50% 以下はヌルでなければなりません (NLEAF>1 の場合のみ検査されます)。

- 式 F6:

$$\frac{(100 - \text{PCTFREE}) * (\text{INDEXPAGESIZE} - 96) / (\text{ISIZE} + 12) ** (\text{NLEVELS} - 2) * (\text{INDEXPAGESIZE} - 96) / (\text{KEYS} * (\text{ISIZE} + 8) + (\text{CARD} - \text{KEYS}) * 4)}{100} < 100$$

索引項目の実際の数、NLEVELS-1 索引ツリーが取り扱うことができる項目の数の 90% 以上 (または 100-PCTFREE) でなければなりません (NLEVELS>1 の場合のみ検査されます)。

注: 多数の表で統計を実行すると、表が大きい場合には特に時間がかかります。

以下も参照

517ページの『REORGANIZE TABLE』

561ページの『RUNSTATS』

RESET ADMIN CONFIGURATION

RESET ADMIN CONFIGURATION

DB2 管理サーバーと関係のあるデータベース・マネージャー構成ファイル内のパラメーターをシステム省略時値にリセットします。この値は、常にリモート・クライアントのサーバーになっているノード・タイプによってリセットされます。DB2 管理サーバーは、DB2 サーバーのリモート管理を使用可能にする特別な DB2 インスタンスです。以下のデータベース・マネージャー構成パラメーターがリセットされます。

- AGENT_STACK_SZ
- AUTHENTICATION
- DIAGLEVEL
- DIAGPATH
- DISCOVER
- DISCOVER_COMM
- FILESERVER
- IPX_SOCKET
- NNAME
- OBJECTNAME
- QUERY_HEAP_SZ
- SYSADM_GROUP
- SYSCTRL_GROUP
- SYSMANT_GROUP
- TPNAME
- TRUST_ALLCLNTS
- TRUST_CLNTAUTH

注: インストール・プログラムで設定される SVCENAME パラメーターは、ユーザーが修正するようにお勧めします。管理サーバー・サービス名は、TCP/IP ポート (523) に登録済みの DB2 を使用するよう設定します。

上記のパラメーターの詳細については、292ページの『GET DATABASE MANAGER CONFIGURATION』を参照してください。

効力範囲

このコマンドはデータベース・マネージャー構成ファイルの \$HOME/sql1lib/db2system をリセットします。これは、\$HOME/sql1lib/db2nodes.cfg ファイル内にリストされているノードすべてに影響を与えます。

許可

sysadm

必須接続

なし、またはインスタンス。インスタンスとの接続は、ローカルの DBM 構成操作を実行する場合には必ずしも必要ではありませんが、リモートの DBM 構成操作の場合には必須です。リモート・インスタンス用のデータベース・マネージャーの構成をリセットするには、最初にそのインスタンスに接続することが必要です。

コマンド構文

```

▶▶—RESET ADMIN—CONFIGURATION—▶▶
                   |
                   |—CONFIG—
                   |
                   |—CFG—
  
```

コマンド・パラメーター

なし

使用上の注意

ADMIN 構成パラメーターのリストを表示または印刷するには、269ページの『GET ADMIN CONFIGURATION』を使用してください。

ADMIN パラメーターの値を変更するには、599ページの『UPDATE ADMIN CONFIGURATION』を使用してください。

上記のパラメーターの詳細については、*管理の手引き* を参照してください。

データベース・マネージャー構成ファイルへの変更は、ファイルがメモリーにロードされた後にのみ有効になります。これは **db2start** の実行中に起こりません。

エラーが生じた場合には、データベース・マネージャー構成ファイルは変更されません。

RESET ADMIN CONFIGURATION

データベース・マネージャ構成ファイルは、そのチェックサムが無効であると、リセットすることができません。これは、データベース・マネージャ構成ファイルが、適切なコマンドを使用せずに変更された場合に起こることがあります。これが起きた場合には、データベース・マネージャを再度インストールし、データベース・マネージャ構成ファイルをリセットしなければなりません。

以下も参照

269ページの『GET ADMIN CONFIGURATION』

599ページの『UPDATE ADMIN CONFIGURATION』

RESET DATABASE CONFIGURATION

特定データベースの構成をシステム省略時値にリセットします。

効力範囲

このコマンドは、それが実行されたノードに対してだけ影響を与えます。

許可

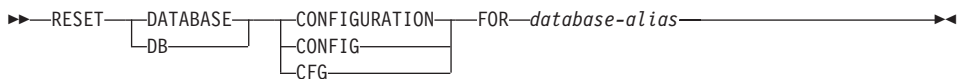
以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

インスタンス。明示的な接続は必要ありません。データベースがリモートとして示されている場合、リモート・ノードへのインスタンス接続はコマンドの持続期間の間、ずっと確立されたままになります。

コマンド構文



コマンド・パラメーター

FOR database-alias

構成がシステム省略時値にリセットされるそのデータベースの別名を指定します。

使用上の注意

データベース構成パラメーターのリストを表示または印刷するには、279ページの『GET DATABASE CONFIGURATION』を使用してください。

構成可能なパラメーターの値を変更するには、606ページの『UPDATE DATABASE CONFIGURATION』を使用してください。

上記のパラメーターの詳細については、*管理の手引き*を参照してください。

RESET DATABASE CONFIGURATION

データベース構成ファイルへの変更は、ファイルがメモリーにロードされた後にのみ有効になります。これを行う前にすべてのアプリケーションはデータベースから切断されている必要があります。

エラーが発生した場合、データベース構成ファイルは変更されません。

チェックサムが無効である場合には、データベースの構成ファイルは、リセットできません。適当なコマンドを使用しないでデータベース構成ファイルを変更するとこれが発生します。これが発生する場合、データベースを復元してデータベース構成ファイルをリセットする必要があります。

以下も参照

279ページの『GET DATABASE CONFIGURATION』

606ページの『UPDATE DATABASE CONFIGURATION』

RESET DATABASE MANAGER CONFIGURATION

データベース・マネージャーの構成ファイルのパラメーターをシステム省略値にリセットします。この値は、常にリモート・クライアントのサーバーになっているノード・タイプによってリセットされます。

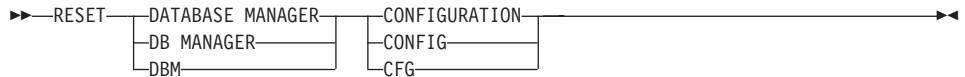
許可

sysadm

必須接続

なし、またはインスタンス。インスタンスとの接続は、ローカルの DBM 構成操作を実行する場合には必ずしも必要ではありませんが、リモートの DBM 構成操作の場合には必須です。リモート・インスタンス用のデータベース・マネージャーの構成をリセットするには、最初にそのインスタンスに接続することが必要です。

コマンド構文



コマンド・パラメーター

なし

使用上の注意

データベース・マネージャー構成パラメーターのリストの表示または印刷を行うには、292ページの『GET DATABASE MANAGER CONFIGURATION』を使用してください。

構成可能なパラメーターの値を変更するには、608ページの『UPDATE DATABASE MANAGER CONFIGURATION』を使用してください。

上記のパラメーターの詳細については、[管理の手引き](#)を参照してください。

データベース・マネージャー構成ファイルへの変更は、ファイルがメモリーにロードされた後にのみ有効になります。サーバー構成パラメーターの場合、これは **db2start** の実行中に起こります。クライアント構成パラメーターの場合、これはアプリケーションが再始動されるときに起こります。クライアントがコマンド行プロセッサである場合は、586ページの『TERMINATE』を呼び出すことが必要です。

RESET DATABASE MANAGER CONFIGURATION

エラーが生じた場合には、データベース・マネージャー構成ファイルは変更されません。

データベース・マネージャー構成ファイルは、そのチェックサムが無効であると、リセットすることができません。これは、データベース・マネージャー構成ファイルが、適切なコマンドを使用せずに変更された場合に起こることがあります。これが起きた場合には、データベース・マネージャーを再度インストールし、データベース・マネージャー構成ファイルをリセットしなければなりません。

以下も参照

292ページの『GET DATABASE MANAGER CONFIGURATION』

608ページの『UPDATE DATABASE MANAGER CONFIGURATION』

RESET MONITOR

指定されたデータベース、またはすべての活動データベースの、内部のデータベース・システム・モニター・データ域をゼロにリセットします。内部のデータベース・システム・モニター・データ域には、データベース用のデータ域のほかに、データベースに接続されるすべてのアプリケーション用のデータ域が含まれます。

許可

以下のどれかが必要です。

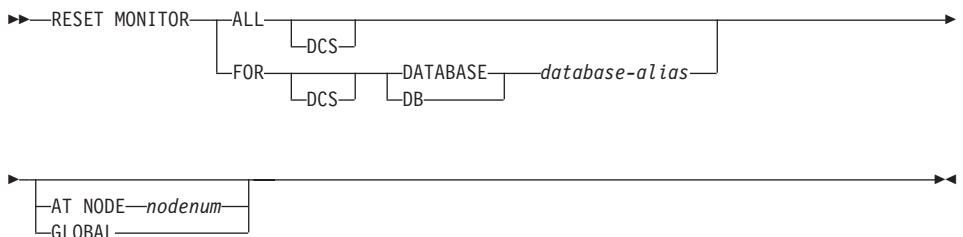
- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

インスタンス。インスタンス接続がない場合、省略時のインスタンス接続が作成されます。

リモート・インスタンス (または異なるローカル・インスタンス) 用のモニター・スイッチをリセットするには、最初にそのインスタンスに接続する必要があります。

コマンド構文



コマンド・パラメーター

ALL このオプションは、すべてのデータベースについて内部カウンターがリセットされることを指示します。

FOR DATABASE database-alias

このオプションは、別名 *database-alias* を持つデータベースの内部カウンターのみがリセットされることを指示します。

RESET MONITOR

DCS このキーワードは、指定された文節に従って、次のいずれかの内部カウンターをリセットします。

- すべての DCS データベース
- 特定の DCS データベース

AT NODE nodenum

モニター・スイッチの状況を表示するノードを指定します。

GLOBAL

区分データベース・システム内のすべてのノードの集合結果を戻します。結果を集める方法については、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。

使用上の注意

各プロセス (接続) は、モニター・データの私用視点を持っています。あるユーザーがリセット、またはモニター・スイッチをオフにしても、その他のユーザーは影響されません。モニター・スイッチ構成パラメーターの設定を、モニター・スイッチの一括変更で変更してください (608ページの『UPDATE DATABASE MANAGER CONFIGURATION』 を参照)。

ALL が指定されると、あるデータベース・マネージャー情報が、戻されるデータの一貫性を維持するためにリセットされ、あるノード・レベルのカウンターがリセットされます。

リセットできるデータ項目のリストは、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。

以下も参照

312ページの『GET SNAPSHOT』

310ページの『GET MONITOR SWITCHES』

RESTART DATABASE

異常終了し、矛盾した状態のままであるデータベースを再始動します。ユーザーが CONNECT 特権を持っている場合には、RESTART DATABASE の正常終了であったアプリケーションはデータベースに接続されたままとなります。

効力範囲

このコマンドは、それが実行されたノードに対してだけ影響を与えます。

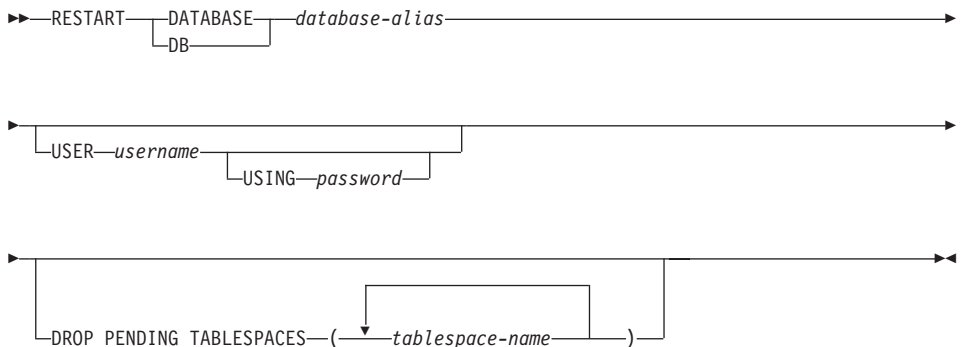
許可

なし

必須接続

このコマンドは、データベース接続を確立します。

コマンド構文



コマンド・パラメーター

DATABASE database-alias

再始動するデータベースを識別します。

USER username

データベースが再始動される際のユーザー名を識別します。

USING password

username を認証するのに使用されるパスワード。パスワードを省略すると、ユーザーに入力を求めるプロンプトが出ます。

RESTART DATABASE

DROP PENDING TABLESPACES tablespace-name

表スペース・コンテナに問題が発生しても、データベース再始動操作を正常に完了するよう指定します。

指定した表スペース用のコンテナで再始動処理中に問題が発生した場合、それに対応する表スペースは、再始動操作後に使用できなくなります (除去保留状態になります)。循環ログの場合には、問題の発生している表スペースが再始動障害の原因となります。問題の発生している表スペース名のリストは、データベースの再始動操作が失敗した原因がコンテナの問題である場合には、db2diag.log に含まれています。データベースにシステム一時表スペースが 1 つしかなく、しかもそのシステム一時表スペースが除去保留状態である場合は、データベースの再始動操作が正常に完了したらすぐに新しい一時表スペースを作成する必要があります。

使用上の注意

データベースへの接続を試行すると、データベースを再始動する必要があることを示すエラー・メッセージが戻される場合、このコマンドを実行してください。このアクションは、このデータベースとの前のセッションが、異常に (たとえば、電源障害により) 終了した場合にのみ起こります。

RESTART DATABASE の完了時に、ユーザーが CONNECT 特権を持っている場合には、データベースへの共用接続は維持され、疑わしいトランザクションが存在する場合には、SQL の警告が発行されます。この場合、データベースはまだ使用可能ですが、未確定トランザクションが、データベースへの最終接続をドロップする前に解決されない場合には、別の RESTART DATABASE を発行してから、再度データベースを使用しなければなりません。386ページの『LIST INDOUBT TRANSACTIONS』を使用して、未確定トランザクションのリストを生成してください。未確定トランザクションの詳細は、管理の手引きを参照してください。

データベースが MPP システム中の単一ノードでのみ再始動される場合、データベースを再始動する必要があることを示すメッセージが、後続のデータベース照会で戻される場合があります。これが起こるのは、照会が依存しているノード上のデータベース区画が再始動された場合です。すべてのノードでデータベースを再始動すれば、問題を解決できます。

以下も参照

SQL 解説書の CONNECT TO ステートメントを参照してください。

RESTORE DATABASE

BACKUP DATABASE を使用してバックアップされた損傷のある、または破壊されたデータベースを再作成します。復元されたデータベースは、バックアップ・コピーが行われた時と同じ状態になります。このユーティリティーは、新規のデータベースに復元できるほかに、バックアップ・イメージのデータベース名と異なる名前のデータベースにも復元することができます。

このユーティリティーは、DB2 の前の 2 つのバージョンによって生成されたバックアップ・イメージを復元するためにも使用できます。移行が必要な場合、これは復元の終了時に自動的に発行されます。

バックアップ操作のときに、データベースがすべてのロールフォワード回復に対して使用可能である場合、RESTORE DATABASE が正常に実行された後に、ROLLFORWARD DATABASE コマンドを出すことによって、データベースを損傷または破壊が起きる前の状態に戻すことができます。

このユーティリティーは、表スペース・レベルのバックアップから復元することもできます。

異なるワークステーション・プラットフォームにバックアップしたデータベースを復元するには、77ページの『db2move - データベース移動ツール』を使用します。

効力範囲

このコマンドは、それが実行されたノードに対してだけ影響を与えます。

許可

既存のデータベースに復元するには、以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

新規データベースに復元するには、以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

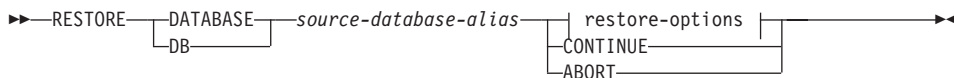
データベース (既存のデータベースに復元する場合)。

RESTORE DATABASE

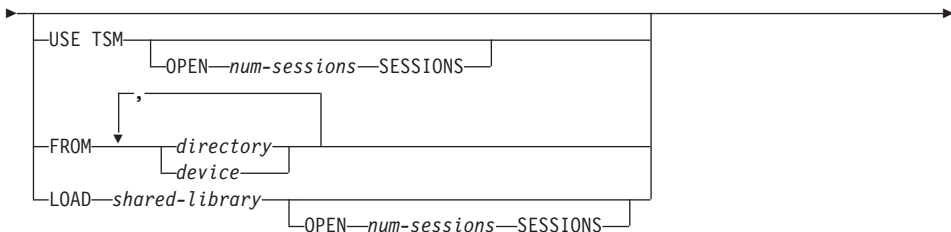
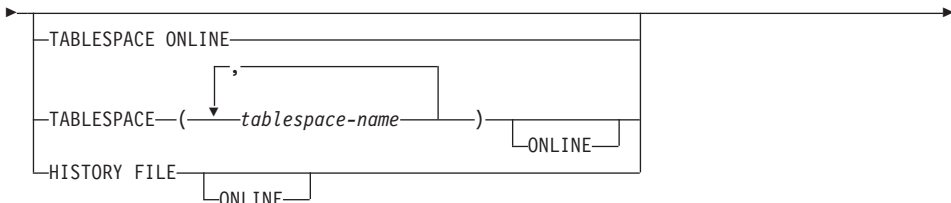
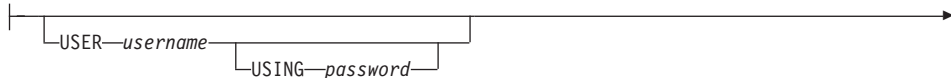
インスタンスおよびデータベース (新規データベースに復元する場合)。インスタンスの接続は、データベースを作成するのに必要です。

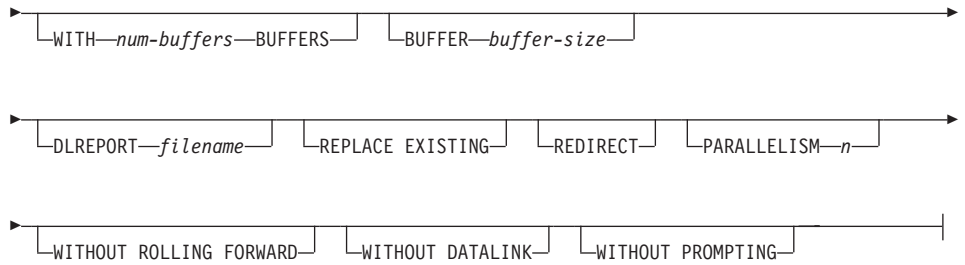
新規のリモート・データベースに復元するには、まず新規データベースを置くインスタンスに接続する必要があります。

コマンド構文



restore-options:





コマンド・パラメーター

DATABASE source-database-alias

バックアップが取得されるソース・データベースの別名です。

CONTINUE

コンテナが再定義されていること、およびリダイレクトした復元の最終ステップを実行する必要があることを指示します。

ABORT

リダイレクトした復元を停止します。これは、1 つまたは複数のステップを繰り返す必要があるエラーが発生したときに便利です。 **ABORT** オプションを指定して **RESTORE DATABASE** を発行した後、**REDIRECT** オプションを指定した **RESTORE DATABASE** を含む、リダイレクトした復元の各ステップを繰り返す必要があります。

USER username

データベースが復元される際のユーザー名を識別します。

USING password

ユーザー名を承認するために使用するパスワード。パスワードを省略すると、ユーザーに入力を求めるプロンプトが出ます。

TABLESPACE tablespace-name

復元される表スペースを指定するときに使用する名前のリストです。

ONLINE

このキーワードは、表スペース・レベル復元を行う場合のみ適用でき、オンラインでバックアップが復元できることを指定します。これは、他のエージェントが、バックアップの復元中にデータベースに接続できることや、表スペースのリストの復元中に他の表スペースのデータを使用できることを意味します。

RESTORE DATABASE

HISTORY FILE

このキーワードは、バックアップから活動記録ファイルを復元するのに、指定されます。

USE TSM

データベースが TSM 管理の出力から復元されることを指示します。

OPEN num-sessions SESSIONS

TSM またはベンダー製品とともに使用する入出力セッションの数です。

FROM directory/device

バックアップ・イメージがあるディレクトリーまたは装置。USE TSM、FROM、および LOAD が省略される場合には、省略時値は現行ディレクトリーです。

注: OS/2 または Windows オペレーティング・システムでは、DB2 が生成するディレクトリーを指定してはなりません。たとえば、次のようなコマンドを実行するとします。

```
db2 backup database sample to c:%backup
db2 restore database sample from c:%backup
```

DB2 は c:%backup ディレクトリーにサブディレクトリーを生成しますが、これらは無視されます。復元するバックアップ・イメージを正確に指定するためには、TAKEN AT パラメーターを使用します。複数のバックアップ・イメージを同じパスに保管することもできます。

複数の項目が指定され、項目の最後がテープ装置である場合には、他のテープが要求されます。有効な応答オプションは、次のとおりです。

- c** 続行。警告メッセージを生成した装置の使用を続けます (たとえば、新しいテープをマウントしたときなど)。
- d** 装置の終了。警告メッセージの原因となった装置の使用だけを停止します (たとえば、これ以上テープがない場合など)。
- t** 終了。復元またはバックアップ・ユーティリティーを打ち切ります。

テープは OS/2 ではサポートされていません。OS/2 の場合、0 または 0: を指定すると、ユーザー出口プログラムが呼び出されて、復元操作が起きます (管理の手引きを参照)。このオプションは、他のプラットフォームでは無効です。

注: ユーザー出口プログラムが復元を実行するために使用されている場合は、リダイレクトした復元は認められていません。

LOAD shared-library

使用するバックアップおよび復元 I/O 関数を含む共用ライブラリー (OS/2 または Windows オペレーティング・システムでは DLL) の名前。フルパスで指定することができます。フルパスを指定しない場合、省略時でユーザー出口プログラムの存在するパスになります。

TAKEN AT date-time

データベース・バックアップのタイム・スタンプ。バックアップ・イメージ・ファイル名には、タイム・スタンプが組み込まれます。

TO target-directory

宛先データベースのディレクトリー。ユーティリティーが存在するデータベースへ復元している場合には、このパラメーターは無視されます。

注: Windows オペレーティング・システムおよび OS/2 上では、このパラメーターを使用する場合はドライブ文字だけを指定してください。さらに長いパスを指定すると、エラーが戻されます。

INTO target-database-alias

宛先データベースの別名。宛先データベースが存在しない場合には、作成されます。

NEWLOGPATH directory

アクティブ・ログ・ファイルの復元後にデータベースが使用する完全修飾ディレクトリー。このパラメーターの機能は、データベース構成パラメーター newlogpath と同じです。ただし、newlogpath の影響は、それが指定された RESTORE コマンドに限定されます。このパラメーターは、バックアップ・イメージのログ・パスが、復元後の使用に適していない場合に使用することができます。たとえば、パスが無効になっていたり、別のデータベースで現在使用中の場合などが考えられます。

WITH num-buffers BUFFERS

使用するバッファの数です。省略時値は 2 です。ただし、複数のソースが読み取られる場合や、並列性パラメーターを増やす場合は、パフォーマンスを向上させるために多数のバッファを使用することができます。

BUFFER buffer-size

復元操作に使用するバッファのサイズ (ページ数)。このパラメーターの最小値は 8 ページです。省略時値は 1024 ページです。バッファ

RESTORE DATABASE

ー・サイズがゼロに指定されている場合、データベース・マネージャー構成パラメーターの値 `restbufsz` が、バッファー割り振りサイズとして使用されます。

復元バッファー・サイズは、バックアップ操作中に指定したバックアップ・バッファー・サイズに正の整数を乗算したサイズでなければなりません。誤ったバッファー・サイズを指定すると、割り振られるバッファーは、許容可能な最小のサイズになります。

テープ装置を使用するには、SCO UnixWare 7 上の DB2 ユーザーは、バッファー・サイズを 16 に指定する必要があります。

DLREPORT filename

ファイル名を指定する場合は、完全修飾にしなければなりません。復元中に (高速調整の結果として) リンク解除されるファイルが報告されません。このオプションが使用されるのは、復元する表に `DATALINK` 列タイプとデータ・リンク・ファイルが含まれている場合だけです。

REPLACE EXISTING

宛先データベースの別名と同じ別名を持つデータベースがすでに存在している場合、このパラメーターは復元ユーティリティに既存のデータベースを復元したデータベースに置換するよう指示します。これは `RESTORE DATABASE` コマンドを含むスクリプトで便利です。 `CLP` は、ユーザーにプロンプトを出さずに、既存のデータベースの削除を検査するからです。 `WITHOUT PROMPTING` パラメーターが指定された場合、 `REPLACE EXISTING` を指定する必要はありませんが、ユーザー介入を標準的に必要とする事象が起こった場合、このコマンドは失敗します。

REDIRECT

リダイレクトした復元を指定します。リダイレクトした復元を完了するには、このコマンドの後に 1 つまたは複数の `SET TABLESPACE CONTAINERS` コマンドを続け、次に `CONTINUE` オプションを指定して `RESTORE DATABASE` コマンドを続ける必要があります。

注: 同一のリダイレクトした復元に関連したコマンドはすべて、同じウィンドウまたは `CLP` セッションから実行しなければなりません。

WITHOUT ROLLING FORWARD

正常に復元された後で、データベースがロールフォワード保留状態にならないように指定します。

正常な復元に続いて、データベースがロールフォワード保留状態にある場合には、データベースが使用できるようになる前に、
ROLLFORWARD DATABASE を実行しなければなりません。

WITHOUT DATALINK

DATALINK 列を持つ任意の表が DataLink_Reconcile_Pending (DRP) 状態に置かれることと、リンクされたファイルの調整が実行されないことを指定します。

PARALLELISM n

復元処理中に作成されたバッファ・マネージャの数と、マニピュレータの数を指定します。省略時は 1 です。

WITHOUT PROMPTING

復元を自動で実行し、ユーザーの介入を通常必要とするアクションでは代わりにエラー・メッセージを戻すことを指定します。

注: テープやディスクなどの取り外し可能媒体装置を使用している場合、このオプションを指定していても、その装置の終わりにはプロンプトが出されます。

例

以下は、別名が MYDB であるデータベースの典型的なリダイレクトした復元のシナリオです。

1. 次のように、REDIRECT オプションを指定して RESTORE DATABASE コマンドを発行する。

```
db2 restore db mydb replace existing redirect
```

ステップ 1 が正常終了した後でステップ 3 が完了する前に、次を発行して復元を打ち切ることができる。

```
db2 restore db mydb abort
```

2. 再定義する必要があるコンテナを持つ表スペースごとに、SET TABLESPACE CONTAINERS コマンドを発行する。たとえば、OS/2 では次のようにします。

```
db2 set tablespace containers for 5 using  
(file 'f:¥ts3con1' 20000, file 'f:¥ts3con2' 20000)
```

復元したデータベースのコンテナが、このステップで指定したものであることを検査するために、LIST TABLESPACE CONTAINERS コマンドを発行する。

3. ステップ 1 および 2 が正常終了した後、次を発行する。

RESTORE DATABASE

```
db2 restore db mydb continue
```

これはリダイレクトした復元の最終ステップです。

4. ステップ 3 が失敗した場合、または復元を打ち切った場合、リダイレクトした復元はステップ 1 から再始動できる。

使用上の注意

データベースおよび表スペースの両方の復元で、復元ユーティリティーは、複数の表スペースから同時に読み取り、複数の場所にある装置に同時に書き込むことができます。

データベース・レベルの復元

既存のデータベースに復元する場合、現行データベースの構成ファイルは、その構成ファイルが破壊されない限り、バックアップ・コピーで置換されません。

WITHOUT ROLLING FORWARD が指定されず、データベースがバックアップされた時点で、ロールフォワード回復が使用可能であった場合には、データベースは正常に復元された後で、ロールフォワード保留状態になります。データベースの状態を検査するには、279ページの『GET DATABASE CONFIGURATION』を使用してください。データベースがロールフォワード保留状態にある場合には、使用できるようになる前に、データベースに対して550ページの『ROLLFORWARD DATABASE』を発行しなければなりません。

BACKUP および RESTORE は、別のファイル・システムまたはノードに、データベースをコピーするのに使用することもできます。

復元されたバックアップ・ファイルが、オンライン・バックアップ中に作成された場合には、順方向回復が、必ず復元の完了時に呼び出されます。順方向回復 (ROLLFORWARD DATABASE を使用する) により、バックアップ操作の途中で発生したすべての変更がキャプチャーされ、データベースを安定状態にすることが確認されます。

オフラインの復元の場合、このユーティリティーは、排他モードでデータベースに接続します。呼び出しアプリケーションを含む、どんなアプリケーションでも、復元されるデータベースにすでに接続していると、そのユーティリティーは失敗します。

復元中に割り込みが発生すると、正常な復元が完了するまで、データベースに正常に接続することが不可能になります。

バックアップ・イメージは、`BACKUP DATABASE` コマンドによって作成されたイメージでなければならず、ディスク、ディスケット (OS/2 または Windows オペレーティング・システムの場合)、テープ、TSM ユーティリティ、またはベンダーの他の製品管理媒体に存在します。テープは OS/2 ではサポートされていません。

データベース・バックアップが既存のデータベースに復元される際、データベースは既存データベースの別名とデータベース名を継承します。存在しないデータベースに復元される場合は、新規データベースが、`target-database-alias` パラメーターで指定された別名とデータベース名を用いて作成されます。宛先データベースの別名が指定されないと、そのデータベースは、バックアップ・データベースの別名とデータベース名を継承します。

リモート・クライアントが復元を開始しても構いませんが、ソースおよびターゲットは、常にサーバーに存在するエンティティを参照します。

データベースの復元には、この解説書で扱う範囲を超える、前提条件の要件と制約がある可能性があります。これらの条件の詳細については、管理の手引きを参照してください。

表スペース・レベルの復元

復元した表スペースが、他のデータベースと同期化されていることを確認するには、ログの最後まで (または表スペースを最後に使用したところまで) 表スペースの下方ロールを行わなければなりません。このため、表スペース・レベルのバックアップおよび復元が行えるのは、ロールフォワード回復が可能な場合のみです。表スペース・レベル・バックアップを実行した後に、ロールフォワード回復が行えない場合、バックアップから復元することができないため、早めに現時点まで表スペースの下方ロールを行ってください。この場合、その時点より前に取った表スペース・レベルのバックアップはすべて復元できなくなります。そのようなバックアップを復元しようとする、復元操作は失敗します。バックアップが無効であるかどうかを判別できない場合 (たとえば、データベースが復元およびロールフォワードされており、それによって新しいログ順序列が作成されている場合)、復元は成功することがあり、ロールフォワード回復処理中に、壊れた復元セットが検出されることとなります。

表のそれぞれの構成要素は、同じ表の他の構成要素からは独立させて、常駐している表スペースでバックアップを取り、復元させることが可能です。

表スペース・レベルのバックアップと復元は、同時に実行することはできません。

RESTORE DATABASE

バックアップを実行した後に名前変更した表スペースを復元する場合、復元コマンドでは、変更後の新しい表スペース名を使用する必要があります。以前の表スペース名を使用しても、それは見つかりません。

DB2 データ・リンク・マネージャーに関する考慮事項

バックアップ・イメージと異なるデータベース名または別名への復元を行うとき、DATALINK 列のある表は DRNP 状態に置かれます。

WITHOUT DATALINK オプションを指定せず、かつ DATALINK データを含む DB2 ファイル・マネージャーが使用できなくなった場合、復元操作は失敗します。このオプションを指定していた場合で、DATALINK データを含む DB2 ファイル・マネージャーが使用できなくなったときは、使用できなくなったサーバーで DATALINK 値を持っていた表を含むすべての表スペースが、復元保留状態になります。

DB2 データ・リンク・マネージャーとデータベース回復の詳細については、[管理の手引き](#) を参照してください。

以下も参照

77ページの『db2move - データベース移動ツール』

157ページの『BACKUP DATABASE』

279ページの『GET DATABASE CONFIGURATION』

455ページの『MIGRATE DATABASE』

550ページの『ROLLFORWARD DATABASE』

REWIND TAPE

DB2 (Windows NT 版) は、ストリーム・テープ装置へのバックアップおよび復元をサポートしています。このコマンドを使用してテープを巻き戻します。

このコマンドは Windows NT でのみ使用可能です。

許可

なし

必須接続

なし

コマンド構文

▶▶—REWIND TAPE—ON—device—▶▶

コマンド・パラメーター

ON device

有効なテープ装置名を指定します。省略時値は ¥¥.¥TAPE0 です。

以下も参照

359ページの『INITIALIZE TAPE』

575ページの『SET TAPE POSITION』

ROLLFORWARD DATABASE

データベースのログ・ファイルに記録されたトランザクションを適用することによって、データベースを回復させます。データベースまたは表スペースのバックアップが復元された後、あるいはすべての表スペースが媒体エラーのためにデータベースによってオフラインにされた場合に呼び出されます。データベースをロールフォワード回復で回復できるのは、それが回復可能な状態にある(すなわち、*logretain* または *userexit*、あるいはその両方のデータベース構成パラメーターがオンに設定されている) ときだけです。

効力範囲

マルチノード環境では、このコマンドはカタログ・ノードからしか発行できません。時間を指定するデータベースまたは表スペースのロールフォワード・コマンドは、*db2nodes.cfg* ファイルにリストされているすべてのノードに影響を与えます。ログの終わりを指定するデータベースまたは表スペースのロールフォワード・コマンドは、指定されたノードに影響を与えます。ノードが指定されない場合、コマンドは、*db2nodes.cfg* ファイルにリストされているすべてのノードに影響を与えます。特定のノードでロールフォワードが必要とされない場合には、そのノードは無視されます。

許可

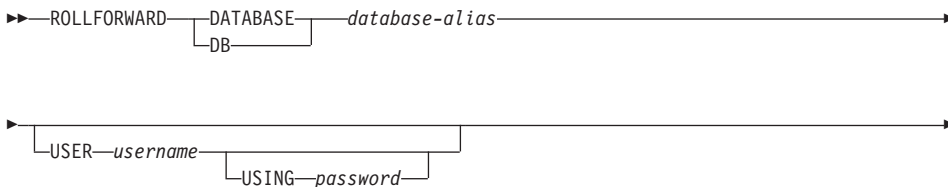
以下のどれかが必要です。

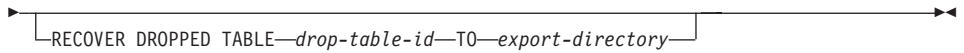
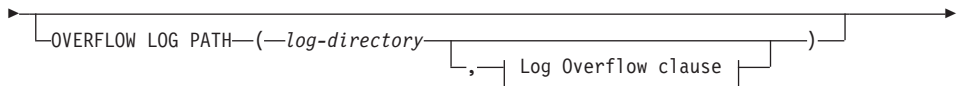
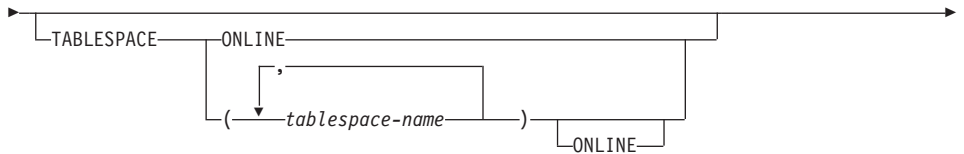
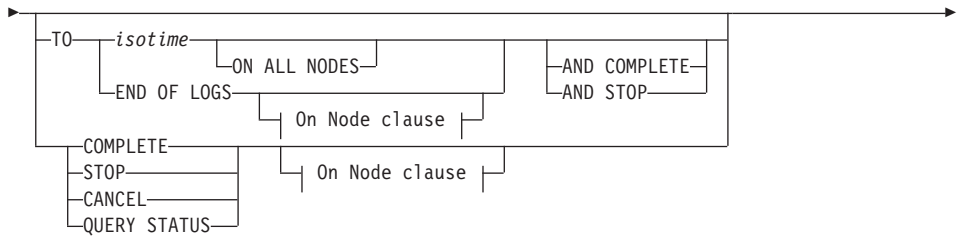
- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

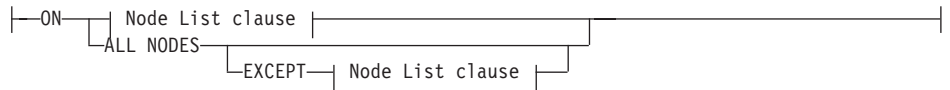
なし。このコマンドは、データベース接続を確立します。

コマンド構文

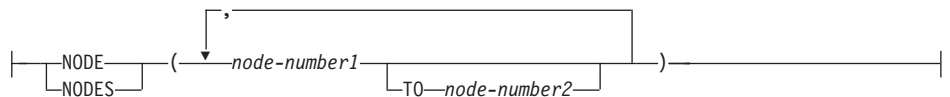




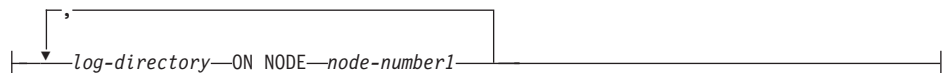
On Node clause:



Node List clause:



Log Overflow clause:



ROLLFORWARD DATABASE

コマンド・パラメーター

DATABASE database-alias

ロールフォワードするデータベースの別名。

USER username

データベースがロールフォワードされる際のユーザー名を識別します。

USING password

ユーザー名を承認するために使用するパスワード。パスワードを省略すると、ユーザーに入力を求めるプロンプトが出ます。

TO

isotime

コミットされたすべてのトランザクションがロールフォワードされる時点 (その時点の前にコミットされたすべてのトランザクションのほかに、ちょうどその時点でコミットされたトランザクションを含む)。

この値は、バインドされた日時を識別する 7 つの部分の文字ストリングからなる、タイム・スタンプとして指定されます。書式は、`yyyy-mm-dd-hh.mm.ss.nnnnnn` (年、月、日、時、分、秒、マイクロ秒) の協定世界時 (CUT) です。

環境変数 **TZ** は、CUT とローカル時間との差を表示します。たとえば、**TZ** 値が `EST5EDT` の場合、ローカル時間帯が `EST` (東部標準時) であり、この時間帯と CUT との差が 5 時間で、夏時間の採用があることを示しています。夏時間採用時には、5 時間の差が 4 時間となり、`CUT = 現在時 + 4` となります。

区分データベース環境では、`ON ALL NODES` が指定されると、ロールフォワード回復がすべてのノードに対して行われます。

END OF LOGS

データベースの構成パラメーターである `logpath` にリストされる、すべてのオンライン・アーカイブ・ログ・ファイルからコミットされた全トランザクションが、適用されることを指定します。

ALL NODES

`db2nodes.cfg` ファイルで指定されている、すべてのノードについてトランザクションがロールフォワードされることを指定します。ノード文節が指定されていない場合、これが省略時値です。

EXCEPT

ノード・リストに指定されているノードを除き、 `db2nodes.cfg` ファイルで指定されている、すべてのノードについてトランザクションがロールフォワードされることを指定します。

ON NODE / ON NODES

一連のノードについてデータベースをロールフォワードします。

node-number1

ノード・リスト内のノード番号を指定します。

node-number2

2 番目のノード番号を指定します。この番号を指定すると、 `node-number1` から `node-number2` までのすべてのノードがノード・リストに含まれます。

COMPLETE / STOP

ログ・レコードのロールフォワードを停止し、未完了のトランザクションをロールバックし、データベースのロールフォワード保留状態をオフにすることによって、ロールフォワード回復処理を完了します。これにより、すでにロールフォワードされたデータベースまたは表スペースへのアクセスが認められます。これらのキーワードは同等です。どちらか一方を指定し、両方は指定しないでください。キーワード **AND** を使用すると、一度に複数の操作を指定することができます (たとえば、 `db2 rollforward db sample to end of logs and complete`)。

注: 表スペースをある時点までロールフォワードすると、表スペースはバックアップ保留状態に置かれます。

CANCEL

ロールフォワード回復処理を取り消します。これにより、順方向回復が開始されている、すべてのノードのデータベースまたは表スペースが回復保留状態で残ります。データベースのロールフォワードが処理中ではない (つまり、データベースがロールフォワード保留状態である) 場合、このオプションは、データベースを回復保留状態に変更します。表スペースのロールフォワードが処理中ではない (つまり、表スペースがロールフォワード保留状態である) 場合は、表スペース・リストを指定しなければなりません。リスト内のすべての表スペースが、回復保留状態に変更されます。表スペースのロールフォワード処理中である (つまり、少なくとも 1 つの表スペースがロールフォワード進行状態にある) 場合は、ロールフォワード進行状態にあるすべての表スペースが回復保留状態に変更されます。表スペース・リストを指定する場合、そのリストには、ロールフォワード進行状態にある表スペースがすべて含まれて

いなければなりません。ある時点までロールフォワードする場合、渡される表スペースはすべて無視され、ロールフォワード進行状態にある表スペースがすべて回復保留状態に変更されます。表スペース・リストを指定してログの終わりまでロールフォワードする場合は、リスト内の表スペースのみが回復保留状態に変更されます。

注: このオプションは十分注意して使用してください。

QUERY STATUS

データベース・マネージャーがロールフォワードしたログ・ファイル、次に必要とされるアーカイブ・ファイル、およびロールフォワード処理が開始されてから、最後にコミットされたトランザクションのタイム・スタンプ (CUT 形式) をリストします。マルチノード環境では、この状況情報は各ノードに関して戻されます。返される情報には、次のフィールドが含まれています。

ノード番号

ロールフォワード状況

データベースまたは表スペースのロールフォワード処理保留、ロールフォワード処理中、ロールフォワード処理の停止、またはロールフォワード処理保留なしの状態です。

読み込む予定の次のログ・ファイル

次に必要なログ・ファイルの名前から成るストリング。マルチノード環境で、ロールフォワード・ユーティリティーに障害が起こり、ログ・ファイルの欠落を示す戻りコードが戻されたとき、またはログ情報の不一致が生じたとき、この情報を使用します。

処理済みログ・ファイル

これ以上回復に必要なく、そのディレクトリーから除去できる、処理済みのログ・ファイルの名前から成るストリング。

コミットされた最終トランザクション

ISO 書式のタイム・スタンプから成る文字列 (yyyy-mm-dd-hh.mm.ss)。このタイム・スタンプは、ロールフォワード回復処理の完了後に、コミットされた最終トランザクションを示しています。タイム・スタンプはデータベースに適用されます。表スペースのロールフォワードでは、データベースにコミットされた最終トランザクションのタイム・スタンプを表します。

注: TO、STOP、COMPLETE、および CANCEL 句が省略されると、
QUERY STATUS が省略時値になります。

TO 句、STOP 句、または COMPLETE 句のいずれかが指定された場合は、コマンドが正常に実行されれば、この情報が表示されます。

TABLESPACE

このキーワードは、表スペース・レベルのロールフォワードを行う場合に使用します。

tablespace-name

ある時点までの表スペース・レベルのロールフォワードを行う場合は必須です。表スペースのサブセットに対してログの終わりまでのロールフォワード処理を指定できます。マルチノード環境では、リストの各表スペースがロールフォワードされている各ノードに存在している必要はありません。もしノード内に表スペースが存在している場合、その表スペースは適正な状態にあることを示します。

ONLINE

このキーワードは、表スペース・レベルのロールフォワード回復をオンラインで行うことができることを指定します。これは、ロールフォワード回復処理の進行中に、他のエージェントが接続できることを意味します。

OVERFLOW LOG PATH log-directory

回復処理中にアーカイブされたログを探索する代替のログ・パスを指定します。マルチノード環境では、これはすべてのノードの省略時オーバーフロー・ログ・パスです。

単一ノード環境では、相対オーバーフロー・ログ・パスの指定もできますが、マルチノード環境では、パスは完全に修飾されていなければなりません。

log-directory ON NODE

マルチノード環境では、特定のノードのために異なるログ・パスで省略時オーバーフロー・ログ・パスをオーバーライドできます。

RECOVER DROPPED TABLE drop-table-id

除去された表をロールフォワード操作中に回復します。表 ID を取得するには、382ページの『LIST HISTORY』を使用します。

ROLLFORWARD DATABASE

TO export-directory

表データが含まれているファイルを書き込むディレクトリーを指定します。指定するディレクトリーは、すべてのノードにアクセスできるものでなければなりません。

例

例 1

ROLLFORWARD コマンドでは、それぞれをキーワード `and` で区切ることによって、一度に複数の操作を指定することができます。たとえば、ログの終わりまでロールフォワードし、完了する場合、コマンドを別々に指定すると、次のようになります。

```
db2 rollforward db sample to end of logs
db2 rollforward db sample complete
```

これらは次のように結合することができます。

```
db2 rollforward db sample to end of logs and complete
```

例 2

ログの終わりまでロールフォワードします (2 つの表スペースが復元されています)。

```
db2 rollforward db sample to end of logs
```

注: ログの終わりまで表スペースをロールフォワードする場合、`AND STOP` または `AND COMPLETE` を使用する必要はありません。表スペース名は必須ではありません。指定しない場合には、ロールフォワード回復を必要としているすべての表スペースが組み込まれます。これらの表スペースの一部のみをロールフォワードする場合は、それらの名前を指定しなければなりません。

例 3

3 つの表スペースが復元された後、1 つをログの終わりまでロールフォワードし、他の 2 つをある時点までロールフォワードします (両方ともオンラインで行われます)。

```
db2 rollforward db sample to end of logs tablespace(TBS1) online
db2 rollforward db sample to 1998-04-03-14.21.56.245378 and stop
tablespace(TBS2, TBS3) online
```

例 4

データベースを復元した後、OVERFLOW LOG PATH でユーザー出口がアーカイブ・ログを保管するディレクトリーを指定して、ある時点までロールフォワードします。

```
db2 rollforward db sample to 1998-04-03-14.21.56.245378 and stop
overflow log path (/logs)
```

例 5 (MPP)

0、1、および 2 の 3 つのノードがあります。表スペース TBS1 はすべてのノードで定義されており、表スペース TBS2 はノード 0 および 2 で定義されています。ノード 1 でデータベースを復元し、ノード 0 および 2 で TBS1 を復元した後、ノード 1 でデータベースをロールフォワードします。

```
db2 rollforward db sample to end of logs and stop
```

これにより、警告 SQL1271 (『データベースは回復されましたが、ノード 0 および 2 で 1 つまたは複数の表スペースがオフラインになっています。』) が戻されます。

```
db2 rollforward db sample to end of logs
```

これにより、ノード 0 および 2 で TBS1 がロールフォワードされます。この場合、文節 TABLESPACE(TBS1) は任意指定です。

例 6 (MPP)

ノード 0 および 2 でのみ表スペース TBS1 を復元した後、ノード 0 および 2 で TBS1 をロールフォワードします。

```
db2 rollforward db sample to end of logs
```

ノード 1 は無視されます。

```
db2 rollforward db sample to end of logs tablespace(TBS1)
```

ノード 1 で TBS1 がロールフォワード可能な状態になっていないため、これは失敗します。SQL4906N が報告されます。

```
db2 rollforward db sample to end of logs on nodes (0, 2) tablespace(TBS1)
```

これは正常に完了します。

```
db2 rollforward db sample to 1998-04-03-14.21.56.245378 and stop
tablespace(TBS1)
```

ノード 1 で TBS1 がロールフォワード可能な状態になっていないため、これは失敗します。すべての部分は一緒にロールフォワードされなければなりません。

ROLLFORWARD DATABASE

注: 表スペースをある時点までロールフォワードする場合、ノード文節は受け入れられません。

ノード 1 で TBS1 を復元した後

```
db2 rollforward db sample to 1998-04-03-14.21.56.245378 and stop
tablespace(TBS1)
```

これは正常に完了します。

例 7 (MPP)

すべてのノードで表スペースを回復した後、PIT2 までロールフォワードしますが、AND STOP は指定しません。ROLLFORWARD コマンドはまだ進行中ですが、それを取り消し、PIT1 までロールフォワードします。

```
db2 rollforward db sample to pit2 tablespace(TBS1)
db2 rollforward db sample cancel tablespace(TBS1)
** restore TBS1 on all nodes **
db2 rollforward db sample to pit1 tablespace(TBS1)
db2 rollforward db sample stop tablespace(TBS1)
```

例 8 (MPP)

db2nodes.cfg ファイルにリスト表示されている 8 個のノード (3 ~ 10) に存在する表スペースをロールフォワードします。

```
db2 rollforward database dwtest to end of logs tablespace (tssprodt)
```

ログの終わり (ある時点ではなく) までのこの操作は正常に完了します。表スペースが存在するノードは指定する必要がありません。ユーティリティーは、省略時値として db2nodes.cfg ファイルを使用します。

例 9 (MPP)

(ノード 6 上の) 1 つのノード・グループに存在する 6 個の小さな表スペースをロールフォワードします。

```
db2 rollforward database dwtest to end of logs on node (6)
tablespace(tsstore, tssbuyer, tsstime, tsswhse, tsslscat, tssvendor)
```

ログの終わり (ある時点ではなく) までのこの操作は正常に完了します。

使用上の注意

データベース・マネージャは、アーカイブおよびアクティブ・ログ・ファイルに保管されている情報を使用して、最終バックアップ以後にデータベースで実行されたトランザクションを再作成します。

ROLLFORWARD DATABASE が呼び出される際、データベースがロールフォワード保留状態であると、データベースはロールフォワードされます。データベースのロールフォワードが成功すると、異常状態が生じて 1 つまたは複数の表スペースがオフラインとならない限り、表スペースは通常の状態に戻されます。

データベースがロールフォワード保留状態ではなく、時点が指定されない場合は、ロールフォワード進行状態にあるすべての表スペースがログの終わりまでロールフォワードされます。ロールフォワード進行状態の表スペースがない場合は、ロールフォワード保留状態にあるすべての表スペースがログの終わりまでロールフォワードされます。

表スペース名を指定すると、表スペースのサブセットに対してロールフォワード処理を行うことができます。

表スペースをある時点までロールフォワードする場合、表スペースのサブセットを指定しなければなりません。指定された表スペースだけがロールフォワードされます。各表スペースはロールフォワード保留状態でなければなりません。あるいは、表スペースのロールフォワードがすでに進行中の場合、表スペースはロールフォワード進行状態でなければなりません。

ロールフォワード回復処理のために既存のデータベースを使用可能にする場合には、1 次ログ・ファイルの数を 1 次ログ・ファイルと 2 次ログ・ファイルの合計数 +1 に変更してください。詳しい情報については、ロールフォワード回復処理に使用可能なデータベースにおいて、LONG VARCHAR フィールドおよび LOB データのためにログに記録されます。

データベースのロールフォワードには、テープ装置を使用したロード回復が必要とされる場合があります。別のテープを要求されたときは、次のうちのいずれか 1 つで応答することができます。

- c** 続行。警告メッセージを生成した装置の使用を続けます (たとえば、新しいテープをマウントしたときなど)。
- d** 装置の終了。警告メッセージを生成した装置の使用を停止します (たとえば、それ以上テープがない場合)。
- t** 終了。すべての装置を終了します。

データベースのロールフォワードには、この解説書に記述されたもの以外の前提条件と制限が含まれる可能性があります。より詳細な情報については、管理の手引きを参照してください。

ROLLFORWARD DATABASE

以下も参照

382ページの『LIST HISTORY』

411ページの『LOAD』

539ページの『RESTORE DATABASE』

RUNSTATS

表およびそれに関連した索引の物理的特性についての統計を更新します。これらの特性には、レコード数、ページ数、および平均レコード長が含まれます。最適化プログラムは、データへのアクセス・パスを決定する際にこれらの統計を使用します。

このユーティリティーは、表が数多く更新されるとき、または表を再編成した後で、呼び出してください。

効力範囲

このコマンドは、`db2nodes.cfg` ファイルのどのノードからでも発行できます。カタログ・ノードのカタログを更新するのに使用します。

このコマンドは、表スペースの呼び出し元のノードの統計を収集します。表がそのノードに存在しない場合、ノード・グループの最初のノードが選択されます。

許可

以下のどれかが必要です。

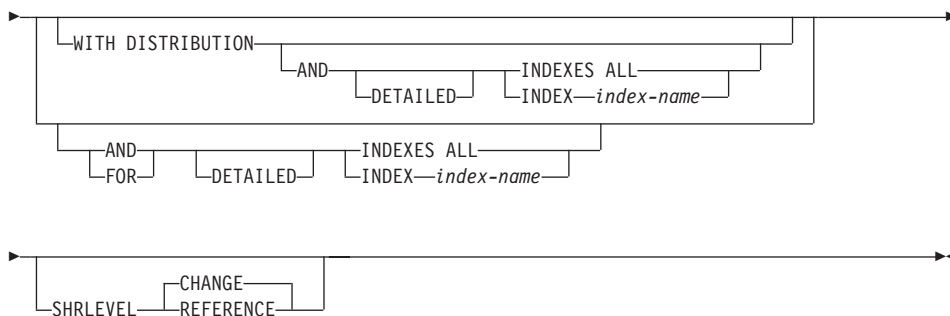
- `sysadm`
- `sysctrl`
- `sysmaint`
- `dbadm`
- 表に対する CONTROL 特権

必須接続

データベース

コマンド構文

▶▶—RUNSTATS ON TABLE—*table-name*—————▶



コマンド・パラメーター

TABLE table-name

統計を更新する表。完全修飾の名前または別名の書式は、*schema.table-name* が使用されなければなりません。 *schema* には、表作成時のユーザー名が入ります。オプションが 1 つも指定されない場合には、表の統計のみが更新されます。その表は、統計がまとめられている間、その他のユーザーによってアクセスされます。

型既定表の場合、指定するテーブル名は階層のルート表の名前でなければなりません。

WITH DISTRIBUTION

分布統計が要求されることを指定します。収集される最も頻度の高い値の数は、 *num_freqvalues* データベース構成パラメーターによって定義されます。収集される変位の数は、 *num_quantiles* データベース構成パラメーターによって定義されます。変動する分布統計についての情報は、*管理の手引き* を参照してください。

AND INDEXES ALL

表とその索引の両方についての統計を更新します。

AND INDEX index-name

表と指定された索引の両方についての統計を更新します。 *index-name* は、次の書式の完全修飾名です *schema.index-name*。

FOR INDEXES ALL

索引のみの統計を更新します。表についての統計を一度も生成したことがない場合には、データベース・マネージャーは、索引の統計と同様に表についての統計も計算します。

FOR INDEX index-name

指定された索引のみの統計を更新します。表の統計を一度も生成したこ

とがない場合には、データベース・マネージャーは、索引の統計と同様に表についての統計も計算します。 `index-name` は、`schema.index-name` の書式の完全修飾名です。

DETAILED

拡張された索引統計を計算します。

SHRLEVEL

CHANGE

統計が計算される間に、他のユーザーがその表から読み込んだりそこに書き込んだりできることを指定します。

REFERENCE

統計が計算される間に、他のユーザーがその表に対して、読み取り専用のアクセスを行うことができることを指定します。

例

分布統計なしで、表のみの統計を収集します。

```
db2 runstats on table smith.table1
```

分布統計とともに、表のみの統計を収集します。

```
db2 runstats on table smith.table1 with distribution
```

索引のみの基本統計を収集します。

```
db2 runstats on table smith.table1 for indexes all
```

表とそのすべての索引 (基本レベル) の統計を収集します。

```
db2 runstats on table smith.table1 and indexes all
```

分布統計および索引統計とともに、表の統計を収集します。

```
db2 runstats on table smith.table1 with distribution and indexes all
```

可能なすべての統計 (分布および拡張索引) を収集します。

```
db2 runstats on table smith.table1 with distribution and detailed index
```

索引 INDEX1 のみの分布統計を収集します。

```
db2 runstats on table smith.table1 with distribution for index smith.index1
```

使用上の注意

このコマンドは、宣言一時表に対してはサポートされていません。

このユーティリティーは、通称の使用はサポートしていません。

RUNSTATS

RUNSTATS は、以下の統計を更新するために使用します。

- 何度も変更されている表 (たとえば、多くの更新が行われている場合、あるいは大量のデータが挿入または削除されている場合)
- 再編成されている表
- 新しい索引が作成されたとき

統計が更新された後、その表に対する新しいアクセス・パスが、165ページの『BIND』を使用してパッケージを再バインドすることによって作成できません。

索引の統計が要求され、索引を含む表に対して統計が実行されたことがない場合は、表と索引の両方に関する統計が計算されます。

このコマンドを発行した後、COMMIT を発行してロックを解除しなければなりません。

新規のアクセス・プランが生成されるようにするには、ターゲット表を参照するパッケージを、このコマンドの発行後に再バインドしなければなりません。

統計は、コマンドが実行されるデータベース区画に存在する表データに基づいて収集されます。区分されている表全体に関するグローバルな表統計は、データベース区画で取得された値に、表が区分されているノード・グループ内のデータベース区画の数を掛けることによって得られます。グローバル統計はカタログ表に保管されます。

コマンドが発行されるデータベース区画には、表の区画が含まれていなくても構いません。

- 表の区画を含むデータベース区画からコマンドが発行されると、ユーティリティはこのデータベース区画で実行されます。
- 表区画を含まないデータベース区画からコマンドが発行されると、要求は、ノード・グループ内で表の区画を保持している最初のノードに送られます。ユーティリティは、このデータベース区画で実行されます。

このコマンドの一部を実行しているときに矛盾 (コマンドの最後の実行以後に表に対して行われた活動から生じた) が検出されると、警告メッセージが戻されます。たとえば、1 回目の発行で表の分布統計が収集され、2 回目の発行で索引統計のみが収集された後、表に対する活動の結果として矛盾が検出されると、表の分布統計は除去されます。この時点で、コマンドを再び発行して表の分布統計を最新表示することをお勧めします。

以下も参照

279ページの『GET DATABASE CONFIGURATION』

517ページの『REORGANIZE TABLE』

520ページの『REORGCHK』

SET CLIENT

SET CLIENT

バック・エンド・プロセス用の接続設定を指定します。

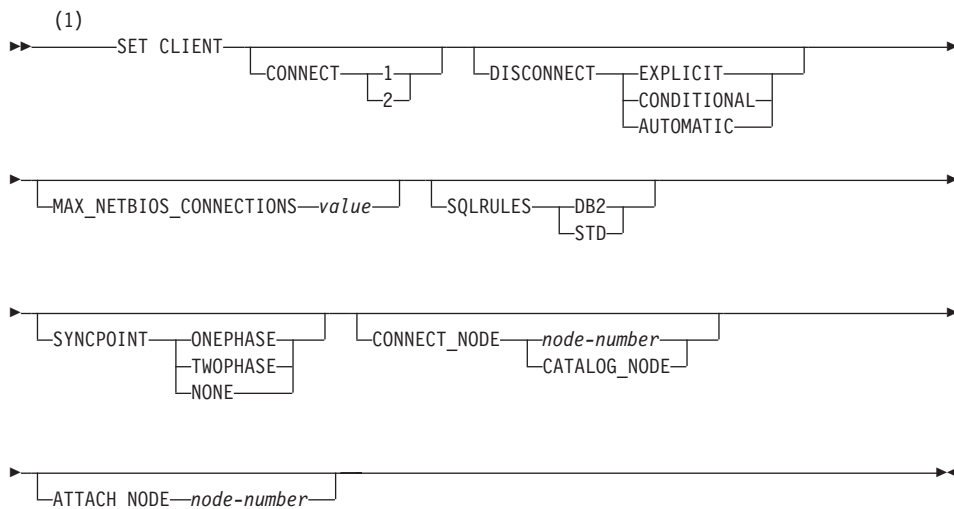
許可

なし

必須接続

なし

コマンド構文



注:

- 1 オプションを指定しない場合、省略時値は設定変更なしです。

コマンド・パラメーター

CONNECT

- 1 CONNECT ステートメントをタイプ 1 の CONNECT として処理するよう指定します。
- 2 CONNECT ステートメントをタイプ 2 の CONNECT として処理するよう指定します。

DISCONNECT**EXPLICIT**

RELEASE ステートメントで明示的に解放をマークしたデータベース接続だけを、コミット時に切断するよう指定します。

CONDITIONAL

RELEASE をマークしたか、またはオープン状態の WITH HOLD カーソルをもたないデータベース接続を、コミット時に切断するよう指定します。

AUTOMATIC

コミット時にすべてのデータベース接続を切断するよう指定します。

MAX_NETBIOS_CONNECTIONS value

NetBIOS アダプターを使用したアプリケーションで可能な同時接続の最大数を指定します。最大値は、254 です。このパラメーターは、最初の NetBIOS 接続が行われる前に、設定する必要があります。最初の接続以降の変更は、無視されます。

SQLRULES

DB2 タイプ 2 CONNECT が、DB2 規則に従って処理されることを指定します。

STD タイプ 2 CONNECT が、ISO/ANS SQL92 に基づく標準 (STD) 規則に従って処理されることを指定します。

SYNCPOINT

複数のデータベース接続にまたがってコミットまたはロールバックを調整する仕方を指定します。

ONEPHASE

2 フェーズ・コミットを実行するのに使用されるトランザクション・マネージャー (TM) がないことを指定します。複数のデータベース・トランザクションの各データベースが行う作業をコミットするときは、1 フェーズ・コミットが使用されません。

TWOPHASE

このプロトコルをサポートする複数のデータベースにまたがって 2 フェーズ・コミットを調整するのに TM が必要であることを指定します。

NONE 2 フェーズ・コミットを実行するのに使用される TM がなく、単一の更新プログラムと複数の読み込みプログラムという

SET CLIENT

形を強制しないことを指定します。コミットは、関連する各データベースに送られます。コミットが失敗したときの回復は、アプリケーションが行います。

CONNECT_NODE (MPP のみ)

node-number

接続先のノードを指定します。値は 0~999 (0 および 999 を含む) です。環境変数 **DB2NODE** の値を上書きします。

CATALOG_NODE

この値を指定すると、クライアントは、データベースのカタログ・ノードをあらかじめ認識していなくても、そのノードに接続できるようになります。

ATTACH_NODE (MPP のみ) node-number

接続先のノードを指定します。値は 0~999 (0 および 999 を含む) です。環境変数 **DB2NODE** の値を上書きします。

たとえば、ノード 1、2、および 3 が定義された場合、クライアントはこれらのうちの 1 つのノードにアクセスできればよいこととなります。データベースを含むノード 1 がカタログ化され、このパラメータが 3 に設定されると、次の接続は、まずノード 1 に行われ、次いでノード 3 に行われます。

例

特定の値を設定するには、次のように出します。

```
db2 set client connect 2 disconnect automatic sqlrules std
syncpoint twophase
```

SQLRULES を DB2 に戻し、それ以外の設定はそのままにしておくには、次のように出します。

```
db2 set client sqlrules db2
```

注: 接続の設定内容は、586ページの『TERMINATE』が出された後で、省略時値に戻ります。

使用上の注意

SET CLIENT は、1 つまたは複数の接続が活動状態である場合には発行できません。

SET CLIENT が正常であると、それに続く作業単位の接続が、指定された接続の設定内容を使用します。SET CLIENT が異常であると、バック・エンド・プロセスの接続設定内容が、未変更のままになります。

分散作業単位 (DUOW) の詳細については、[管理の手引き](#) を参照してください。

以下も参照

489ページの『QUERY CLIENT』

SET RUNTIME DEGREE

SET RUNTIME DEGREE

指定した活動アプリケーションの SQL ステートメントについて、区画内並行性の実行時の最大レベルを設定します。

効力範囲

このコマンドは、\$HOME/sqllib/db2nodes.cfg ファイルにリストされているすべてのノードに影響を与えます。

許可

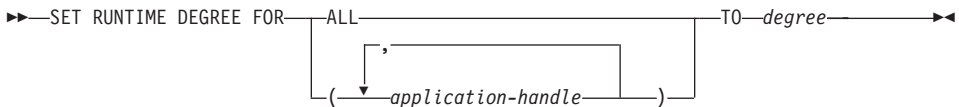
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

インスタンス。リモート・サーバーでの区画内並行性の実行時の最大レベルを変更するには、最初にそのサーバーに接続する必要があります。接続がない場合、SET RUNTIME DEGREE コマンドの実行は失敗します。

コマンド構文



コマンド・パラメーター

FOR

ALL 指定した並行性レベルはすべてのアプリケーションに適用されます。

application-handle

新しい並行性レベルが適用されるエージェントを指定します。
364ページの『LIST APPLICATIONS』を使用して値をリストします。

TO degree

区画内並行性の実行時最大レベル。

例

次の例では、*application-handle* の値が 41408 と 55458 の 2 つのユーザーについて、並行性の実行時最大レベルを 4 に設定します。

```
db2 SET RUNTIME DEGREE FOR ( 41408, 55458 ) TO 4
```

使用上の注意

このコマンドは、活動中のアプリケーションの最大並行性レベルを修正するメカニズムを提供します。また、SQL ステートメント・コンパイル時間で決定された値を上書きするときにも使用します。

区画内並行性の実行時レベルは、ステートメントの実行時に使用される並列操作の最大数を指定します。SQL ステートメントの区画内並行性のレベルは、ステートメントのコンパイル時に、CURRENT DEGREE 特殊レジスターまたは **degree** バインド・オプションを使用して指定することができます。活動アプリケーションの区画内並行性の実行時の最大レベルは、SET RUNTIME DEGREE コマンドを使用して指定することができます。データベース・マネージャーの *max_querydegree* 構成パラメーターは、データベース・マネージャーのこのインスタンスを実行する SQL ステートメントの最大実行時並行性レベルを指定します。

実際の実行時並行性レベルは次の 3 つのうち、最も低いものです。

- *max_querydegree* 構成パラメーター
- アプリケーションの実行時並行性レベル
- SQL ステートメントのコンパイル度

SET TABLESPACE CONTAINERS

SET TABLESPACE CONTAINERS

リダイレクト復元とは、復元されたデータベースの表スペース・コンテナのセットが、バックアップが行われた時の元のデータベースの表スペース・コンテナと異なる復元のことです。このコマンドでは、復元されるデータベースのための表スペース・コンテナの追加、変更、または除去が可能となります。たとえば、1 つまたは複数のコンテナが何かの理由でアクセス不能となった場合、復元が別のコンテナにリダイレクトされない限り復元は失敗します。

注: ユーザー出口プログラムが復元のために使用されている場合、リダイレクト復元はできません。

許可

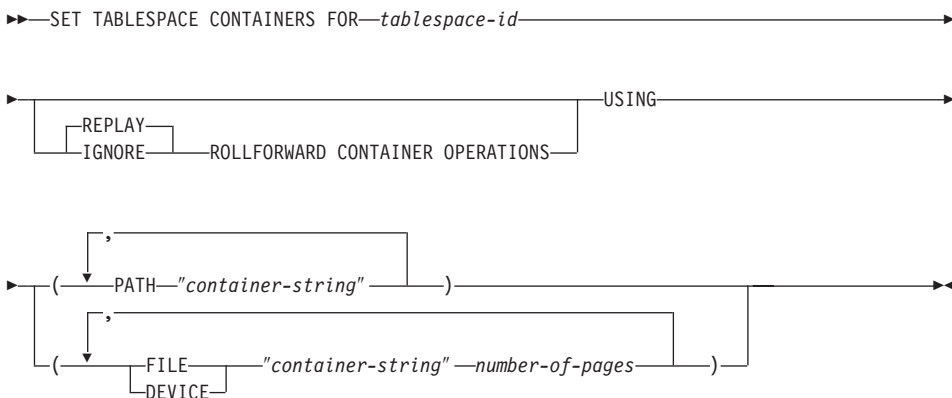
以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

データベース

コマンド構文



コマンド・パラメーター

FOR *tablespace-id*

復元されるデータベースが使用する表スペースを表す固有の整数。

REPLAY ROLLFORWARD CONTAINER OPERATIONS

データベースのバックアップが行われた後にこの表スペースに対して発行された ALTER TABLESPACE 操作が、それ以後のデータベースのロールフォワード処理中に再実行されることを指定します。

IGNORE ROLLFORWARD CONTAINER OPERATIONS

ロールフォワード中に、ログ内の ALTER TABLESPACE 操作が無視されることを指定します。

USING PATH "container-string"

SMS 表スペースに関して、表スペースに属し、表スペース・データの保管先となる、1 つまたは複数のコンテナを識別します。絶対または相対ディレクトリー名です。ディレクトリー名が絶対でない場合、データベース・ディレクトリーに対して相対的です。ストリング長は 240 バイト以下です。

USING FILE/DEVICE "container-string" number-of-pages

DMS 表スペースの場合、表スペースに属し、表スペース・データの保管先となる、1 つまたは複数のコンテナを識別します。コンテナ・タイプ (FILE または DEVICE) およびコンテナ・サイズ (4KB ページ) を指定します。ファイル・コンテナおよび装置コンテナの混在も可能です。ストリング長は 254 バイト以下です。

ファイル・コンテナの場合、ストリングは絶対または相対ファイル名です。ファイル名が絶対ファイル名でない場合、データベース・ディレクトリーに対して相対的です。

装置コンテナの場合、ストリングは装置名でなければなりません。また、装置がすでに存在していなければなりません。装置コンテナは OS/2 ではサポートされていません。

例

539ページの『RESTORE DATABASE』の例を参照してください。

使用上の注意

このコマンドは 539ページの『RESTORE DATABASE』とともに使用します。

データベースのバックアップ、または 1 つまたは複数の表スペースには、バックアップされた表スペースにより使用されている表スペース・コンテナのすべての記録が保持されています。復元操作中に、バックアップにリストされているすべてのコンテナは、現在存在し、アクセス可能かどうかをチェックされます。1 つまたは複数のコンテナが何らかの理由でアクセス不能の場合、復元は失敗します。そのようなときに復元したい場合、表スペース・コンテナ

SET TABLESPACE CONTAINERS

一のリダイレクトがサポートされています。このサポートには、表スペース・コンテナの追加、変更、または除去が含まれます。ユーザーはこのコマンドを使用して、これらのコンテナの追加、変更、または除去ができます。詳しくは、*管理の手引き* を参照してください。

以下も参照

157ページの『BACKUP DATABASE』

539ページの『RESTORE DATABASE』

550ページの『ROLLFORWARD DATABASE』

SET TAPE POSITION

DB2 (Windows NT 版) は、ストリーム・テープ装置へのバックアップおよび復元をサポートしています。このコマンドを使用して、テープの位置決めを行います。

このコマンドは Windows NT でのみ使用可能です。

許可

なし

必須接続

なし

コマンド構文

▶▶—SET TAPE POSITION—ON—device—TO—*position*—▶▶

コマンド・パラメーター

ON device

有効なテープ装置名を指定します。省略時値は `¥¥.¥TAPE0` です。

TO position

テープ位置のマークを指定します。DB2 (Windows NT 版) は、バックアップの度にテープ・マークを書き込みます。値 1 は 1 番目の位置、2 は 2 番目の位置、以下同じ手順で指定します。テープがテープ・マーク 1 に位置している場合、アーカイブ 2 が復元される位置に置かれます。

以下も参照

359ページの『INITIALIZE TAPE』

549ページの『REWIND TAPE』

START DATABASE MANAGER

START DATABASE MANAGER

単一ノードまたはマルチノード環境で定義されているすべてのノードで、現行のデータベース・マネージャー・インスタンス背景処理を開始します。

このコマンドはクライアントでは無効です。

効力範囲

多重ノード環境では、このコマンドは、`$HOME/sql1lib/db2nodes.cfg` ファイルにリストされているすべてのノードに影響を与えます。ただし、`nodenum` パラメーターが使用されていない場合に限りです。

許可

以下のどれかが必要です。

- `sysadm`
- `sysctrl`
- `sysmaint`

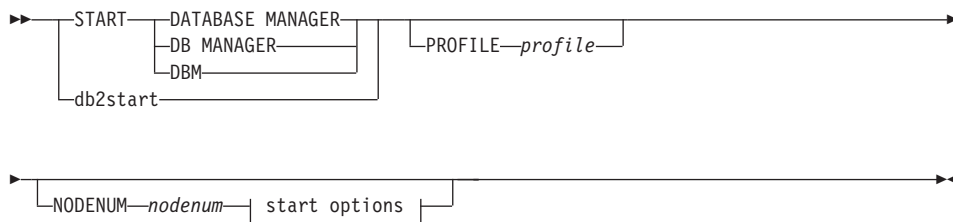
注:

1. OS/2 の場合、`ss_logon` データベース・マネージャー構成パラメーターが NO に設定されていれば、権限は必要ありません。
2. ADDNODE 開始オプションは、`sysadm` または `sysctrl` 権限のどちらかを必要とします。

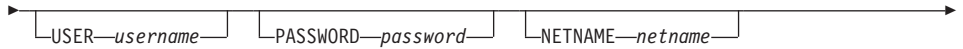
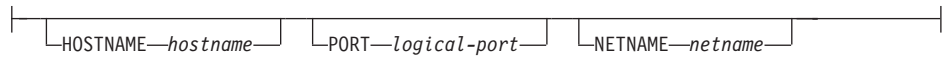
必須接続

なし

コマンド構文



start options:

**addnode options:****restart options:****コマンド・パラメーター**

注: 以下のパラメーターは MPP 環境でのみ有効です。

PROFILE profile

DB2 環境を定義するために、各ノードで実行しなければならないプロファイル・ファイル名を指定します。このファイルは、ノードの開始前に実行されます。プロファイル・ファイルはインスタンス所有者の `sqllib` ディレクトリーに常駐していなければなりません。

注: プロファイル・ファイルの環境変数をすべてユーザー・セッションで定義する必要はありません。

NODENUM nodenum

開始するノードを指定します。他のオプションが指定されていない場合、このノードで通常の始動が行われます。

有効な値は、0～999 (0 および 999 を含む) です。ADDNODE が指定されていない場合、値はインスタンス所有者の `db2nodes.cfg` ファイ

START DATABASE MANAGER

ルにすでに存在していなければなりません。ノード番号が指定されていない場合、ノード構成ファイルで定義されているすべてのノードが開始されます。

ADDNODE

インスタンス所有者の `db2nodes.cfg` ファイルに、`hostname` および `logical-port` 値とともに、新しいノードを追加することを指定します。

`hostname` および `logical-port` の組み合わせが固有であることを確認してください。

ノード追加ユーティリティーが内部で実行され、追加されたノードに既存のデータベースすべてが作成されます。ノード追加後、**db2stop** が発行されるまで、`db2nodes.cfg` ファイルは新しいノードで更新されません。**db2stop** に続く **db2start** が実行されるまで、ノードは MPP システムの一部とはみなされません。

注: 新しいノードにデータベース区画が作成される場合、構成パラメーターは省略時値に設定されます。

HOSTNAME `hostname`

ADDNODE を使用して、`db2nodes.cfg` ファイルに追加するホスト名を指定します。

PORT `logical-port`

ADDNODE を使用して、`db2nodes.cfg` ファイルに追加する論理ポートを指定します。有効な値は 0 ~ 999 です。

COMPUTER `computer-name`

新しいノードが作成されるマシンのコンピューター名。このパラメーターは、Windows NT では必須ですが、その他のオペレーティング・システムでは無視されます。

USER `username`

新しいノード上のアカウントのユーザー名。このパラメーターは、Windows NT では必須ですが、その他のオペレーティング・システムでは無視されます。

PASSWORD `password`

新しいノード上のアカウントのパスワード。このパラメーターは、Windows NT では必須ですが、その他のオペレーティング・システムでは無視されます。

NETNAME netname

db2nodes.cfg ファイルに追加する *netname* を指定します。指定されていない場合、*hostname* で指定された値が省略時値となります。

LIKE NODE node-number

システム一時表スペース用のコンテナが、インスタンス内の各データベース用に指定した *node-number* のコンテナと同一になるように指定します。指定するノードは、db2nodes.cfg ファイル中にすでに指定してあるノードでなければなりません。

WITHOUT TABLESPACES

システム一時表スペースのコンテナがどのデータベースに対しても作成されないことを指定します。データベースを使用する前に、ALTER TABLESPACE ステートメントを使用して、システム一時表スペース・コンテナを各データベースに追加しなければなりません。

STANDALONE

ノードが独立方式で開始されることを指定します。FCM は他のノードとの接続を確立しようとはしません。このオプションはノードの追加の時に使用します。

RESTART

障害発生後、データベース・マネージャーを起動します。他のノードの操作は続いており、このノードは他のノードとの接続を試みます。

hostname と *logical-port* のどちらのパラメーターも指定されないと、データベース・マネージャーは、db2nodes.cfg で指定された *hostname* および *logical-port* 値を用いて再始動されます。どちらかのパラメーターが指定されている場合は、接続の確立時に新しい値が他のノードに送信されます。db2nodes.cfg ファイルは、この情報に基づいて更新されます。

HOSTNAME hostname

RESTART を使用して、ノード構成ファイルにあるホスト名を上書きするために使用するホスト名を指定します。

PORT logical-port

RESTART を使用して、ノード構成ファイルにある論理ポート番号を上書きするために使用する論理ポート番号を指定します。指定されていない場合、db2nodes.cfg ファイルの

START DATABASE MANAGER

nodenum の値に対応する、*logical-port* の値が省略時値となります。有効な値は 0 ~ 999 です。

NETNAME netname

db2nodes.cfg ファイルで指定されたネット名を上書きするために使用される *netname* を指定します。指定されていない場合、db2nodes.cfg ファイルの *nodenum* の値に対応する、*netname* の値が省略時値となります。

例

次に示すのは、ノード 10、20、および 30 を使用する 3 ノード・システムで発行された **db2start** からの出力例です。

```
04-07-1997 10:33:05    10  0  SQL1063N  DB2START processing was successful.
04-07-1997 10:33:07    20  0  SQL1063N  DB2START processing was successful.
04-07-1997 10:33:07    30  0  SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.
```

使用上の注意

このコマンドをクライアント・ノードで発行しない場合もあります。旧クライアントとの互換性が提供されていますが、データベース・マネージャーには何も影響ありません。

一度開始されると、データベース・マネージャーのインスタンスは、たとえそれを使用していたすべてのアプリケーション・プログラムが終了したとしても、ユーザーが停止しない限り稼働しています。

データベース・マネージャーが正常に開始されると、正常終了メッセージが標準出力装置に送られます。エラーが発生すると、処理は停止され、エラー・メッセージが標準出力装置に送られます。マルチノード環境では、メッセージは START DATABASE MANAGER コマンドを発行したノードに戻されます。

マルチノード・データベース環境でパラメーターが指定されていない場合、ノード構成ファイルで指定されたパラメーターを使用してすべて並列ノードでデータベース・マネージャーが開始されます。

START DATABASE MANAGER コマンド実行中の場合、データベースへの要求を発行する前に、適用可能なノードが開始されていることを確認してください。

db2cshrc ファイルはサポートされておらず、環境の定義付けに使用できません。

UNIX プラットフォームでは、START DATABASE MANAGER コマンドは SIGINT および SIGALRM 信号をサポートしています。CTRL+C を押すと、SIGINT 信号が発行されます。データベース・マネージャー構成パラメーターの *start_stop_time* の指定値が着信すると、SIGALRM 信号が発行されます。いずれかの信号が発行されると、ノードの始動操作に割り込みが生じ、メッセージ (SIGINT の場合は SQL1044N、SIGALRM の場合は SQL6037N) が割り込みが生じたノードから、\$HOME/sqllib/log/db2start.timestamp.log エラー・ログ・ファイルに戻されます。すでに開始済みのノードには影響がありません。開始しているノードに対して CTRL+C が押された場合、そのノードを再始動する前に、そのノードに対して **db2stop** が発行されなければなりません。

Windows NT オペレーティング・システムの場合、開始することを失敗した通信サブシステムがあったとしても、**db2start** コマンドも **NET START** コマンドも警告を戻しません。Windows NT 環境のデータベース・マネージャーは NT サービスとして実装されます。サービスが正常に開始された場合にはエラーを戻しません。NT イベント・ログまたは DB2DIAG.LOG ファイルを調べて、**db2start** の実行中にエラーが発生しなかったか確認してください。

以下も参照

152ページの『ADD NODE』

582ページの『STOP DATABASE MANAGER』

STOP DATABASE MANAGER

現行のデータベース・マネージャー・インスタンスを停止します。明示的に停止されない限り、データベース・マネージャーは、活動状態のまま続きます。データベースに接続されたアプリケーションがある場合、このコマンドはデータベース・マネージャーを停止しません。データベース接続はなくても、インスタンス接続機構がある場合には、そのインスタンス接続機構を強制的に使用して、データベース・マネージャーを停止させます。また、データベース・マネージャーを停止させる前に、活動化されている主だったデータベースを非活動化します。

MPP システムでは、このコマンドは、特定のまたはすべてのノードにある現行のデータベース・マネージャーのインスタンスを停止します。すべてのノードでデータベース・マネージャーを停止する場合、ノード構成ファイル `db2nodes.cfg` を使用して、各ノードに関する情報を入手します。

このコマンドを使用して、`db2nodes.cfg` ファイルからノードを消去することもできます (MPP システムの場合のみ)。

このコマンドはクライアントでは無効です。

効力範囲

省略時および多重ノード環境では、このコマンドは、`db2nodes.cfg` ファイルにリストされているすべてのノードに影響を与えます。

許可

以下のどれかが必要です。

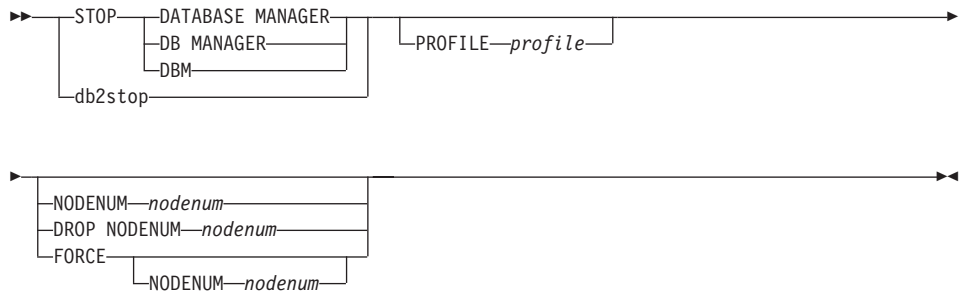
- `sysadm`
- `sysctrl`
- `sysmaint`

注: OS/2 の場合、`ss_logon` データベース・マネージャー構成パラメーターが `NO` に設定されていれば、権限は必要ありません。

必須接続

なし

コマンド構文



コマンド・パラメーター

PROFILE *profile*

MPP の場合のみ使用します。開始されたノード用に DB2 環境を定義するために、始動時に実行されたプロファイル・ファイル名を指定します。576ページの『START DATABASE MANAGER』でプロファイルが指定されると、ここでも同じプロファイルを指定しなければなりません。プロファイル・ファイルはインスタンス所有者の `sqllib` ディレクトリに常駐していなければなりません。

NODENUM *nodenum*

MPP の場合のみ使用します。停止するノードを指定します。

有効な値は 0~999 で、その値が `db2nodes.cfg` ファイルになければなりません。ノード番号が指定されていない場合、ノード構成ファイルで定義されているすべてのノードが停止されます。

DROP NODENUM *nodenum*

MPP の場合のみ使用します。 `db2nodes.cfg` ファイルから消去するノードを指定します。

このパラメーターを使用する前に、254ページの『DROP NODE VERIFY』を実行して、このノードにユーザー・データが存在しないことを確認してください。

このオプションを指定した場合、 `db2nodes.cfg` ファイルにあるすべてのノードが停止します。

FORCE

各ノードでデータベース・マネージャーを停止する際に `FORCE APPLICATION ALL` を使用することを指定します。

NODENUM *nodenum*

MPP の場合のみ使用します。ノード上のすべてのアプリケーションが

STOP DATABASE MANAGER

強制的に停止された後で停止されるノード（データベース区画サーバ）を指定します。このパラメーターを指定せずに **FORCE** オプションを使用すると、すべてのノードが停止される前に、すべてのノード上のすべてのアプリケーションが強制的に停止されます。

例

次に示すのは、ノード 10、20、および 30 を使用する 3 ノード・システムで発行された **db2stop** からの出力例です。

```
04-07-1997 10:32:53    10    0    SQL1064N  DB2STOP processing was successful.
04-07-1997 10:32:54    20    0    SQL1064N  DB2STOP processing was successful.
04-07-1997 10:32:55    30    0    SQL1064N  DB2STOP processing was successful.
SQL1064N  DB2STOP processing was successful.
```

使用上の注意

このコマンドをクライアント・ノードで発行しない場合もあります。旧クライアントとの互換性が提供されていますが、データベース・マネージャーには何も影響ありません。

一度開始されると、データベース・マネージャーのインスタンスは、たとえそれを使用していたすべてのアプリケーション・プログラムが終了したとしても、ユーザーが停止しない限り稼働しています。

データベース・マネージャーが停止されると、正常終了メッセージが標準出力装置に送られます。エラーが発生すると、処理は停止され、エラー・メッセージが標準出力装置に送られます。

アプリケーション・プログラムがまだデータベースに接続されているため、データベース・マネージャーが停止できない場合には、267ページの『**FORCE APPLICATION**』を使用して、まず最初にすべてのユーザーを切断するか、**FORCE** オプションで **STOP DATABASE MANAGER** コマンドを再発行してください。

次の情報は、現時点では多重ノード環境にのみ適用されます。

- パラメーターが指定されない場合、データベース・マネージャーはノード構成ファイルにリストされている各ノードで停止します。db2diag.log ファイルには、他のノードが遮断されていることを示すメッセージが含まれていません。
- 前の **STOP DATABASE MANAGER** コマンドが発行されてから **MPP** システムに追加されたノードは、db2nodes.cfg ファイル内で更新されます。

- UNIX プラットフォームでは、このコマンドは SIGALRM 信号をサポートします。SIGALRM 信号は、データベース・マネージャー構成パラメーターの *start_stop_time* の指定値が着信すると発行されます。この信号が発行されると、ノードの停止操作に割り込みが生じ、メッセージ SQL6037N が割り込みが生じたノードから、`$HOME/sql1lib/log/db2stop.timestamp.log` エラー・ログ・ファイルに戻されます。すでに停止しているノードには影響がありません。
- `db2cshrc` ファイルはサポートされておらず、`PROFILE` パラメーターの値として指定することはできません。

考慮事項: UNIX `kill` コマンドは、制御された終了および終結処置プロセスは行わずに、突然データベース・マネージャー・プロセスを終了するため、このコマンドをデータベース・マネージャーを終了するために使用することはできません。

以下も参照

243ページの『DEACTIVATE DATABASE』

254ページの『DROP NODE VERIFY』

267ページの『FORCE APPLICATION』

576ページの『START DATABASE MANAGER』

TERMINATE

TERMINATE

コマンド行プロセッサのバック・エンド・プロセスを明示的に終了させます。バック・エンド・プロセスとフロントエンド・プロセスの詳細については、136ページの『コマンド行プロセッサの設計』を参照してください。

許可

なし

必須接続

なし

コマンド構文

▶—TERMINATE—▶

コマンド・パラメーター

なし

使用上の注意

アプリケーションがデータベースに接続されているかまたは処理単位の途中で、**TERMINATE** コマンドを出すと、データベース接続は失われます。その場合、内部コミットは実行されます。

TERMINATE と **CONNECT RESET** は両方ともデータベースへの接続を中断しますが、**TERMINATE** のみがバックエンド・プロセスを終了します。

バック・エンド・プロセスが活動状態である間に、**db2start** および **db2stop** が実行された場合、**TERMINATE** を出すことが推奨されています。これによりバック・エンド・プロセスは、それ以上使用不可能なデータベース・マネージャーのインスタンスへの接続を保持することができなくなります。

セッション中に **DB2NODE** 環境変数が更新された場合、**MPP** システムのバック・エンド・プロセスも終了しなければなりません。この環境変数は、**MPP** 多重論理ノード構成内の座標ノード番号を指定する時に使用します。

UNCATALOG DATABASE

データベース項目をシステム・データベース・ディレクトリーから削除します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし。ディレクトリー操作は、ローカル・ディレクトリーだけに影響します。

コマンド構文

```
▶▶—UNCATALOG—DATABASE  
DB—database-alias—▶▶
```

コマンド・パラメーター

DATABASE database-alias

アンカタログするデータベースの別名を指定します。

使用上の注意

システム・データベース・ディレクトリーにある項目だけをアンカタログできます。ローカル・データベース・ディレクトリーにある項目は、251ページの『DROP DATABASE』を使用して削除できます。

データベースを再カタログ化する場合は、194ページの『CATALOG DATABASE』を使用してください。ノードにカタログ化されているデータベースをリストする場合は、369ページの『LIST DATABASE DIRECTORY』を使用してください。

下位レベル・サーバーと通信するときを使用される、データベースの承認タイプの変更は、最初にデータベースをアンカタログし、次に別のタイプでもう一度カタログ化することによって行えます。

注: ディレクトリーをキャッシュできる場合 (292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照)、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッ

シュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときにのみ最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベースを停止してから (**db2stop**)、再始動します (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

UNCATALOG DCS DATABASE

データベース接続サービス (DCS) ディレクトリーから項目を削除します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし。ディレクトリー操作は、ローカル・ディレクトリーだけに影響します。

コマンド構文

```

▶▶—UNCATALOG DCS—┬—DATABASE—database-alias————▶▶
                    └—DB————┘
  
```

コマンド・パラメーター

DATABASE database-alias

アンカタログする DCS データベースの別名を指定します。

使用上の注意

DCS データベースは、587ページの『UNCATALOG DATABASE』を使用してアンカタログできるリモート・データベースとして、システム・データベース・ディレクトリーにもカタログ化されています。

DCS ディレクトリーでデータベースを再カタログ化する場合は、199ページの『CATALOG DCS DATABASE』を使用してください。ノードにカタログ化されている DCS データベースをリストする場合は、378ページの『LIST DCS DIRECTORY』を使用してください。

注: ディレクトリーをキャッシュできる場合 (292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照)、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときにの

UNCATALOG DCS DATABASE

み最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベースを停止してから (**db2stop**)、再始動します (**db2start**)。別のアプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

UNCATALOG LDAP DATABASE

Lightweight Directory Access Protocol (LDAP) からデータベースを登録解除するのに使用します。

このコマンドは、Windows NT、Windows 98、Windows 95、AIX、および Solaris でのみ使用可能です。

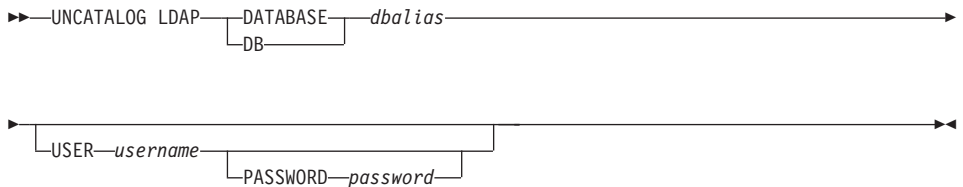
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

DATABASE dbalias

アンカタログする LDAP データベースの別名を指定します。

USER username

ユーザーの LDAP 識別名 (DN) を指定します。LDAP ユーザー DN には、LDAP ディレクトリーからオブジェクトを削除するための十分な権限が必要です。ユーザーの LDAP DN が指定されない場合、現行ログオン・ユーザーの認証が使用されます。

PASSWORD password

アカウント・パスワード。

使用上の注意

データベースを除去すると、データベース・オブジェクトも LDAP から除去されます。データベースを管理するデータベース・サーバーが LDAP から登

UNCATALOG LDAP DATABASE

録解除されると、データベースも LDAP から自動的に登録解除されます。ただし、次のようなときには、データベースを手動でアンカタログしなければならない場合があります。

- データベース・サーバーが LDAP をサポートしない場合。データベースが除去されるたびに、管理者はデータベースを LDAP から手動でアンカタログしなければなりません。
- DROP DATABASE を実行している間は、データベース・オブジェクトを LDAP から除去することができません (その間は LDAP にアクセスできないからです)。その場合、データベースはローカル・マシンから除去されますが、LDAP 内にある既存の項目は削除されません。

以下も参照

208ページの『CATALOG LDAP DATABASE』

213ページの『CATALOG LDAP NODE』

593ページの『UNCATALOG LDAP NODE』

UNCATALOG LDAP NODE

Lightweight Directory Access Protocol (LDAP) のノード項目をアンカタログします。

このコマンドは、Windows NT、Windows 98、Windows 95、AIX、および Solaris でのみ使用可能です。

許可

なし

必須接続

なし

コマンド構文

```

▶▶—UNCATALOG LDAP—NODE—nodename—┬──┬──┬──▶
                                     |   |   |
                                     |   |   └──PASSWORD—password—┬──▶
                                     |   └──USER—username—┬──▶

```

コマンド・パラメーター

NODE *nodename*

アンカタログするノードの名前を指定します。

USER *username*

ユーザーの LDAP 識別名 (DN) を指定します。LDAP ユーザー DN には、LDAP ディレクトリーからオブジェクトを削除するための十分な権限が必要です。ユーザーの LDAP DN が指定されない場合、現行ログオン・ユーザーの認証が使用されます。

PASSWORD *password*

アカウント・パスワード。

使用上の注意

DB2 サーバーが LDAP から登録解除されると、LDAP ノードは自動的にアンカタログされます。

以下も参照

208ページの『CATALOG LDAP DATABASE』

213ページの『CATALOG LDAP NODE』

UNCATALOG LDAP NODE

591ページの『UNCATALOG LDAP DATABASE』

UNCATALOG NODE

ノード・ディレクトリーから項目を削除します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*

必須接続

なし。ディレクトリー操作は、ローカル・ディレクトリーだけに影響します。

コマンド構文

▶▶—UNCATALOG NODE—*nodename*—————▶▶

コマンド・パラメーター

NODE*nodename*

アンカタログするノード項目を指定します。

使用上の注意

UNCATALOG NODE はどのタイプのノードでも実行できますが、リモート・インスタンスまたは別のインスタンスへの接続があっても、影響が及ぶのはローカル・ディレクトリーだけです。

注: ディレクトリーをキャッシュできる場合 (292ページの『GET DATABASE MANAGER CONFIGURATION』の構成パラメーター *dir_cache* を参照)、データベース、ノード、および DCS のディレクトリー・ファイルはメモリーにキャッシュされます。アプリケーションのディレクトリー・キャッシュは、最初のディレクトリー参照の間に作成されます。キャッシュはアプリケーションがディレクトリー・ファイルのどれかを修正したときのみ最新にされるため、他のアプリケーションが行ったディレクトリーの変更は、アプリケーションを再始動するまで有効にならないことがあります。

CLP のディレクトリー・キャッシュを最新表示するには、586ページの『TERMINATE』を使用します。DB2 の共用キャッシュを最新表示するには、データベースを停止してから (**db2stop**)、再始動します (**db2start**)。別の

UNCATALOG NODE

アプリケーション用のディレクトリー・キャッシュを最新にするには、そのアプリケーションを停止してから再始動してください。

以下も参照

186ページの『CATALOG APPC NODE』

189ページの『CATALOG APPCLU NODE』

191ページの『CATALOG APPN NODE』

204ページの『CATALOG IPX/SPX NODE』

215ページの『CATALOG LOCAL NODE』

221ページの『CATALOG NETBIOS NODE』

218ページの『CATALOG NAMED PIPE NODE』

226ページの『CATALOG TCP/IP NODE』

UNCATALOG ODBC DATA SOURCE

ユーザーまたはシステム ODBC データ・ソースをアンカタログします。

ODBC (オープン・データベース・コネクティビティ) でのデータ・ソースという語は、指定したデータベースまたはファイル・システムのユーザー定義名のことです。この名前は、ODBC API を介してデータベースまたはファイル・システムにアクセスするときに使用されます。Windows NT および Windows 95 では、ユーザー・データ・ソースまたはシステム・データ・ソースのいずれかのアンカタログができます。ユーザー・データ・ソースはそれをカタログ化したユーザーにのみ可視になりますが、システム・データ・ソースは他のすべてのユーザーから可視であり使用可能です。

このコマンドは、Windows NT、Windows 95、および Windows 3.1 でのみ使用可能です。

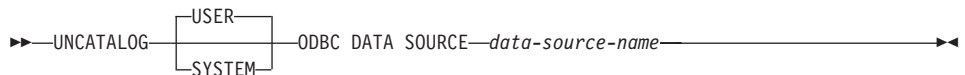
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

USER ユーザー・データ・ソースをアンカタログします。キーワードを指定しない場合、これが省略時値です。

SYSTEM

システム・データ・ソースをアンカタログします。

ODBC DATA SOURCE data-source-name

アンカタログするデータ・ソースの名前を指定します。最大長は 32 文字です。

以下も参照

224ページの『CATALOG ODBC DATA SOURCE』

UNCATALOG ODBC DATA SOURCE

398ページの『LIST ODBC DATA SOURCES』

UPDATE ADMIN CONFIGURATION

データベース・マネージャ構成ファイル内の、DB2 管理サーバーに関連する個々の項目を修正します。DB2 管理サーバーは、DB2 サーバーのリモート管理を使用可能にする特別な DB2 インスタンスです。次に示すデータベース・マネージャ構成パラメーターの修正ができます。

- AGENT_STACK_SZ
- AUTHENTICATION
- DIAGLEVEL
- DIAGPATH
- DISCOVER
- DISCOVER_COMM
- FILESERVER
- IPX_SOCKET
- NNAME
- OBJECTNAME
- QUERY_HEAP_SZ
- SYSADM_GROUP
- SYSCtrl_GROUP
- SYSMaint_GROUP
- TPNAME
- TRUST_ALLCLNTS
- TRUST_CLNTAUTH

注: インストール・プログラムで設定される SVCENAME パラメーターは、ユーザーが修正するようにお勧めします。管理サーバー・サービス名は、TCP/IP ポート (523) に登録済みの DB2 を使用するよう設定します。

上記のパラメーターの詳細については、292ページの『GET DATABASE MANAGER CONFIGURATION』を参照してください。

効力範囲

このコマンドは、db2nodes.cfg ファイルにリストされているどのノードからでも発行できます。また、このファイルにリストされているすべてのノードに影響を与えます。

UPDATE ADMIN CONFIGURATION

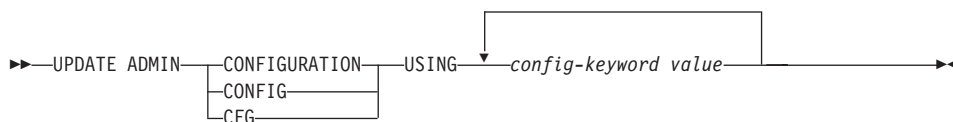
許可

sysadm

必須接続

なし、またはインスタンス。インスタンスとの接続は、ローカルの DBM 構成操作を実行する場合には必ずしも必要ではありませんが、リモートの DBM 構成操作の場合には必須です。リモート・インスタンスに対するデータベース・マネージャー構成を更新するためには、最初にそのインスタンスに接続する必要があります。

コマンド構文



コマンド・パラメーター

USING config-keyword value

更新する admin 構成パラメーターを指定します。

使用上の注意

ADMIN 構成パラメーターのリストを表示または印刷するには、269ページの『GET ADMIN CONFIGURATION』を使用してください。

admin 構成パラメーターを推奨されているデータベース・マネージャーの省略時値にリセットするには、528ページの『RESET ADMIN CONFIGURATION』を使用してください。

admin 構成パラメーターの詳細については、[管理の手引き](#)を参照してください。

これらのパラメーターの値は、構成するデータベース・ノードの各タイプ (サーバー、クライアント、またはリモート・クライアントを持つサーバー) によって異なります。各ノード・タイプで設定できる範囲と省略時値については、[管理の手引き](#)、または、[概説およびインストール 資料](#)のいずれかを参照してください。

データベース・マネージャー構成ファイルへの変更は、ファイルがメモリーにロードされた後にのみ有効になります。これは **db2start** の実行中に起こります。

エラーが生じた場合には、データベース・マネージャー構成ファイルは変更されません。

データベース・マネージャー構成ファイルは、そのチェックサムが無効であると、更新することができません。これは、データベース・マネージャー構成ファイルが、適切なコマンドを使用せずに変更された場合に起こることがあります。これが発生する場合、データベース・マネージャーを再インストールして、データベース・マネージャー構成ファイルをリセットする必要があります。

以下も参照

269ページの『GET ADMIN CONFIGURATION』

528ページの『RESET ADMIN CONFIGURATION』

UPDATE CLI CONFIGURATION

UPDATE CLI CONFIGURATION

db2cli.ini ファイル内の指定されたセクションの内容を更新します。

db2cli.ini ファイルは、DB2 コール・レベル・インターフェース (CLI) 構成ファイルとして使用されます。このファイルには、DB2 CLI およびそれを使用するアプリケーションの動作を変更するために使用できるさまざまなキーワードと値が含まれます。このファイルは複数のセクションに分かれており、それぞれのセクションはデータベース別名に対応します。このファイルと CLI/ODBC 構成キーワードの詳細については、[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。

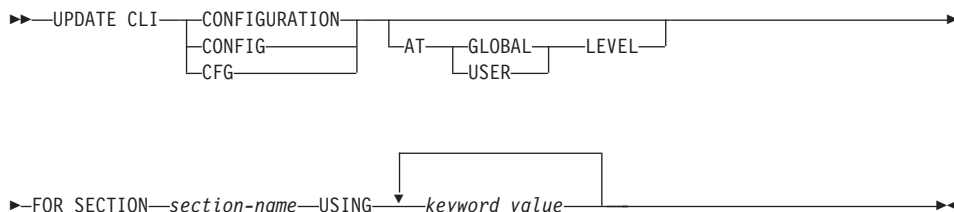
許可

sysadm

必須接続

なし

コマンド構文



コマンド・パラメーター

FOR SECTION section-name

キーワードが更新されるセクションの名前。指定されたセクションが存在しない場合は、新しいセクションが作成されます。

AT GLOBAL LEVEL

CLI 構成パラメーターをグローバル・レベルで更新するよう指定します。

注: このパラメーターを適用できるのは、LDAP サポートが使用可能な場合だけです。

AT USER LEVEL

CLI 構成パラメーターをユーザー・レベルで更新するよう指定します。このパラメーターを適用できるのは、LDAP サポートが使用可能な場合だけです。

USING keyword value

更新される CLI/ODBC パラメーターを指定します。

使用上の注意

このコマンドで指定されるセクション名とキーワードでは、大文字小文字が区別されません。ただし、キーワードの値では大文字小文字が区別されます。

キーワード値が単一引用符または組み込みブランクを含むストリングである場合には、ストリング全体を二重引用符で囲む必要があります。たとえば、次のようにします。

```
db2 update cli cfg for section tstclilx
using TableType "'TABLE','VIEW','SYSTEM TABLE'"
```

AT USERLEVEL キーワードを指定した場合、指定されたセクションの CLI 構成パラメーターは現行ユーザーについてのみ更新されます。指定しなかった場合は、ローカル・マシン上のすべてのユーザーについて更新されます。ユーザー・レベルの CLI 構成は、LDAP ディレクトリーに保持され、ローカル・マシンでキャッシュされます。CLI 構成を読み取るとき、DB2 は常にキャッシュから読み取ります。キャッシュは、次のときに更新されます。

- ユーザーが CLI 構成を更新するとき。
- ユーザーが REFRESH LDAP コマンドを使用して、明示的に CLI 構成の最新表示を強制するとき。

LDAP 環境では、ユーザーは LDAP ディレクトリーにカタログ化されたデータベースに対して、省略時 CLI 設定値のセットを構成することができます。LDAP カタログ化データベースが、DSN (データ・ソース名) として、CCA (クライアント構成アシスタント) または ODBC 構成ユーティリティーのどちらかを使用して追加されると、省略時の CLI 設定が LDAP ディレクトリーにある場合には、それらはローカル・マシン上のその DSN 用に構成されます。CLI パラメーターを省略時設定として構成するには、AT GLOBAL LEVEL 文節を指定する必要があります。

以下も参照

274ページの『GET CLI CONFIGURATION』

509ページの『REFRESH LDAP』

UPDATE COMMAND OPTIONS

UPDATE COMMAND OPTIONS

対話式セッションの間に、またはバッチ入力ファイルから、1 つまたは複数のコマンド・オプションを設定します。対話式セッションまたはバッチ入力ファイルが終了すると、設定値はシステム省略時値に戻ります (またはシステム省略時値は **DB2OPTIONS** の中で指定変更されます)。

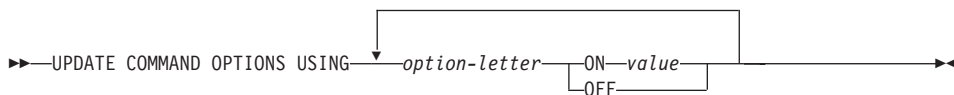
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

USING option-letter

次の option-letters を設定できます。

- a** SQLCA の表示
- c** SQL ステートメントの自動コミット
- e** SQLCODE/SQLSTATE の表示
- l** 活動記録ファイルへのコマンドのログ書き出し
- n** 改行文字の除去
- o** 標準出力への表示
- p** DB2 対話式プロンプトの表示
- r** 出力レポートのファイルへの保管
- s** コマンド・エラー時の実行停止
- v** 現行コマンドのエコー
- w** SQL ステートメント警告メッセージの表示
- z** 全出力のファイルへのリダイレクト

ON value

e、l、r、および z オプションは、オンにする場合に値が必要です。e オプションでは、*value* を、SQLCODE を表示する場合は c、SQLSTATE を表示する場合は s にできます。l、r、および z オプションの場合、*value* は、活動記録ファイルまたはレポート・ファイルに対して使用する名前を表します。他のオプションは値を受け付けません。

使用上の注意

これらの設定値は、システム省略時値、**DB2OPTIONS** の設定値、およびコマンド行オプション・フラグで指定したオプションを指定変更します。

ファイル入力オプション (-f)、およびステートメント終了オプション (-t) は、このコマンドを使用して更新できません。

現行オプションの設定値を表示する場合は、367ページの『LIST COMMAND OPTIONS』を使用してください。

詳しくは、125ページの『コマンド行プロセッサのオプション』を参照してください。

UPDATE DATABASE CONFIGURATION

UPDATE DATABASE CONFIGURATION

特定のデータベース構成ファイルの中の個々の項目を修正します。

データベース構成ファイルは、データベースが作成されたノードすべてに常駐しています。

効力範囲

このコマンドは、それが実行されたノードに対してだけ影響を与えます。

許可

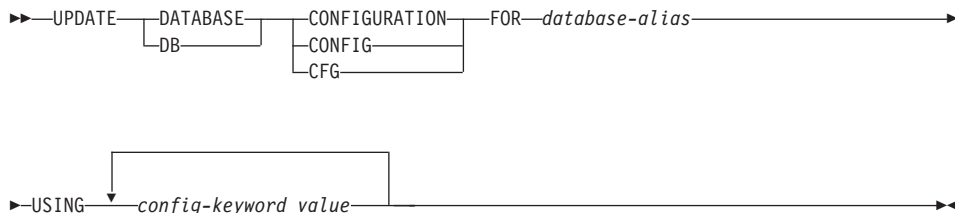
以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

必須接続

インスタンス。明示的な接続は必要ありません。データベースがリモートとして示されている場合、リモート・ノードへのインスタンス接続はコマンドの持続期間の間、ずっと確立されたままになります。

コマンド構文



コマンド・パラメーター

FOR database-alias

構成を更新するデータベースの別名を指定します。

USING config-keyword value

更新するデータベース構成パラメーターを指定します。構成可能なパラメーターの要旨については、279ページの『GET DATABASE CONFIGURATION』を参照してください。

使用上の注意

データベース構成パラメーターのリストを表示または印刷するには、279ページの『GET DATABASE CONFIGURATION』を使用してください。

データベース構成パラメーターを推奨されているデータベース・マネージャーの省略時値にリセットするには、531ページの『RESET DATABASE CONFIGURATION』を使用してください。

DB2 構成パラメーターの詳細については、*管理の手引き* を参照してください。

これらのパラメーターの値は、構成するデータベース・ノードの各タイプ (サーバー、クライアント、またはリモート・クライアントを持つサーバー) によって異なります。各ノード・タイプに設定できる範囲および省略時値については、*管理の手引き* を参照してください。

すべてのパラメーターを更新できるわけではありません。

データベース構成ファイルへの変更のほとんどは、ファイルがメモリーにロードされた後にのみ有効になります。これを行う前にすべてのアプリケーションはデータベースから切断されている必要があります。

エラーが発生した場合、データベース構成ファイルは変更されません。

チェックサムが無効な場合、データベース構成ファイルは更新できません。適当なコマンドを使用しないでデータベース構成ファイルを変更するとこれが発生します。これが発生する場合、データベースを復元してデータベース構成ファイルをリセットする必要があります。

以下も参照

279ページの『GET DATABASE CONFIGURATION』

531ページの『RESET DATABASE CONFIGURATION』

UPDATE DATABASE MANAGER CONFIGURATION

UPDATE DATABASE MANAGER CONFIGURATION

データベース・マネージャー構成ファイルの中の個々の項目を修正します。

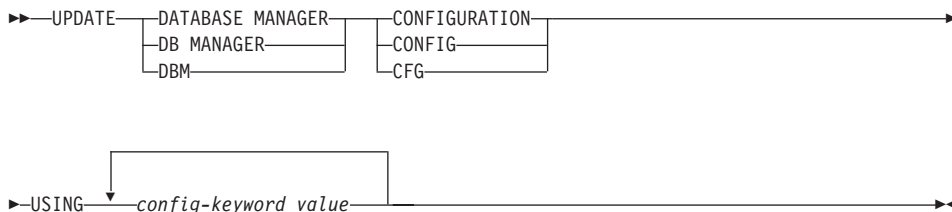
許可

sysadm

必須接続

なし、またはインスタンス。インスタンスとの接続は、ローカルの DBM 構成操作を実行する場合には必ずしも必要ではありませんが、リモートの DBM 構成操作の場合には必須です。リモート・インスタンスに対するデータベース・マネージャー構成を更新するためには、最初にそのインスタンスに接続する必要があります。

コマンド構文



コマンド・パラメーター

USING config-keyword value

更新するデータベース・マネージャー構成パラメーターを指定します。構成可能なパラメーターの要旨については、292ページの『GET DATABASE MANAGER CONFIGURATION』を参照してください。

使用上の注意

データベース・マネージャー構成パラメーターのリストの表示または印刷を行うには、292ページの『GET DATABASE MANAGER CONFIGURATION』を使用してください。

データベース・マネージャー構成パラメーターを推奨されているデータベース・マネージャーの省略時値にリセットするには、533ページの『RESET DATABASE MANAGER CONFIGURATION』を使用してください。

データベース・マネージャー構成パラメーターの詳細については、管理の手引きを参照してください。

UPDATE DATABASE MANAGER CONFIGURATION

これらのパラメーターの値は、構成するデータベース・ノードの各タイプ (サーバー、クライアント、またはリモート・クライアントを持つサーバー) によって異なります。各ノード・タイプに設定できる範囲および省略時値については、[管理の手引き](#) を参照してください。

すべてのパラメーターを更新できるわけではありません。

データベース・マネージャー構成ファイルへの変更のほとんどは、ファイルがメモリーにロードされた後にのみ有効になります。サーバー構成パラメーターの場合、これは **db2start** の実行中に起こります。クライアント構成パラメーターの場合、これはアプリケーションが再始動される時に起こります。クライアントがコマンド行プロセッサである場合は、586ページの『**TERMINATE**』を呼び出すことが必要です。

エラーが生じた場合には、データベース・マネージャー構成ファイルは変更されません。

データベース・マネージャー構成ファイルは、そのチェックサムが無効であると、更新することができません。これは、データベース・マネージャー構成ファイルが、適切なコマンドを使用せずに変更された場合に起こることがあります。これが発生する場合、データベース・マネージャーを再インストールして、データベース・マネージャー構成ファイルをリセットする必要があります。

現在のインスタンスに対する SVCENAME、NNAME、または TPNAME データベース・マネージャー構成パラメーターを更新する場合、LDAP サポートが使用可能であり、このインスタンスに対して登録されている LDAP ノードがある場合、LDAP ノードは新しい値によって更新されます。

以下も参照

292ページの『**GET DATABASE MANAGER CONFIGURATION**』

533ページの『**RESET DATABASE MANAGER CONFIGURATION**』

UPDATE LDAP NODE

UPDATE LDAP NODE

Lightweight Directory Access Protocol (LDAP) の DB2 サーバーを表すノード項目に関連したプロトコル情報を更新します。

このコマンドは、Windows NT、Windows 98、Windows 95、AIX、および Solaris でのみ使用可能です。

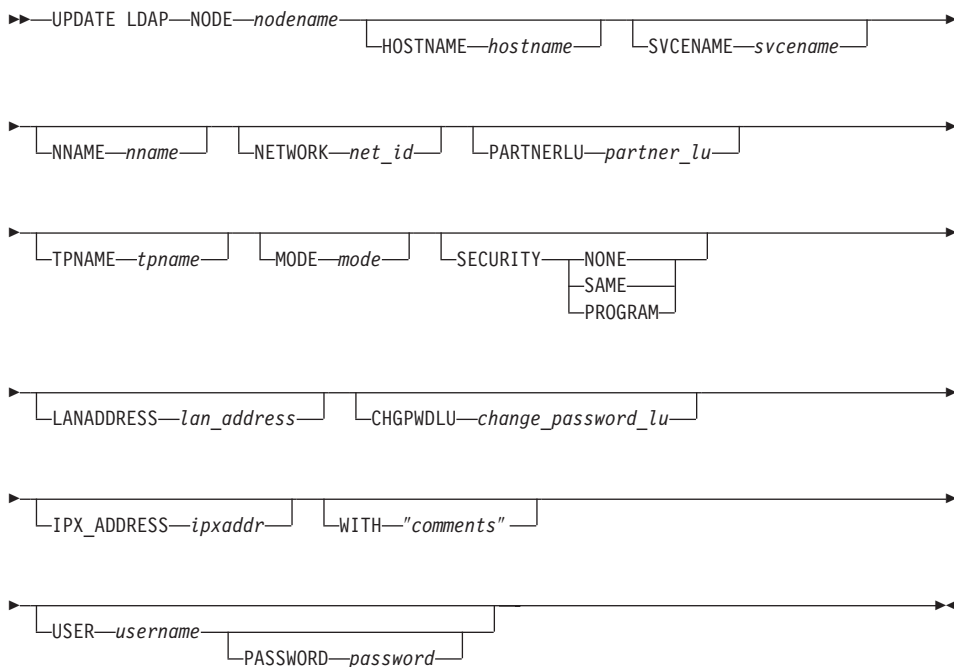
許可

なし

必須接続

なし

コマンド構文



コマンド・パラメーター

NODEnodename

リモート DB2 サーバーを更新するときのノード名を指定します。ノード名は、DB2 サーバーを LDAP に登録するときに指定した値です。

HOSTNAME hostname

TCP/IP ホスト名 (または IP アドレス) を指定します。

SVCENAME svcename

TCP/IP サービス名またはポート番号を指定します。

NNAME nname

NETBIOS ワークステーション名を指定します。

注: NETBIOS は、AIX および Solaris オペレーティング・システム上ではサポートされていません。ただし、このプロトコルは、Windows NT などのオペレーティング・システムを使用するリモート・サーバーに対しては更新できます。

NETWORK net_id

APPN ネットワーク ID を指定します。

PARTNERLU partner_lu

DB2 サーバー・マシン用の APPN パートナー LU 名を指定します。

TPNAME tpname

APPN トランザクション・プログラム名を指定します。

MODE mode

APPN モード名を指定します。

SECURITY

APPN セキュリティー・レベルを指定します。有効な値は以下のとおりです。

NONE サーバーに送信する割り振り要求に、安全保護情報が含まれないということを指定します。これは、DB2 UDB サーバーの場合の省略時セキュリティです。

SAME サーバーに送信する割り振り要求に、ユーザー名が含まれないということを指定します。これは、ユーザー名が「すでに検査済み」という標識で指定されます。サーバーは、「すでに検査済み」という保証を受け入れられるように構成されていなければなりません。

PROGRAM

サーバーに送信する割り振り要求に、ユーザー名とパスワードの両方が含まれるということを指定します。これは、DB2 (MVS 版)、DB2 (AS/400 版)、DB2 (VM 版) などのホスト・データベース・サーバーの場合の省略時セキュリティです。

UPDATE LDAP NODE

LANADDRESS lan_address

APPN ネットワーク・アダプター・アドレスを指定します。

CHGPWDLU change_password_lu

ホスト Database Server のパスワード変更時に使用される、パートナー LU の名前を指定します。

IPX_ADDRESS ipxaddr

完全な IPX アドレスを指定します。DB2 UDB がインストールされているシステムの IPX アドレスは、**db2ipxad** コマンドを呼び出せば分かります。IPX アドレスは、12 桁のネットワーク・アドレス、8 桁のノード・アドレス、4 桁のソケット番号によって、
<NetworkAddress>.<NodeAddress>.<socket> の形式で構成されます。

WITH "comments"

DB2 サーバーを記述します。ネットワーク・ディレクトリーで登録されるサーバーについての記述を補足する、任意の注釈を入力することができます。最大長は 30 文字です。復帰文字や改行文字は許可されません。注釈テキストは必ず二重引用符で囲んでください。

USER username

ユーザーの LDAP 識別名 (DN) を指定します。LDAP ユーザー DN には、LDAP ディレクトリー内でオブジェクトを作成したり更新したりするための権限が必要です。ユーザーの LDAP DN が指定されない場合、現行ログオン・ユーザーの認証が使用されます。

PASSWORD password

アカウント・パスワード。

以下も参照

64ページの『db2ipxad - IPX/SPX インターネットワーク・アドレスの獲得』

245ページの『DEREGISTER』

511ページの『REGISTER』

UPDATE MONITOR SWITCHES

1 つまたは複数のデータベース・モニター記録スイッチをオンまたはオフにします。データベース・マネージャーが開始するとき、6 個のスイッチの設定値が *dft_mon* データベース・マネージャー構成パラメーターによって判別されます (292ページの『GET DATABASE MANAGER CONFIGURATION』を参照してください)。

データベース・モニターはいつでも基本情報セットを記録します。この基本情報以上の情報を必要とするユーザーは、該当するスイッチをオンにできますが、代わりにシステム性能は低下します。312ページの『GET SNAPSHOT』から出力として利用できる情報の量は、存在するどのスイッチがオンになっているかを反映しています。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*

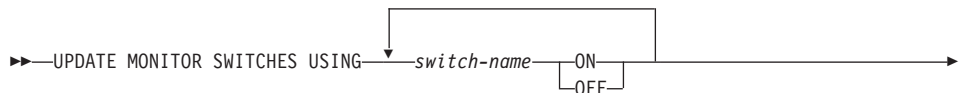
必須接続

インスタンスまたはデータベース

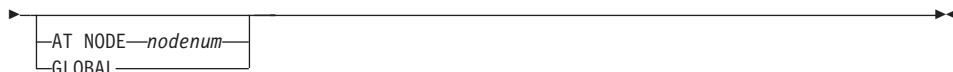
- インスタンスへの接続や、データベースへの接続がない場合、省略時のインスタンス接続が作成されます。
- インスタンスへの接続とデータベース接続の両方がある場合、インスタンス接続が使用されます。

リモート・インスタンス 1 (または異なるローカル・インスタンス) のモニター・スイッチを更新するには、最初にそのインスタンスに接続する必要があります。

コマンド構文



UPDATE MONITOR SWITCHES



コマンド・パラメーター

USING switch-name

次のスイッチ名が使用可能です。

BUFFERPOOL

バッファ・プール活動情報

LOCK ロック情報

SORT ソート情報

STATEMENT SQL ステートメント情報

TABLE 表活動情報

UOW 作業単位情報

AT NODE nodenum

モニター・スイッチの状況を表示するノードを指定します。

GLOBAL

区分データベース・システム内のすべてのノードの集合結果を戻します。結果を集める方法については、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。

使用上の注意

データベース・マネージャーが収集するのは、スイッチがオンになってからの情報だけです。**db2stop** が出されるか、UPDATE MONITOR SWITCHES コマンドを出したアプリケーションが終了するまで、スイッチは設定されたままです。特定のスイッチに関連した情報をクリアするには、まずスイッチをオフに設定し、それからオンにしてください。

あるアプリケーションでスイッチを更新しても、他のアプリケーションには影響がありません。

スイッチ設定値を表示するには、310ページの『GET MONITOR SWITCHES』を使用してください。

UPDATE HISTORY FILE

活動記録ファイル項目にあるロケーション、装置タイプ、または注釈を更新します。

許可

以下のどれかが必要です。

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

必須接続

データベース

コマンド構文

```

▶▶—UPDATE HISTORY FOR—object-part—WITH—————▶
|
|—LOCATION—new-location—DEVICE TYPE—new-device-type————▶
|—COMMENT—new-comment————▶

```

コマンド・パラメーター

FOR *object-part*

イメージのバックアップまたはコピーの識別子を指定します。この識別子は、タイム・スタンプと 001 ~ 999 の任意選択の順序番号で構成されます。

LOCATION *new-location*

バックアップの新しい物理ロケーションを指定します。このパラメーターの解釈は装置タイプに依存します。

DEVICE TYPE *new-device-type*

バックアップを保管する新しい装置タイプを指定します。有効な装置タイプは次のとおりです。

- | | |
|----------|--------|
| D | ディスク |
| K | ディスケット |
| T | テープ |

UPDATE HISTORY FILE

A	TSM
U	ユーザー出口
O	その他

COMMENT new-comment

項目を記述する新しい注釈を指定します。

例

1997 年 4 月 13 日午前 10 時 00 分にとった全データベース・バックアップの活動記録ファイルを更新するには、次のように入力します。

```
db2 update history for 19970413100000001 with
location /backup/dbbackup.1 device type d
```

使用上の注意

活動記録ファイルは記録を保存する目的でのみ使用されます。データベース回復中は使用されません。

以下も参照

487ページの『PRUNE HISTORY/LOGFILE』

第4章 コマンド行構造化照会言語ステートメントの使用

この節では、コマンド行から SQL (構造化照会言語) ステートメントを使用する方法について説明します。これらのステートメントは、オペレーティング・システムのコマンド・プロンプトから直接実行できるもので、コマンドをアプリケーション・プログラムに書き込む場合とほぼ同じようにして、データベースの表や索引や視点に格納されている情報を定義したり実行したりするのに使用できます。情報については追加や削除や更新ができます。レポートは表の内容から生成できます。

コマンド行プロセッサによって実行できる SQL ステートメントはすべて、622ページの表8 の CLP 列にリストされています。SQL ステートメントの構文は、コマンド行から実行できるものであれ、ソース・プログラムに組み込まれているものであれ、すべてが *SQL 解説書* に記述されています。多くの場合、組み込み SQL ステートメントと CLP SQL ステートメントの構文は同じです。ただし、ホスト変数、パラメーター・マーカー、記述子名、およびステートメント名は、組み込み SQL にのみ適用できます。

CLOSE、CONNECT、DECLARE CURSOR、FETCH、OPEN、および SELECT の構文は、組み込み型のもものと CLP によって実行されるものとは異なります。これらのステートメントの CLP 構文を次に示します。

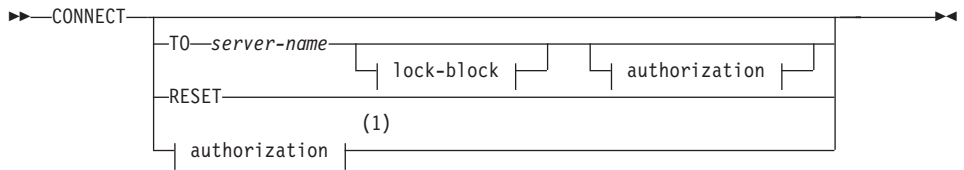
CALL

```
▶▶—CALL—SQLJ.INSTALL_JAR—(—jar-url, jar-id—)—————▶▶
          |
          |—SQLJ.REPLACE_JAR—(—jar-url, jar-id—)—————▶▶
          |
          |—SQLJ.REMOVE_JAR—(—jar-id—)—————▶▶
          |
          |—SQLJ.REFRESH_CLASSES—(—void—)—————▶▶
```

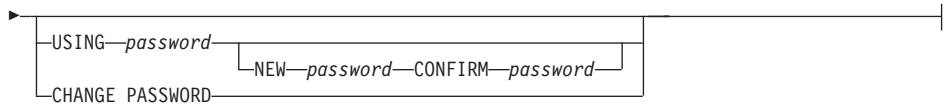
CLOSE

```
▶▶—CLOSE—cursor-name—————▶▶
```

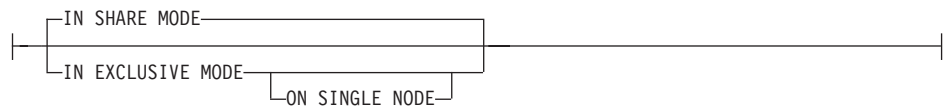
CONNECT



authorization:



lock-block:



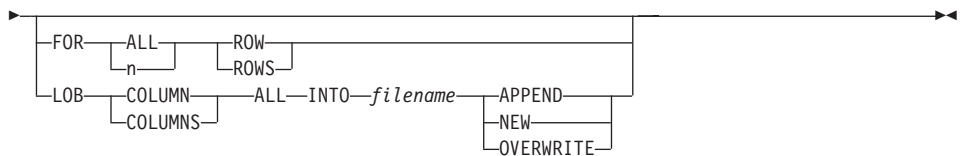
注:

- 1 この形式は、暗黙接続が使用可能である場合にのみ有効です。

DECLARE CURSOR



FETCH

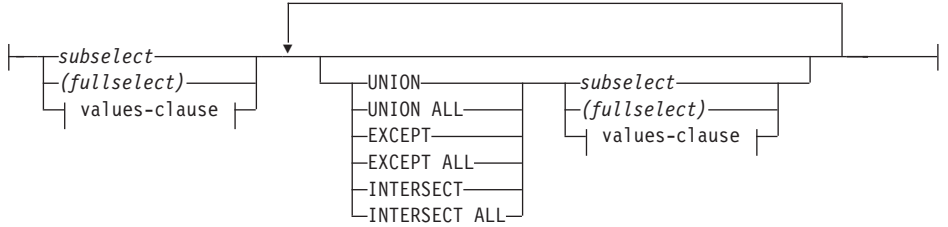


OPEN

▶—OPEN—*cursor-name*—▶

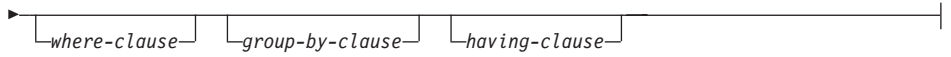
SELECT

fullselect:

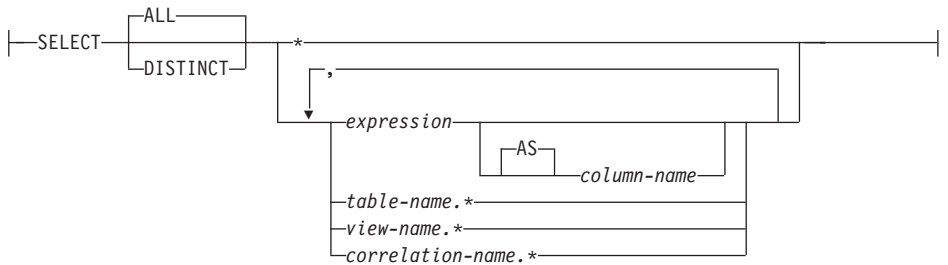


subselect:

|—*select-clause*—*from-clause*—▶



select-clause:



values-clause:



row-expression:



注:

1. コマンド行プロセッサから CALL を発行する場合は、明示されたプロシージャとそれらに伴う個別パラメーターだけがサポートされます。

jar-url インストールまたは置換する jar ファイルが含まれている URL を指定します。サポートされている URL スキームは、file: だけです。

jar-id jar-url に指定されたファイルに関連付ける、データベースの jar 識別子を指定します。

たとえば、次のようにします。

```
db2 call sqlj.install_jar ( "file:///C:/java/jarfiles/myprocs.zip", "myprocs" )
```

SQLJ.REFRESH_CLASSES

DB2 が、インスタンスを再始動せずに、更新されたストアード・プロシージャ Java クラスを使用するようにします。ストアード・プロシージャ Java クラスを更新した後にこのステートメントが呼びだされない場合、DB2 は以前のバージョンのストアード・プロシージャ・クラスの使用を継続します。

2. CONNECT の CLP バージョンを使用すると、ユーザーは、次のパラメーターを使用してパスワードを変更することができます。

NEW password

ユーザー名に割り当てられる新規パスワードを指定します。パスワードの長さは、最大で 18 文字です。パスワードが変更されるシステムは、ユーザー認証がセットアップされた方法によって異なります。

CONFIRM password

新規パスワードと同一のストリング。このパラメーターは、入力エラーを検出するために使用されます。

CHANGE PASSWORD

このオプションが指定されていると、ユーザーにプロンプトが出され、現在のパスワード、新規パスワード、および新規パスワードの確認を要求します。入力時にパスワードは表示されません。

3. FETCH または SELECT がコマンド行プロセッサから出されると、10 進数および浮動小数点が、各国の 10 進数区切り文字と共に表示されます。米国、カナダ、英国の場合はピリオド (.)、他のほとんどの国の場合はコンマ (,) です。ただし、INSERT、UPDATE、およびその他の SQL ステートメントをコマンド行プロセッサから出して表を更新する場合は、10 進数区切り文字としてピリオドを使用しなければなりません。
4. FETCH または SELECT がコマンド行プロセッサから出されると、通常ヌル値はハイフン (-) で表示されます。DFT_SQLMATHWARN YES で構成されたデータベースでは、算術計算エラーとなる式はヌル値として処理されます。そのような算術計算エラー・ヌル値は、プラス (+) で表示されま

たとえば、表 t1 を次のように作成します。

```
create table t1 (i1 int , i2 int);
insert into t1 values (1,1),(2,0),(3,null);
```

ステートメント select i1/i2 from t1 は、次の結果を生成します。

```
1
---
1
+
-
3 records selected
```

5. 新しい LOB オプションが FETCH に追加されています。LOB 文節を指定する場合は、次の行だけが取り出されます。
 - それぞれの LOB 列値は filename.xxx という名前のファイルに取り出されます。この場合、filename は LOB 文節に指定され、xxx は 001 から 999 までのファイル拡張子です。(001 は対応する DECLARE CURSOR ステートメントの選択リストの最初の LOB 列で、002 は、2 番目の LOB 列、以降 999 は 999 番目の列になります。) ファイルに取り出せる LOB 列の最高数は 999 です。
 - データを含むファイルの名前は LOB 列に表示されます。
6. LOB 列を含む表を照会するために SELECT がコマンド行プロセッサを通じて発行される場合、各 LOB 列は出力で 4KB に切り捨てられます。
7. コマンド行プロセッサは、BLOB 列を 16 進表記で表示します。
8. 適切な変換関数を使用できない場合には、構造型列への参照を含む SQL ステートメントを発行することはできません。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。

CLP を介する SQL ステートメントを使用してデータベースを照会する場合、CLP がデータを表示する仕方を変更することができます。このことは、CLP バインド・ファイルを照会するデータベースに対して再バインドすることにより行います。たとえば、日時を ISO 形式で表示したい場合、次のようにできます。

1. CLP バインド・ファイルの名前を含むテキスト・ファイルを作成する。このファイルは、1 回の BIND コマンドで複数のファイルをバインドする場合のリスト・ファイルとして使用します。この例では、とりあえずこのファイルの名前を `clp.lst` としておきます。このファイルの中身は次のようになっています。

```
db2clpcs.bnd +
db2clpr.r.bnd +
db2clpur.bnd +
db2clprs.bnd +
db2clpns.bnd
```

2. データベースに接続する。
3. 次のコマンドを発行する。

```
db2 bind @clp.lst collection nullid datetime iso
```

コマンド行プロセッサの詳細については、121ページの『第2章 コマンド行プロセッサ (CLP)』を参照してください。SQL ステートメントの構文およびSQL ステートメントの関数の詳細については、SQL 解説書を参照してください。構文図の読み方については、627ページの『付録A. 構文図の読み方』を参照してください。

表 8. SQL ステートメント (DB2 ユニバーサル・データベース)

SQL ステートメント	動的 ¹	コマンド 行プロセ ッサ (CLP)	コール・レベル・ インターフェース ³ (CLI)	SQL プロ シージャ ー
ALLOCATE CURSOR				X
割り当てステートメント				X
ASSOCIATE LOCATORS				X
ALTER { BUFFERPOOL, NICKNAME, ¹⁰ NODEGROUP, SERVER, ¹⁰ TABLE, TABLESPACE, USER MAPPING, ¹⁰ TYPE, VIEW }	X	X	X	
BEGIN DECLARE SECTION ²				
CALL		X ⁹	X ⁴	X

表 8. SQL ステートメント (DB2 ユニバーサル・データベース) (続き)

SQL ステートメント	動的 ¹	コマンド 行プロセ ッサー (CLP)	コール・レベル・ インターフェース ³ (CLI)	SQL プロ シージャ ー
CASE ステートメント				X
CLOSE		X	SQLCloseCursor(), SQLFreeStmt()	X
COMMENT ON	X	X	X	X
COMMIT	X	X	SQLEndTran、SQLTransact()	X
複合 SQL (組み込み)			X ⁴	
複合ステートメント				X
CONNECT (タイプ 1)		X	SQLBrowseConnect(), SQLConnect(), SQLDriverConnect()	
CONNECT (タイプ 2)		X	SQLBrowseConnect(), SQLConnect(), SQLDriverConnect()	
CREATE { ALIAS, BUFFERPOOL, X DISTINCT TYPE, EVENT MONITOR, FUNCTION, FUNCTION MAPPING, ¹⁰ INDEX, INDEX EXTENSION, METHOD, NICKNAME, ¹⁰ NODEGROUP, PROCEDURE, SCHEMA, TABLE, TABLESPACE, TRANSFORM TYPE MAPPING, ¹ TRIGGER, USER MAPPING, ¹⁰ TYPE, VIEW, WRAPPER ¹⁰ }	X	X	X	X ¹¹
DECLARE CURSOR ²		X	SQLAllocStmt()	X
DECLARE GLOBAL X TEMPORARY TABLE	X	X	X	X
DELETE	X	X	X	X
DESCRIBE ⁸		X	SQLColAttributes(), SQLDescribeCol(), SQLDescribeParam() ⁶	
DISCONNECT		X	SQLDisconnect()	
DROP	X	X	X	X ¹¹

表 8. SQL ステートメント (DB2 ユニバーサル・データベース) (続き)

SQL ステートメント	動的 ¹	コマンド 行プロセ ッサー (CLP)	コール・レベル・ インターフェース ³ (CLI)	SQL プロ シージャ ー
END DECLARE SECTION ²				
EXECUTE			SQLExecute()	X
EXECUTE IMMEDIATE			SQLExecDirect()	X
EXPLAIN	X	X	X	X
FETCH		X	SQLExtendedFetch() ⁷ 、 SQLFetch()、SQLFetchScroll() ⁷	X
FLUSH EVENT MONITOR	X	X	X	
FOR ステートメント				X
FREE LOCATOR			X ⁴	X
GET DIAGNOSTICS				X
GOTO ステートメント				X
GRANT	X	X	X	X
IF ステートメント				X
INCLUDE ²				
INSERT	X	X	X	X
ITERATE				X
LEAVE ステートメント				X
LOCK TABLE	X	X	X	X
LOOP ステートメント				X
OPEN		X	SQLExecute()、SQLExecDirect()	X
PREPARE			SQLPrepare()	X
REFRESH TABLE	X	X	X	
RELEASE		X		X
RELEASE SAVEPOINT	X	X	X	X
RENAME TABLE	X	X	X	
RENAME TABLESPACE	X	X	X	
REPEAT ステートメント				X
RESIGNAL ステートメント				X
RETURN ステートメント				X
REVOKE	X	X	X	

表 8. SQL ステートメント (DB2 ユニバーサル・データベース) (続き)

SQL ステートメント	動的 ¹	コマンド 行プロセ ッサー (CLP)	コール・レベル・ インターフェース ³ (CLI)	SQL プロ シージャ ー
ROLLBACK	X	X	SQLEndTran()、SQLTransact()	X
SAVEPOINT	X	X	X	X
select-statement	X	X	X	X
SELECT INTO				X
SET CONNECTION		X	SQLSetConnection()	
SET CURRENT DEFAULT TRANSFORM GROUP	X	X	X	X
SET CURRENT DEGREE	X	X	X	X
SET CURRENT EXPLAIN MODE	X	X	X、SQLSetConnectAttr()	X
SET CURRENT EXPLAIN SNAPSHOT	X	X	X、SQLSetConnectAttr()	X
SET CURRENT PACKAGESET				
SET CURRENT QUERY OPTIMIZATION	X	X	X	X
SET CURRENT REFRESH AGE	X	X	X	X
SET EVENT MONITOR STATE	X	X	X	X
SET INTEGRITY	X	X	X	
SET PASSTHRU ¹⁰	X	X	X	X
SET PATH	X	X	X	X
SET SCHEMA	X	X	X	X
SET SERVER OPTION ¹⁰	X	X	X	X
SET transition-variable ⁵	X	X	X	X
SIGNAL ステートメント				X
SIGNAL SQLSTATE ⁵	X	X	X	
UPDATE	X	X	X	X
VALUES INTO				X
WHENEVER ²				
WHILE ステートメント				X

表 8. SQL ステートメント (DB2 ユニバーサル・データベース) (続き)

SQL ステートメント	動的 ¹	コマンド コール・レベル・ 行プロセ インターフェース ³ (CLI) ッサー (CLP)	SQL プロ シージャ ー
-------------	-----------------	---	---------------------

注:

1. このリストのすべてのステートメントは静的 SQL としてコーディングできますが、動的 SQL としてコーディングできるのは X になっているステートメントだけです。
2. このステートメントは実行できません。
3. X は、該当するステートメントが SQLExecDirect() または SQLPrepare() と SQLExecute() のどちらによっても実行できるという意味です。同等の DB2 CLI 機能がある場合は、機能名がリストされています。
4. このステートメントは動的ではないものの、DB2 CLI によって SQLExecDirect() または SQLPrepare() と SQLExecute() のどちらかを呼び出すときにステートメントは指定されます。
5. CREATE TRIGGER ステートメント内だけで使用できます。
6. SQL DESCRIBE ステートメントで出力の記述ができますが、DB2 CLI を使用すると、入力の記述も可能となります (SQLDescribeParam() 関数を使用する場合)。
7. SQL FETCH ステートメントで 1 つの行を一度に 1 方向に取り出すことができますが、DB2 CLI の SQLExtendedFetch() および SQLFetchScroll() を使用すると、配列の形で取り出すことができます。さらに、どの方向でも、また結果セットのどの位置でも取り出しができます。
8. DESCRIBE SQL ステートメントの構文は、CLP DESCRIBE コマンドの構文と異なります。DESCRIBE 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。DESCRIBE CLP コマンドについては、コマンド解説書を参照してください。
9. コマンド行プロセッサから CALL を発行する場合は、特定のプロシージャとそれらに伴う個別パラメーターだけがサポートされます (617 ページを参照)。
10. ステートメントは、連合データベース・サーバーでのみサポートされます。
11. SQL プロシージャは、索引、表、視点には CREATE および DROP ステートメントしか発行できません。

付録A. 構文図の読み方

構文図では、オペレーティング・システムが入力を正しく判別できるようなコマンドの指定方法を示します。

構文図は、左から右、上から下に、横線 (メイン・パス) に沿って読んでください。行が矢印で終わっている場合は、コマンド構文が続くことを示しており、次の行が矢印で始まります。垂直線はコマンド構文の終わりを示します。

構文図からの情報を入力する時は、引用符や等号などの記号類を必ず含めてください。

パラメーターは、キーワードと変数に分類されます。

- キーワードは定数を表し、英大文字です。しかし、コマンド・プロンプトでは、大文字でも、小文字でも、大文字小文字の混合でも構いません。コマンド名はキーワードの一例です。
- 変数はユーザーが提供した名前や値を表し、英小文字です。しかし、コマンド・プロンプトでは、文字の種類がはっきり指定されている場合以外は、大文字、小文字、大文字小文字の混合のどれで入力しても構いません。ファイル名は変数の一例です。

パラメーターはキーワードと変数の組み合わせにもなります。

必須パラメーターはメイン・パスに表示されます。



オプション・パラメーターはメイン・パスの下に表示されます。

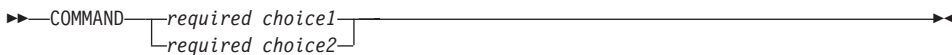


パラメーターの省略時の値はパスの上に表示されます。

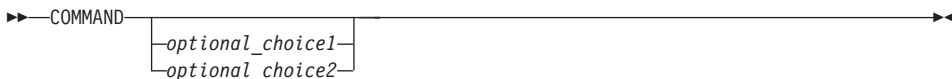


構文図の読み方

最初のパラメーターだけがメイン・パスに表示されているパラメーター・スタックの場合は、一つのパラメーターを選択しなければなりません。

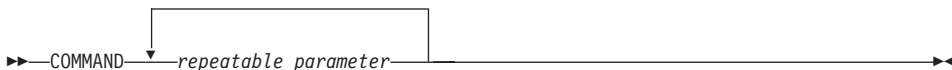


最初のパラメーターだけがメイン・パスの下に表示されているパラメーター・スタックの場合は、一つのパラメーターを選択できます。



パスの上に左向きの矢印がある場合は、次の規則に従って項目を繰り返すことができます。

- 矢印が中断されていない場合は、項目をブランクで区切って並べたリストの中で項目を繰り返すことができます。

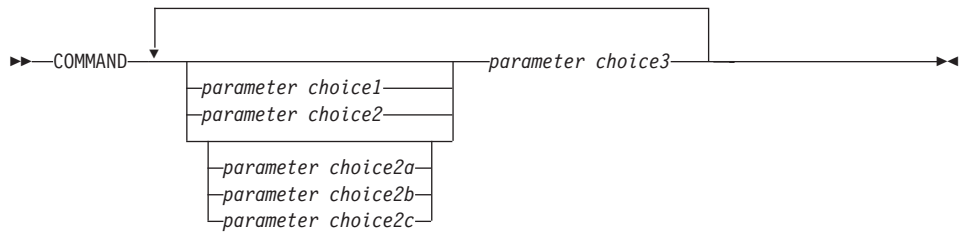


- 矢印にコンマが含まれている場合は、項目をコンマで区切って並べたリストの中で項目を繰り返すことができます。



パラメーター・スタックの項目は、前に取り上げた必須パラメーターとオプション・パラメーターのスタック規則に従って繰り返すことができます。

構文図の中には、他のパラメーター・スタックの中に、さらにパラメーター・スタックを含むものがあります。スタックの項目を繰り返す場合は、前に取り上げた規則に必ず従わなければなりません。つまり、繰り返し矢印が内部スタックの上になく、外部スタックの上にある場合は、内部スタックからパラメーターを一つだけ選択し、外部スタックの任意のパラメーターと組み合わせた上で、その組み合わせを繰り返すことができます。たとえば、次の図では、パラメーター *choice2a* とパラメーター *choice2* を組み合わせ、その組み合わせ (*choice2* と *choice2a*) を繰り返すことができます。



一部のコマンドの前には、オプションのパス・パラメーターが付いています。



このパラメーターがない場合、システムは現行ディレクトリーでコマンドを探します。コマンドが見つからない場合、システムは `.profile` にリストされているパスの上のすべてのディレクトリーでコマンドを探し続けます。

一部のコマンドには、機能が同等の構文変数があります。



構文図の読み方

付録B. 命名規則

データベースや表、認証 ID などのデータベース・マネージャー・オブジェクトの命名の際に適用される規則について説明します。

- データベース・マネージャー・オブジェクトの名前を表す文字ストリングには、a～z、A～Z、0～9、@、#、および \$ などが含まれます。
- ストリングの最初の文字はアルファベット、@、#、または \$ にしなければなりません。数字や文字順序列の SYS、DBM、IBM などは不可です。
- 特に注記のない限り、名前は小文字で入力して構いません。ただし、データベース・マネージャーはそれらを大文字とみなして処理します。

ただし、システム・ネットワーク体系 (SNA) 下の名前を表す文字ストリングは例外です。論理装置名 (partner_lu および local_lu) など、値の多くは大文字小文字を区別します。こうした名前は、それらの用語に対応する SNA 定義に出ているとおりに入力してください。

- データベース名やデータベース別名は、前に説明した集合内の 1 つから 8 つの文字、数字、キーボード文字を含む固有の文字ストリングです。

データベースはシステム内にカタログ作成されており、ローカル・データベース・ディレクトリーの別名が一つのフィールドに、元名が別のフィールドに入っています。ほとんどの機能の場合、データベース・マネージャーは、データベース・ディレクトリーの別名フィールドに入力された名前を使用します。(ただし、CHANGE DATABASE COMMENT および CREATE DATABASE は例外です。この場合は、ディレクトリー・パスを指定しなければなりません。)

- 表や視点の名前または別名は、1～128 文字の固有な文字ストリングで構成される SQL 識別子です。列名の長さは 1～30 文字です。

完全修飾された表名は、*schema.tablename* から成っています。スキーマ (schema) は固有のユーザー ID で、その下に表が作成されます。宣言一時表のスキーマ名は SESSION でなければなりません。

- 認証 ID の長さは、Windows 32 ビット オペレーティング・システムでは 30 文字、その他のオペレーティング・システムでは 8 文字を超えてはなりません。
- グループ ID の長さは、8 文字を超えてはなりません。
- ノード・ディレクトリーでカタログ化されるリモート・ノードのローカル別名の長さは、8 文字を超えてはなりません。

命名規則の詳細については、[管理の手引き](#) を参照してください。すべての DB2 識別子の長さ制限の詳細については、[SQL 解説書](#) を参照してください。

付録C. DB2 ライブラリーの使用法

DB2 ユニバーサル・データベース ライブラリーは、オンライン・ヘルプ、ブック (PDF および HTML)、および HTML 形式のサンプル・プログラムから成っています。このセクションでは、ユーザーに提供される情報について紹介し、その入手方法を示します。

オンライン製品情報をご利用になるには、インフォメーション・センターを使用することができます。詳細については、649ページの『インフォメーション・センターを使用した情報へのアクセス』を参照してください。ここではタスク情報、DB2 ブック、トラブルシューティング情報、サンプル・プログラム、および Web の DB2 情報を見ることができます。

DB2 PDF ファイルおよびハードコピー版資料

DB2 情報

以下に示す表では、DB2 ブックを 4 つのカテゴリーに分類しています。

DB2 の手引きおよび解説書

これらの資料は、すべてのプラットフォームに共通の DB2 情報を含んでいます。

DB2 のインストールおよび構成の情報

これらの資料は、特定のプラットフォーム上の DB2 ごとに用意されています。たとえば、OS/2、Windows、および UNIX ベースのプラットフォームで稼働するそれぞれの DB2 用に、別個の概説およびインストール 資料が用意されています。

プラットフォーム共通のサンプル・プログラム (HTML 形式)

これらのサンプルは、アプリケーション開発クライアントとともにインストールされるサンプル・プログラムの HTML 版です。これらのサンプルは参考用であり、実際のプログラムに代わるものではありません。

リリース情報

これらのファイルには、DB2 ブックには含まれなかった最新の情報が記載されています。

インストール情報、リリース情報、およびチュートリアルは、製品 CD-ROM から HTML 形式で参照することができます。ほとんどの資料は、製品

CD-ROM から HTML 形式で表示できますし、DB2 の資料 CD-ROM から Adobe Acrobat (PDF) 形式で表示し印刷することができます。IBM にハードコピー版の資料を注文したい場合は、645ページの『印刷資料の注文方法』を参照してください。注文可能な資料については、以下の表をご覧ください。

OS/2 および Windows プラットフォームの場合、HTML ファイルは `sqllib¥doc¥html` ディレクトリーにインストールできます。DB2 情報はいくつかの言語で提供されています。しかし、すべての言語に翻訳されているわけではありません。ある言語で情報が提供されていない場合は、英語版の情報が提供されます。

UNIX プラットフォームの場合、言語ごとに異なる複数の HTML ファイルを `doc/%L/html` ディレクトリーにインストールできます。ここで、`%L` は地域を表しています。詳細については、適切な「概説およびインストールの手引き」を参照してください。

DB2 ブックを入手して情報を利用するには、次のようなさまざまな方法があります。

- 648ページの『オンライン情報の表示』
- 653ページの『オンライン情報の検索』
- 645ページの『印刷資料の注文方法』
- 645ページの『PDF 資料の印刷』

表9. DB2 情報

資料名	説明	資料番号 PDF ファイル名	HTML ディレクトリー
DB2 の手引きおよび解説書情報			
管理の手引き	管理の手引き: 計画 は、データベース概念について概説し、設計 (たとえば、論理および物理データベース設計) に関する情報を提供し、高い可用性について解説しています。	第 1 巻 SC88-8513 db2d1x70	db2d0
	管理の手引き: インプリメンテーション は、設計、データベースへのアクセス、監査、バックアップ、および回復などのインプリメンテーションについて説明しています。	第 2 巻 SC88-8511 db2d2x70	
	管理の手引き: パフォーマンス は、データベース環境について解説し、さらにアプリケーションのパフォーマンスの評価と調整の方法について説明しています。	第 3 巻 SC88-8512 db2d3x70	
管理 API 解説書	データベースの管理に使用できる DB2 アプリケーション・プログラミング・インターフェース (API) およびデータ構造について説明します。また、この資料は、アプリケーションから API を呼び出す方法も示します。	SC88-8514 db2b0x70	db2b0
アプリケーション構築の手引き	環境設定に関する情報を提供し、Windows、OS/2、および UNIX ベースのプラットフォームでの DB2 アプリケーションのコンパイル、リンク、実行の各ステップについて説明します。	SC88-8515 db2axx70	db2ax
APPC, CPI-C, and SNA Sense Codes	DB2 ユニバーサル・データベース製品をご使用中に発生する可能性のあるセンス・コード APPC、CPI-C、および SNA についての一般情報を提供します。 HTML 形式でのみご利用いただけます。	資料番号なし db2apx70	db2ap

表9. DB2 情報 (続き)

資料名	説明	資料番号	HTML
		PDF ファイル名	ディレクトリー
アプリケーション開発の手引き	DB2 データベースにアクセスするアプリケーションを、組み込み SQL または Java (JDBC および SQLJ) を使用して開発する方法について説明します。さらに、ストアド・プロシージャの作成方法、ユーザー定義関数の作成方法、ユーザー定義タイプの作成方法、トリガーの使用法、区画化されている環境または統合されているシステムでのアプリケーションの開発方法などについて解説されています。	SC88-8516	db2a0
		db2a0x70	
コール・レベル・インターフェースの手引きおよび解説書	DB2 データベースにアクセスするアプリケーションを、DB2 コール・レベル・インターフェース (Microsoft ODBC 仕様互換の呼び出し可能 SQL) を使用して開発する方法について説明します。	SC88-8517	db2l0
		db2l0x70	
コマンド解説書	コマンド行プロセッサの使用法について説明し、データベースの管理に使用できる DB2 コマンドについて解説しています。	SC88-8518	db2n0
		db2n0x70	
コネクティビティー 補足	DB2 (AS/400 版)、DB2 (OS/390 版)、DB2 (MVS 版)、または DB2 (VM 版) を DRDA アプリケーション・リクエスターとして DB2 ユニバーサル・データベースとともに使用するためのセットアップ情報および参照情報を提供します。また、この資料は DRDA アプリケーション・サーバーを DB2 コネクト アプリケーション・リクエスターとともに使用する方法の詳細を示します。	資料番号なし	db2h1
		db2h1x70	
	HTML と PDF でのみ利用可能		
データ移動ユーティリティー 手引きおよび解説書	データの移動を行う DB2 ユーティリティー (インポート、エクスポート、ロード、AutoLoader、および DPROF など) の使用法について説明しています。	SC88-8522	db2dm
		db2dmx70	

表9. DB2 情報 (続き)

資料名	説明	資料番号	HTML
		PDF ファイル名	ディレクトリー
データウェアハウスセンター 管理の手引き	データウェアハウスセンターを使用してデータウェアハウスを構築および保守する方法を説明します。	SC88-8545 db2ddx70	db2dd
データウェアハウスセンター アプリケーション統合の手引き	プログラマーがアプリケーションをデータウェアハウスセンターおよび情報カタログ・マネージャーと統合するのに役立つ情報を提供します。	SC88-8546 db2adx70	db2ad
DB2 コネクト 使用者の手引き	DB2 コネクト製品の概念、プログラミング、および一般的な使用方法に関する情報を提供します。	SC88-8521 db2c0x70	db2c0
DB2 クエリー・パトローラー 管理の手引き	DB2 クエリー・パトローラー・システムの運用の概説を行い、運用および管理に関する詳細情報、および管理用グラフィカル・ユーザー・インターフェース・ユーティリティについてのタスク情報を提供します。	SC88-8525 db2dwx70	db2dw
DB2 クエリー・パトローラー 使用者の手引き	DB2 クエリー・パトローラーのツールや関数の使用方法を説明します。	SC88-8527 db2wwx70	db2ww
用語集	DB2 およびその構成要素で使用される用語の定義を示します。 HTML 形式と SQL 解説書 で利用可能	資料番号なし db2t0x70	db2t0
イメージ、オーディオ、およびビデオ・エクステンダー 管理およびプログラミングの手引き	DB2 エクステンダーの一般情報について提供し、画像、音声、およびビデオ (IAV) エクステンダーの管理と構成について、および IAV エクステンダーを使用したプログラミングについて説明しています。さらに、参照情報、診断情報 (メッセージ解説)、およびサンプルも収録されています。	SC88-8609 dmbu7x70	dmbu7
情報カタログ・マネージャー 管理の手引き	情報カタログを管理するためのガイドです。	SC88-8547 db2dix70	db2di
情報カタログ・マネージャー プログラミングの手引きおよび解説書	情報カタログ・マネージャー用の体系化されたインターフェースの定義を示します。	SC88-8549 db2bix70	db2bi

表9. DB2 情報 (続き)

資料名	説明	資料番号 PDF ファイル名	HTML ディレクトリー
情報カタログ・マネージャー 使用者の手引き	情報カタログ・マネージャー・ユーザー・インターフェースの使用に関する情報を提供します。	SC88-8548 db2aix70	db2ai
インストールおよび構成補足	プラットフォーム固有の DB2 クライアントの計画、インストール、およびセットアップのガイドです。この補足資料には、バインド、クライアント / サーバー通信の設定、DB2 GUI ツール、DRDA AS、分散インストール、分散要求の構成、および異種データ・ソースへのアクセスについても説明されています。	GC88-8524 db2iyx70	db2iy
メッセージ解説書	DB2、情報カタログ・マネージャー、およびデータウェアハウスセンターから出されるメッセージとコードをリストし、取るべき処置を解説しています。	第 1 巻 GC88-8543 db2m1x70 第 2 巻 GC88-8544 db2m2x70	db2m0
<i>OLAP Integration Server Administration Guide</i>	<i>OLAP Integration Server の Administration Manager</i> 構成要素の使用方法を説明します。	SC27-0782 db2dpx70	n/a
<i>OLAP Integration Server Metaoutline User's Guide</i>	標準の OLAP Metaoutline インターフェースを使用して (Metaoutline Assistant を使用するのではなく) OLAP metaoutline を作成しデータを取り込む方法を説明しています。	SC27-0784 db2upx70	n/a
<i>OLAP Integration Server Model User's Guide</i>	(Model Assistant ではなく) 標準的な OLAP Model Interface を使用して OLAP モデルを作成する方法を説明します。	SC27-0783 db2lpx70	n/a
<i>OLAP Setup and User's Guide</i>	OLAP Starter Kit の構成およびセットアップに関する情報を提供します。	SC27-0702 db2ipx70	db2ip
<i>OLAP Spreadsheet Add-in User's Guide for Excel</i>	Excel 作表計算プログラムを使用して OLAP データを分析する方法を説明します。	SC27-0786 db2epx70	db2ep

表9. DB2 情報 (続き)

資料名	説明	資料番号	HTML
		PDF ファイル名	ディレクトリー
<i>OLAP Spreadsheet Add-in User's Guide for Lotus 1-2-3</i>	ロータス 1-2-3 作表計算プログラムを使用して OLAP データを分析する方法を説明します。	SC27-0785 db2tpx70	db2tp
レプリケーションの手引きおよび解説書	DB2 に付属の IBM レプリケーション・ツールの計画、構成、管理、および使用方法に関する情報を提供します。	SC88-8550 db2e0x70	db2e0
地理情報エクステンダー使用者の手引きおよび解説書	地理情報エクステンダーのインストール、構成、管理、プログラミング、およびトラブルシューティングに関する情報を提供します。また、地理情報データの概念についての重要事項を示し、地理情報エクステンダー固有の参照情報 (メッセージおよび SQL) を提供します。	SC88-8624 db2sbx70	db2sb
SQL 概説	SQL の概念を紹介し、構造体とタスクの例を多数提供しています。	SC88-8539 db2y0x70	db2y0
SQL 解説書	SQL の構文、セマンティクス、および言語規則について説明します。また、この資料には、各リリース間の互換性、製品の制限事項、およびカタログ・ビューも含まれます。	第 1 巻 SC88-8540 db2s1x70 第 2 巻 SC88-8657 db2s2x70	db2s0
システム・モニター 手引きおよび解説書	データベースおよびデータベース・マネージャーに関連したさまざまな情報を収集する方法を示します。この資料は、この情報を利用して、データベース活動の把握、パフォーマンス向上、および問題原因の判別を行う方法を説明しています。	SC88-8523 db2f0x70	db2f0

表9. DB2 情報 (続き)

資料名	説明	資料番号 PDF ファイル名	HTML ディレクトリー
テキスト・エクステンダー管理およびプログラミング	DB2 エクステンダーの一般情報、テキスト・エクステンダーの管理および構成情報、およびテキスト・エクステンダーを使用したプログラミングの方法について解説します。この資料には、参照情報、診断情報 (メッセージ解説)、およびサンプルが含まれています。	SC88-8610 desu9x70	desu9
問題判別の手引き	エラーの原因の判別、問題からの回復、および DB2 カスタマー・サービスの支援の下での診断ツールの使用法を記載しています。	GD88-7271 db2p0x70	db2p0
新機能	DB2 ユニバーサル・データベースバージョン 7 の新しい機能および拡張機能について説明します。	SC88-8541 db2q0x70	db2q0
DB2 のインストールおよび構成の情報			
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	OS/2 および Windows 32 ビット オペレーティング・システム版の DB2 コネクト エンタープライズ・エディションで、計画、移行、インストール、および構成を行う場合の情報を提供します。また、この資料はサポートされている多数のクライアントのインストールおよびセットアップについても説明します。	GC88-8520 db2c6x70	db2c6
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	UNIX ベースのプラットフォームでの DB2 コネクト エンタープライズ・エディションの計画、移行、インストール、構成、およびタスクに関する情報を提供します。また、この資料はサポートされている多数のクライアントのインストールおよびセットアップについても説明します。	GC88-8519 db2cyx70	db2cy

表9. DB2 情報 (続き)

資料名	説明	資料番号	HTML
		PDF ファイル名	ディレクトリー
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	OS/2 および Windows 32 ビット オペレーティング・システムの DB2 コネクト パーソナル・エディションで、計画、移行、インストール、および構成を行う場合のタスク情報を提供します。また、この資料はサポートされているすべてのクライアントのインストールおよびセットアップについても説明します。	GC88-8533	db2c1
		db2c1x70	
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	サポートされる Linux 配布プログラムの DB2 コネクト パーソナル・エディションで、計画、インストール、移行、および構成を行う場合の情報を提供します。	GC88-8528	db2c4
		db2c4x70	
DB2 データ・リンク・マネージャー (Windows 版) 概説およびインストール	AIX および Windows 32 ビット・オペレーティング・システムの DB2 データ・リンク・マネージャーで、計画、インストール、構成を行う場合の情報を提供します。	GC88-8532	db2z6
		db2z6x70	
DB2 エンタープライズ拡張エディション (UNIX 版) 概説およびインストール	UNIX ベースのプラットフォームでの DB2 エンタープライズ拡張エディションの計画、インストール、および構成に関する情報を提供します。また、この資料はサポートされている多数のクライアントのインストールおよびセットアップについても説明します。	GC88-8530	db2v3
		db2v3x70	
DB2 エンタープライズ拡張エディション (Windows 版) 概説およびインストール	Windows 32 ビット・オペレーティング・システムの DB2 エンタープライズ拡張エディションで、計画、インストール、および構成を行う場合の情報を提供します。また、この資料はサポートされている多数のクライアントのインストールおよびセットアップについても説明します。	GC88-8529	db2v6
		db2v6x70	

表9. DB2 情報 (続き)

資料名	説明	資料番号 PDF ファイル名	HTML ディレクトリー
DB2 ユニバーサル・データベース (OS/2 版) 概説およびインストール	OS/2 オペレーティング・システムでの DB2 ユニバーサル・データベースの計画、インストール、移行、および構成に関する情報を提供します。また、この資料はサポートされている多数のクライアントのインストールおよびセットアップについても説明します。	GC88-8534 db2i2x70	db2i2
DB2 ユニバーサル・データベース (UNIX 版) 概説およびインストール	UNIX ベースのプラットフォームでの DB2 ユニバーサル・データベースの計画、インストール、移行、および構成に関する情報を提供します。また、この資料はサポートされている多数のクライアントのインストールおよびセットアップについても説明します。	GC88-8536 db2ixx70	db2ix
DB2 ユニバーサル・データベース (Windows 版) 概説およびインストール	Windows 32 ビット オペレーティング・システムの DB2 ユニバーサル・データベースで、計画、インストール、移行、および構成を行う場合の情報を提供します。また、この資料はサポートされている多数のクライアントのインストールおよびセットアップについても説明します。	GC88-8537 db2i6x70	db2i6
DB2 パーソナル・エディション 概説およびインストール	OS/2 および Windows 32 ビット オペレーティング・システム 版の DB2 ユニバーサル・データベース パーソナル・エディションで、計画、インストール、移行、および構成を行う場合の情報を提供します。	GC88-8535 db2i1x70	db2i1
DB2 パーソナル・エディション (Linux 版) 概説およびインストール	サポートされる Linux 配布プログラムの DB2 ユニバーサル・データベース・パーソナル・エディションで、計画、インストール、移行、および構成を行う場合の情報を提供します。	GC88-8538 db2i4x70	db2i4
DB2 クエリー・パトローラー インストールの手引き	DB2 クエリー・パトローラーのインストール情報を提供します。	GC88-8526 db2iwx70	db2iw

表9. DB2 情報 (続き)

資料名	説明	資料番号 PDF ファイル名	HTML ディレクトリー
ウェアハウス・マネージャ インストールの手引き	ウェアハウス・エージェント、ウェアハウス・トランスフォーマー、および情報カタログ・マネージャのインストール情報を提供します。	GC88-8572 db2idx70	db2id
プラットフォーム共通のサンプル・プログラム (HTML 形式)			
サンプル・プログラム (HTML)	DB2 のサポートするすべてのプラットフォームでのプログラム言語用に、サンプル・プログラム (HTML 形式) を提供します。これらのサンプル・プログラムは、参照用としてのみ提供されています。サンプルは、すべてのプログラミング言語で利用できるわけではありません。HTML サンプルが利用できるのは、DB2 アプリケーション開発クライアントがインストールされている場合だけです。 プログラムの詳細については、アプリケーション構築の手引き を参照してください。	資料番号なし	db2hs
リリース情報			
DB2 コネクト リリース情報	DB2 コネクトの資料には含められなかった最新の情報が収録されています。	注 #2 を参照してください。	db2cr
DB2 インストール情報	DB2 ブックには含められなかったインストールに関する最新の情報が収録されています。	製品 CD-ROM からのみ利用できます。	
DB2 リリース情報	DB2 ブックには含められなかった製品とその機能に関する最新の情報が収録されています。	注 #2 を参照してください。	db2ir

注:

1. ファイル名の 6 桁目の文字 *x* は、その資料の言語を表します。たとえば、ファイル名 db2d0e70 は、管理の手引き の英語版であることを示し、ファイル名 db2d0f70 は同じ資料のフランス語版を示します。資料の言語を表すためにファイル名の 6 桁目で使用されている文字は以下のとおりです。

言語	識別子
ブラジル・ポルトガル語	b
ブルガリア語	u
チェコ語	x
デンマーク語	d
オランダ語	q
英語	e
フィンランド語	y
フランス語	f
ドイツ語	g
ギリシャ語	a
ハンガリー語	h
イタリア語	i
日本語	j
韓国語	k
ノルウェー語	n
ポーランド語	p
ポルトガル語	v
ロシア語	r
簡体字中国語	c
スロベニア語	l
スペイン語	z
スウェーデン語	s
繁体字中国語	t
トルコ語	m

2. DB2 ブックには含められなかった最新の情報が、「リリース情報」で HTML 形式および ASCII ファイルとして利用できます。HTML 版は、インフォメーション・センターおよび製品 CD-ROM からご利用になれます。ASCII ファイルの参照方法:

- UNIX ベースのプラットフォームでは、ファイル `Release.Notes` を参照してください。このファイルは `DB2DIR/Readme/%L` ディレクトリーにあります。ここで `%L` は地域名を、`DB2DIR` は以下のものを表します。
 - `/usr/lpp/db2_07_01` (AIX の場合)
 - `/opt/IBMd2/V7.1` (HP-UX、DYNIX/ptx、Solaris、および Silicon Graphics IRIX の場合)
 - `/usr/IBMd2/V7.1` (Linux の場合)
- これ以外のプラットフォームでは、ファイル `RELEASE.TXT` を参照してください。このファイルは、製品がインストールされているディレクトリーにあります。OS/2 プラットフォームでは、**IBM DB2** フォルダをダブルクリックし、**Release Notes** アイコンをダブルクリックすることもできます。

PDF 資料の印刷

資料のハードコピー版が必要な場合、DB2 の資料 CD-ROM にある PDF ファイルを印刷することができます。Adobe Acrobat Reader を使用すれば、資料全体または特定のページを印刷することができます。ライブラリー内の各資料のファイルについては、635ページの表9 を参照してください。

Adobe Acrobat Reader の最新版は、Adobe の Web サイト <http://www.adobe.com> から入手できます。

PDF ファイルは、DB2 の資料 CD-ROM に収録されており、ファイル拡張子 PDF が付いています。PDF ファイルにアクセスするには以下のようにします。

1. DB2 の資料 CD-ROM を挿入します。UNIX ベースのプラットフォームの場合は、DB2 資料 CD-ROM をマウントします。マウントの手順については、概説およびインストール を参照してください。
2. Acrobat Reader を起動します。
3. 以下に示すいずれかの位置から必要な PDF ファイルを開きます。
 - OS/2 および Windows プラットフォームでは:
`x:%doc%language` ディレクトリー。ここで、*x* は CD-ROM ドライブを、*language* は 2 桁の言語を表す国コード (たとえば、EN は英語) を示します。
 - UNIX ベースのプラットフォームでは:
CD-ROM の `/cdrom/doc/%L` ディレクトリー。ここで、*/cdrom* は CD-ROM のマウント・ポイントを、*%L* は地域名を表します。

さらに、PDF ファイルを CD-ROM からローカル・ドライブまたはネットワーク・ドライブにコピーし、そこから参照することもできます。

印刷資料の注文方法

ハードコピー版の DB2 ブックは、個別に注文することができます。資料を注文するには、IBM 承認の販売業者または営業担当員に連絡してください。

オンライン・ヘルプへのアクセス

すべての DB2 構成要素で、オンライン・ヘルプを利用できます。以下の表に、さまざまな種類のヘルプを示します。

ヘルプの種類	内容	利用方法
コマンド・ヘルプ	コマンド行プロセッサの コマンド構文について説明 します。	コマンド行プロセッサの対話モードから、次のよ うに入力します。 ? <i>command</i> ここで <i>command</i> はキーワードまたはコマンド全体 を表します。 たとえば、? <i>catalog</i> と入力すると、すべての CATALOG コマンドに関するヘルプが表示され、 ? <i>catalog database</i> と入力すると、CATALOG DATABASE コマンドのヘルプが表示されます。
クライアント構成アシ スタントのヘルプ	そのウィンドウまたはノー トブックで実行できるタス クについて説明します。こ のヘルプは、知っておく必 要のある概説および前提条 件に関する情報を含みま す。また、ウィンドウやノ ートブックの制御の使用方 法を示します。	ウィンドウまたはノートブックから、「ヘルプ (Help)」押しボタンをクリックするか、または F1 キーを押します。
コマンド・センターの ヘルプ		
コントロール・センタ ーのヘルプ		
データウェアハウスセ ンターのヘルプ		
イベント・アナライザ ーのヘルプ		
情報カタログ・マネー ジャーのヘルプ		
サテライト管理センタ ーのヘルプ		
スクリプト・センター のヘルプ		

ヘルプの種類	内容	利用方法
メッセージ・ヘルプ	メッセージの原因、および取るべき処置を説明します。	<p>コマンド行プロセッサの対話モードから、次のように入力します。</p> <pre>? XXXnnnnn</pre> <p>ここで、<i>XXXnnnnn</i> は有効なメッセージ識別子を表します。</p> <p>たとえば、? SQL30081 と入力すると、メッセージ SQL30081 に関するヘルプを表示します。</p> <p>一度に 1 画面分のメッセージ・ヘルプを表示させるには、次のように入力します。</p> <pre>? XXXnnnnn more</pre> <p>メッセージ・ヘルプをファイルに保管するには、次のように入力します。</p> <pre>? XXXnnnnn > filename.ext</pre> <p>ここで、<i>filename.ext</i> はメッセージ・ヘルプを保管するファイルを表します。</p>
SQL ヘルプ	SQL ステートメントの構文について説明します。	<p>コマンド行プロセッサの対話モードから、次のように入力します。</p> <pre>help statement</pre> <p>ここで、<i>statement</i> は SQL ステートメントを表します。</p> <p>たとえば、help SELECT と入力すると、SELECT ステートメントのヘルプが表示されます。</p> <p>注: UNIX ベースのプラットフォームでは、SQL ヘルプを利用できません。</p>
SQLSTATE ヘルプ	SQL 状態およびクラス・コードについて説明します。	<p>コマンド行プロセッサの対話モードから、次のように入力します。</p> <pre>? sqlstate or ? class code</pre> <p>ここで、<i>sqlstate</i> は有効な 5 桁の SQL 状態を、<i>class code</i> は SQL 状態の最初の 2 桁を表します。</p> <p>たとえば、? 08003 によって SQL 状態 08003 のヘルプが表示され、? 08 によってクラス・コード 08 のヘルプが表示されます。</p>

オンライン情報の表示

この製品に付属のブックは、ハイパーテキスト・マークアップ言語 (HTML) ソフトコピー形式です。ソフトコピー形式では情報を検索または表示したり、ハイパーテキスト・リンクを利用して関連情報に移動したりすることができます。また、1 つの端末を超えてライブラリーを容易に共用することができます。

オンライン・ブックやサンプル・プログラムは、HTML バージョン 3.2 仕様に準拠するすべてのブラウザを使って表示できます。

オンライン・ブックまたはサンプル・プログラムは、次のようにして表示します。

- DB2 管理ツールを実行している場合、インフォメーション・センターを使用します。
- ブラウザーで、**ファイル (File) → ページを開く (Open Page)** をクリックします。次のようなページを開いて、DB2 情報に関する説明とリンクを表示してください。

- UNIX ベースのプラットフォームでは、以下のページを開きます。

```
INSTHOME/sql1lib/doc/%L/html/index.htm
```

ここで %L はロケール名です。

- その他のプラットフォームでは、以下のページを開きます。

```
sql1lib¥doc¥html¥index.htm
```

パスは DB2 がインストールされているドライブです。

インフォメーション・センターをインストールしていない場合、**DB2 Information** アイコンをダブルクリックしてページを開くことができます。このアイコンは、ご使用のシステムに応じて、製品のメイン・フォルダー内または Windows 「スタート」メニューにあります。

Netscape ブラウザーのインストール

システムに Web ブラウザーがインストールされていない場合、製品の箱の中にある Netscape CD-ROM から Netscape をインストールすることができます。インストールに関する詳細な説明については、以下を参照してください。

1. Netscape CD-ROM を挿入します。
2. UNIX ベースのプラットフォームでは、CD-ROM をマウントします。マウントの手順については、**概説およびインストール** を参照してください。

3. インストールの手順については、`CDNAVnn.txt` ファイルを参照します。ここで、`nn` は 2 桁の言語識別子を表します。ファイルは CD-ROM のルート・ディレクトリーにあります。

インフォメーション・センターを使用した情報へのアクセス

インフォメーション・センターを使用すると、DB2 製品情報にすばやくアクセスすることができます。インフォメーション・センターは、DB2 管理ツールを使用できるすべてのプラットフォームで利用できます。

インフォメーション・センターは「インフォメーション・センター (Information Center)」アイコンをダブルクリックすることによってオープンできます。このアイコンのある場所はシステムによって異なります。メイン・プロダクト・フォルダーか Windows の「スタート」メニューのどちらかです。

Windows プラットフォームの DB2 では、ツールバーおよびヘルプ・メニューを使用して、インフォメーション・センターにアクセスすることもできます。

インフォメーション・センターは 6 種類の情報を提供します。適切なタブをクリックすると、種類ごとに提供されているトピックが表示されます。

タスク (Tasks)

DB2 を使用して実行できる主要なタスク。

参照 (Reference)

DB2 参照情報 (キーワード、コマンド、API など)。

ブック (Books)

DB2 ブック。

トラブルシューティング (Troubleshooting)

エラー・メッセージのカテゴリと、メッセージに対する回復処置。

サンプル・プログラム (Sample Programs)

DB2 アプリケーション開発クライアントに付属のサンプル・プログラム。DB2 アプリケーション開発クライアントをインストールしていない場合、このタブは表示されません。

Web

WWW 上にある DB2 情報。この情報にアクセスするには、ご使用のシステムから Web への接続が必要です。

リストから項目を 1 つ選択すると、インフォメーション・センターはビューアーを立ち上げて情報を表示します。選択した情報の種類に応じて、ビューアーはシステム・ヘルプ・ビューアー、エディター、または Web ブラウザーです。

インフォメーション・センターには検索機能が備わっており、リストを参照せずに特定のトピックを探すことができます。

テキストの全検索を行うには、インフォメーション・センター内のハイパーテキスト・リンク「**DB2 オンライン情報の検索 (Search DB2 Online Information)**」検索フォームに従います。

通常、HTML 検索サーバーは自動的に始動します。HTML 情報の検索がうまくいかない場合は、以下の方法の 1 つを使用して、検索サーバーを始動しなければならない場合もあります。

Windows では

「スタート」をクリックし、「プログラム」→「IBM DB2」→「Information」→「Start HTML Search Server」を選択します。

OS/2 では

「DB2 (OS/2 版)」フォルダーをダブルクリックして、「Start HTML Search Server」アイコンをダブルクリックします。

HTML 情報の検索でこの他の問題が発生した場合は、リリース情報を参照してください。

注: 検索機能は、Linux、DYNIX/ptx、および Silicon Graphics IRIX 環境では利用できません。

DB2 ウィザードの使用

ウィザードを使用すると、各タスクをステップごとに進めることによって、さまざまな管理タスクを遂行することができます。ウィザードは、コントロール・センターおよびクライアント構成アシスタントを通して使用できます。以下の表では、ウィザードとその目的をリストしています。

注: データベース作成、索引作成、複数サイト更新の構成、およびパフォーマンス構成ウィザードは、区分化データベース環境で使用できます。

ウィザード	内容	利用方法
データベース追加 (Add Database)	クライアント・ワークステーション上にデータベースのカatalogを作成します。	クライアント構成アシスタントから、「追加 (Add)」をクリックします。

ウィザード	内容	利用方法
データベース・バックアップ (Back up Database)	バックアップ計画を決定、作成、およびスケジューリングします。	「コントロール・センター (Control Center)」からバックアップするデータベースを右クリックし、「バックアップ (Backup)」→「ウィザードを使用するデータベース (Database Using Wizard)」を選択します。
複数サイト更新の構成 (Configure Multisite Update)	複数サイト更新、分散トランザクション、または 2 フェーズ・コミットを構成します。	「コントロール・センター (Control Center)」から、「データベース (Databases)」フォルダーを右クリックして、「複数サイト更新 (Multisite Update)」を選択します。
データベース作成 (Create Database)	データベースを作成し、いくつかの基本的な構成タスクを実行します。	「コントロール・センター (Control Center)」から、「データベース (Databases)」フォルダーを右クリックして、「作成 (Create)」→「ウィザードを使用するデータベース (Database Using Wizard)」を選択します。
表作成 (Create Table)	基本的なデータ・タイプを選択して、表の基本キーを作成します。	「コントロール・センター (Control Center)」から、「表 (Tables)」アイコンを右クリックして、「作成 (Create)」→「ウィザードを使用する表 (Table Using Wizard)」を選択します。
表スペース作成 (Create Table Space)	新しい表スペースを作成します。	「コントロール・センター (Control Center)」から、「表スペース (Table Spaces)」アイコンを右クリックして、「作成 (Create)」→「ウィザードを使用する表スペース (Table Space Using Wizard)」を選択します。
索引作成 (Create Index)	すべての照会について、作成すべき索引および除去すべき索引を提案します。	「コントロール・センター (Control Center)」から、「索引 (Index)」アイコンを右クリックして、「作成 (Create)」→「ウィザードを使用する索引 (Index Using Wizard)」を選択します。

ウィザード	内容	利用方法
パフォーマンス構成 (Performance Configuration)	ビジネス要件に適合するように構成パラメーターを更新して、データベースのパフォーマンスを調整します。	「コントロール・センター (Control Center)」から、調整したいデータベースを右クリックして、「ウィザードを使用するパフォーマンスの構成 (Configure Performance Using Wizard)」を選択します。 区分データベース環境では、「Database Partitions」視点から、調整したい最初のデータベース区画を右クリックして、「ウィザードを使用するパフォーマンスの構成 (Configure Performance Using Wizard)」を選択します。
データベース復元 (Restore Database)	障害の後、データベースを回復します。どのバックアップを使用し、どのログを再生するかを判別を支援します。	「コントロール・センター (Control Center)」から復元するデータベースを右クリックし、「復元 (Restore)」→「ウィザードを使用するデータベース (Database Using Wizard)」を選択します。

文書サーバーのセットアップ

デフォルトでは、DB2 情報はローカル・システムにインストールされます。つまり、DB2 情報にアクセスする必要のある各担当者が同じファイルをインストールする必要があります。DB2 情報を 1 か所に格納するには、次のようにします。

1. `¥sqllib¥doc¥html` のすべてのファイルとサブディレクトリーを、ローカル・システムから Web サーバーにコピーします。各ブックには独自のサブディレクトリーがあり、そのブックを構成する必要な HTML および GIF ファイルが入っています。ディレクトリー構造は常に同じ状態に保つ必要があります。
2. Web サーバーを構成して、ファイルを新しい場所で検索するようにします。さらに詳しい情報については、インストールおよび構成 補足の NetQuestion 付録を参照してください。
3. インフォメーション・センターの Java バージョンをご使用の場合は、すべての HTML ファイルのベース URL を指定できます。この URL はブックのリストに使用してください。

4. 資料ファイルが表示されるようになったなら、よく使うトピックにはブックマークを付けておいてください。ブックマークを付けるページは、たとえば以下のものがあります。
 - ブックのリスト
 - 頻繁に使用されるブックの目次
 - 頻繁に参照する情報 (たとえば、ALTER TABLE トピックなど)
 - 検索フォーム

中央のマシンから DB2 ユニバーサル・データベース オンライン文書ファイルを提供する方法については、インストールおよび構成 補足の NetQuestion 付録を参照してください。

オンライン情報の検索

HTML ファイルの情報を検索するには、以下の方法のどれか 1 つを使用してください。

- 最上部にある「**検索 (Search)**」をクリックします。検索フォームを使用して特定のトピックを見つけます。この機能は、Linux、DYNIX/ptx、または Silicon Graphics IRIX 環境ではご利用になれません。
- 最上部にある「**索引 (Index)**」をクリックします。索引を使用して、ブック内の特定のトピックを見つけます。
- HTML 資料またはヘルプの目次あるいは索引を表示してから、Web ブラウザーの検索機能を利用して資料内の特定のトピックを見つけます。
- Web ブラウザーのブックマーク機能を使用して、特定のトピックにすばやく戻ります。
- インフォメーション・センターの検索機能を使用して、特定のトピックを検索します。詳しくは、649ページの『インフォメーション・センターを使用した情報へのアクセス』を参照してください。

付録D. 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、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]

アクセス・パス
最適化 561
新規に作成する 564
アンカタログ
システム・データベース・ディレクトリー 587
データベース項目 587
ホスト DCS データベース項目 589
暗黙接続
データベース・アクセス 135
異常終了
再始動 537
一時ファイル
ロード・ユーティリティー 425
一貫性
必須、バックアップに 162
移動、データベース間のデータの 345
イベント・アナライザー 45
インスタンスからのデータベース区画サーバーの除去 88
インスタンスの移行 63
インスタンスの更新 65
インスタンスの作成 57
インスタンスの除去 60
インスタンスのリスト 62
インスタンスへのデータベース区画サーバーの追加 85
インストール
Netscape ブラウザー 648
インフォメーション・センター 649

インポート
コード・ページに関する考慮事項 345
制限 346
存在しない表または階層への 345
タイプ表への 346
データベース・アクセス、DB2
コネクトを使う 346
ファイルをデータベース表へ 332
ファイル・タイプ修飾子 348
リモート・データベースへの 344
DB2 データ・リンク・マネージャーに関する考慮事項 347
PC/IXF ファイルを表へ 335
PC/IXF、複数パーツ・ファイル 346
ウィザード
索引 651
タスクを遂行する 650
データベース作成 651
データベース追加 650, 651, 652
データベース復元 652
データベース・バックアップ 650
パフォーマンス構成 651
表作成 651
表スペース作成 651
複数サイト更新の構成 651
エクスポート
データベース表からファイルへ 257
ファイル・タイプ修飾子 263
エラー・メッセージ
データベース構成ファイル 290
ドロップ、リモート・データベースの 252
バックアップ中 162

エラー・メッセージ (続き)
無効なチェックサム、データベース構成ファイル 532, 607
無効なチェックサム、データベース・マネージャー構成ファイル 529, 533, 601, 609
大文字小文字の区別 139
命名規則における 631
オンライン情報
検索 653
表示 648
オンライン・ヘルプ 646

[カ行]

カーソル固定 (CS)
変更 232
開始、コントロール・センターの 23
開始、トレースの 114
回復
ロールフォワードなし 544, 546
ロールフォワードを用いた 550
回復、データベースの 539
カタログ作成
データベース 194
ホスト・データベース 199
監査機能管理者ツール 13
環境変数
改行文字除去オプション
(-n) 129
警告メッセージ表示オプション
(-w) 133
コマンド・エラー時の実行の停止
(-s) 131
自動コミット・オプション
(-c) 127
出力表示オプション (-o) 130
冗長出力オプション (-v) 132
ステートメント終了文字オプション (-t) 132

- 環境変数 (続き)
 - すべての出力のファイルへの保管オプション (-z) 133
 - 入力ファイルからの読み取りオプション (-f) 128
 - 報告ファイルへの保管オプション (-r) 131
 - 履歴ファイルへのコマンドのログ (-l) 129
 - DB2 対話式プロンプト表示オプション (-p) 130
 - DB2OPTIONS 125
 - SQLCA データ表示オプション (-a) 127
 - SQLCODE/SQLSTATE 表示オプション (-e) 128
- 管理サーバー (DAS)
 - 構成 269
 - 作成 3
 - 除去 3
- キーワード
 - 構文 627
- 記憶域
 - 物理的 517
- 規則、命名
 - データベース・マネージャ・オブジェクト 631
- 行継続文字
 - コマンド行プロセッサ内の 135
- 継続文字、行
 - コマンド行プロセッサ内の 135
- 権限と特権
 - 付与、データベース作成時の 241
- 権限レベル
 - 間接定義された 273
 - 直接定義された 273
 - データベース作成のための付与 241
 - 報告 272
- 言語識別子
 - ブック 643
- 検索
 - オンライン情報 650, 653
- 構成、データベース
 - 更新 606
 - リセット、省略時値に 531
 - 例 279
- 構成、データベース・マネージャ
 - 例 292
- 構成、ADMIN
 - リセット、省略時値に 528
 - 例 270
- 構成、CLI
 - 例 275
- 構文
 - コマンド行プロセッサ SQL ステートメント 617
 - コマンド行プロセッサ内ではサポートされないホスト変数の 141
 - 構文図 627
 - コマンド行プロセッサ
 - アクセス、データベースへの 122
 - アクセス、ヘルプへの 123
 - オプション 125
 - コマンド・モード 122
 - シェル・コマンド 122
 - 終了 122, 495, 586
 - 使用 135
 - 説明 122
 - 対話式入力モード 122
 - バッチ・モード 122
 - 呼び出し 122
 - コマンド構文
 - 解釈 627
 - コミットされていない読み取り (UR)
 - 変更 232
 - コミットなし (NC)
 - 変更 232
 - コンテナ・タグの解放 119
 - コントロール・センター 23
- [サ行]**
 - 最新情報 644
 - 最適化 517
 - 索引
 - 再編成 524, 527
 - 統計 561
- 索引ウィザード 651
- サンプル・データベースの作成 100
- サンプル・プログラム
 - プラットフォーム共通の 643
 - HTML 643
- 事象モニター生産性向上ツール 47
- システム・コマンド 1
- システム・データベース・ディレクトリ
 - アンカタログ 587
- 自動ローダー 14
- 修飾子、ファイル・タイプ
 - インポート・ユーティリティに使用する 348
 - エクスポート・ユーティリティの 263
 - ロード・ユーティリティで使用する 437
- 終了 586
 - 異常 537
 - 正常 584
- 省略時値
 - データベース構成、リセット 531
 - ADMIN 構成、リセット 528
- スキーマ
 - 作成、データベース作成時の 240
- ストアード・プロシージャ
 - 呼び出し 360
- すべてのパッケージの再バインド 98
- 制御データベース 25
- 静止
 - ファントム 493
- 接続構成インポート・ツール 28
- 接続構成エクスポート・ツール 26
- 切斷
 - コマンド行プロセッサ・フロントエンド・プロセスおよびバックエンド・プロセス 586
- セットアップ、文書サーバーの 652
- 宣言生成プログラム 39
- 装置、テープ 159

[タ行]

ダンプ、トレースをファイルへ 114
データの断片化
 消去、表の再編成による 517
データベース
 暗黙接続 135
 移行、コマンド 455
 インポート、ファイルを表へ 332
 エクスポート、表のファイルへの 257
 回復 550
 カタログ作成 194
 間接 (indirect) ディレクトリー項目の定義 371
 検査、許可の 272
 項目の除去 (アンカタログ) 587
 再始動 537
 再編成 520
 削除、ログ・ファイルによる回復の確認 252
 情報 314
 除去 251
 除去、ホスト DCS 項目の 589
 接続の概説 135
 統計 561
 復元 (再構築) 539
 変更、ディレクトリー内の注釈 230
 ホーム (home) ディレクトリー項目の定義 371
 モニター、リセット 535
 リモート (remote) ディレクトリー項目の定義 371
 ロード、ファイルを表へ 411
 ロールフォワード回復 550
データベース移動ツール 77
データベース区画サーバー構成の変更 83
データベース構成
 更新 606
 ネットワーク・パラメーター値 607
 リセット、省略時値に 531
 例 279

データベース作成ウィザード 651
データベース事前移行ツール 34
データベース接続サービス (DCS) ディレクトリー
 アンカタログ、項目の 589
データベース追加ウィザード 650, 651, 652
データベース・アクセス
 開始、データベース・マネージャーの 135
データベース・システム・モニター
 GET DATABASE MANAGER MONITOR SWITCHES 307
 GET MONITOR SWITCHES 310
 GET SNAPSHOT 313
 RESET MONITOR 535
 UPDATE MONITOR SWITCHES 613
データベース・ディレクトリー
 定義 371
 内容例 369
 変更、注釈の 230
データベース・バックアップ
 活動記録ファイル 487
データベース・バックアップ・ウィザード 650
データベース・パフォーマンス値のリセット 90
データベース・マネージャー
 アクセス、コマンド・プロンプトからの 1
 インスタンス 309
 開始 576
 システム・コマンド 1
 停止 582
 統計 313
 モニター・スイッチの検査 307, 310
データベース・マネージャー構成
 ネットワーク・パラメーター値 609
 ファイル 305
 例 292
データベース・モニター
 説明 613

データ保全性
 維持、分離レベルを使用した 232
データ・スキュー、ノード・グループ中でのデータの再配布 505
テープ装置 159
ディレクトリー
 アンカタログ 587
 削除、項目の 595
 システム・データベース、除去 587
 データベース、注釈の変更 230
 データベース接続サービス (DCS)、項目のアンカタログ 589
 ノード、項目の除去 595
統計
 再編成、索引の 524
 データベース 561
 データベース・マネージャー 313
 REORGCHK 520
特権
 間接定義された 273
 直接定義された 273
特権と権限
 付与、データベース作成時の 241
トレース 113

[ナ行]

ノード
 ディレクトリー、項目の除去 595
ノード、SOCKS 227

[ハ行]

バインド
 暗黙的に作成されるスキーマ 181, 484
 エラー、バインド中の 240
バインド・ファイル記述ツール 22
パスワード
 ATTACH を介した変更 154
バックアップの検査 31

パッケージ
再作成 496
新規のアクセス・パスの強制、統計の実行後 564
パフォーマンス、改善する 526
再編成表による 518
パフォーマンス構成ウィザード 651
パフォーマンス・カウンター登録ユーティリティ 91
パフォーマンス・モニター登録ツール 92
パラメーター
構文 627
反復可能読み取り (RR)
変更 232
表
インポート、ファイルの 332
エクスポート、ファイルへの 257
再編成、必要性の判別 520
統計 561
ロード、ファイルを表へ 411
表作成ウィザード 651
表示
オンライン情報 648
表スペース作成ウィザード 651
表スペース状態の獲得 112
表の再編成
コマンド 517
ファイル形式
インポート用、ファイルを表へ 339
表のファイルへのエクスポート用 259
ファイル・タイプ修飾子
インポート・ユーティリティ 348
エクスポート・ユーティリティ 263
ロード・ユーティリティ 437
ファントム静止 493
復元、旧バージョンの DB2 データベース 539
復元ウィザード 652
複数サイト更新の構成ウィザード 651

複数ページ・ファイル割り振りの使用可能化 44
ブック 633, 645
変数
構文 627
ベンチマーク・ツール 15
ホスト変数
コマンド行プロセッサではサポートされない 141
ホスト・システム
カタログ作成、～にあるデータベースの 199
除去、DCS カatalog項目の 589
DB2 コネクトがサポートする接続 200

[マ行]

未確定トランザクション・フィールドの説明 389
命名、出力のバイナリー・ファイルに 114
命名規則
データベース・マネージャー・オブジェクト 631
メタ文字 139
メッセージ
アクセス、ヘルプ・テキストへの 123
文字、特殊
許可されている、CLP コマンド内に 139
モニター、データベースの 307, 310

[ヤ行]

ユーザー
許可 272
読み取り固定 (RS)
変更 232

[ラ行]

ライセンス管理ツール 69
リダイレクト、出力の 140

リモート・サーバー
呼び出し、ストアード・プロシージャの 360
リモート・データベース移行 29
リリース情報 644
ローカル・ノード
アンカatalog 595
ロード
ファイルをデータベース表へ 411
ファイル・タイプ修飾子 437
ロールフォワード
使用可能、バックアップ時 162
ログ順序番号の検出 51
ログ・ファイル
ロールフォワード中のリスト 554
ロック
リセット、最大値を省略時値に 531

[ワ行]

ワークステーション、リモート
アンカatalog、ローカル・ワークステーションから 595
カタログ作成、データベースの 194
除去、データベースのカタログ項目の 587

A

action
プリコンパイル / バインド・オプション 169, 463
ACTIVATE DATABASE 148
ADD DATALINKS MANAGER 150
ADD NODE 152
ADMIN 構成
ネットワーク・パラメーター値 600
ファイル 271
リセット、省略時値に 528
例 270

agentpri
データベース・マネージャー構成
パラメーター 294

agent_stack_sz
データベース・マネージャー構成
パラメーター 294

anyorder 437

APPC ノード
アンカatalog 595

applheapsz
データベース構成パラメーター
281

app_ctl_heap_sz
データベース構成パラメーター
281

ASC、インポート・ファイル・タイ
プとして 337

aslheapsz
データベース・マネージャー構成
パラメーター 294

ATTACH 154

audit_buf_sz
データベース構成パラメーター
281
データベース・マネージャー構成
パラメーター 294

authentication
データベース・マネージャー構成
パラメーター 294

autorestart
データベース構成パラメーター
281

avg_appls
データベース構成パラメーター
282

B

backbufsz
データベース・マネージャー構成
パラメーター 294

BACKUP DATABASE 157

backup_pending
データベース構成パラメーター
282

binarynumerics 446

BIND 165
新しいアクセス・パスを作成する
ために 564

bindfile
プリコンパイル・オプション
464

blocking
プリコンパイル / バインド・オブ
ション 170, 464

buffpage
データベース構成パラメーター
282

C

CALL 184
CALL ステートメント
実行、CLP による 617, 620

CATALOG APPC NODE 186

CATALOG APPCLU NODE 189

CATALOG APPN NODE 191

CATALOG DATABASE 194

CATALOG DCS DATABASE 199

CATALOG GLOBAL
DATABASE 202

CATALOG IPX/SPX NODE 204

CATALOG LDAP DATABASE 208

CATALOG LDAP NODE 213

CATALOG LOCAL NODE 215

CATALOG NAMED PIPE
NODE 218

CATALOG NETBIOS NODE 221

CATALOG ODBC DATA
SOURCE 224

CATALOG TCP/IP NODE 226

catalogcache_sz
データベース構成パラメーター
282

catalog_noauth
データベース・マネージャー構成
パラメーター 294

CCSIDG
プリコンパイル / バインド・オブ
ション 170, 465

CCSIDM
プリコンパイル / バインド・オブ
ション 170, 465

CCSIDS
プリコンパイル / バインド・オブ
ション 170, 465

CHANGE DATABASE
COMMENT 230

CHANGE ISOLATION LEVEL 232

chardel 263, 354, 448

charsub
プリコンパイル / バインド・オブ
ション 171, 465

chngpgs_thresh
データベース構成パラメーター
282

CLI 構成
例 275

CLIPKG
プリコンパイル / バインド・オブ
ション 171

CLOSE ステートメント
実行、CLP による 617

cnulreqd
プリコンパイル / バインド・オブ
ション 171, 466

codepage 441
データベース構成パラメーター
282

codeset
データベース構成パラメーター
282

codel 263, 354, 448

collection
プリコンパイル / バインド・オブ
ション 172, 466

comm_bandwidth
データベース・マネージャー構成
パラメーター 295

compound 348

connect
プリコンパイル・オプション
466

CONNECT ステートメント
実行、CLP による 617
データベース接続 135

conn_elapse	データベース・マネージャ構成 パラメーター 295	DB2 コネクト サポートされる他のシステムへの 接続 200	db2dclgn 39
copyprotect	データベース構成パラメーター 282	DB2 コマンド・ウィンドウのオープ ン 37	db2drdat 42
country	データベース構成パラメーター 282	DB2 索引アドバイザー 10	db2empfa 44
cpuspeed	データベース・マネージャ構成 パラメーター 295	DB2 シンクロナイザーの開始 111	db2eva 45
CREATE DATABASE	235	DB2 対話機能 CLI 35	db2evmon 47
		DB2 統計および DDL 抽出ツール 71	db2exfmt 49
		PC/IXF ファイルのインポート時 に使用 346	db2expln 50
		DB2 の開始 109	db2flsn 51
		DB2 の停止 110	db2fs 53
		DB2 プロファイル・レジストリー・ コマンド 102	db2gov 54
		DB2 ライブラリー	db2govlg 56
		印刷版のブックの注文 645	db2icrt 57
		インフォメーション・センター 649	db2idrop 60
		ウィザード 650	db2ilist 62
		オンライン情報の検索 653	db2imigr 63
		オンライン情報の表示 648	db2ipxad 64
		オンライン・ヘルプ 646	db2iupdt 65
		構成内容 633	db2ldcfg 68
		最新情報 644	db2licm 69
		セットアップ、文書サーバーの 652	db2look 71
		ブック 633	db2move 77
		ブックの言語識別子 643	db2mscs 82
		PDF 資料の印刷 645	db2nchg 83
			db2ncrt 85
			db2ndrop 88
			DB2OPTIONS
			環境変数 125
			db2perfc 90
			db2perfi 91
			db2perfr 92
			db2profc 93
			db2profp 96
			db2rbind 98
			db2sampl 100
			db2set 102
			db2sql92 106
			db2start 109, 576
			db2stop 110, 582
			db2sync 111
			db2tbst 112
			db2tre 113
			db2uiddl 117
			db2untag 119
			dbheap
			データベース構成パラメーター 283
db2	コマンド構文 122 CMD 記述 122	db2admin 3	
DB2 SQL Explain ツール	50	db2adutl 5	
DB2 SQLJ プロファイル・カスタ マイザー 93		db2advis 10	
DB2 SQLJ プロファイル・プリン ター 96		db2atld 14	
DB2 管理サーバー 3		db2batch 15	
DB2 管理プログラム 54		db2bfd 22	
DB2 管理プログラム・ログ照会 56		db2cc 23	
		db2cdbcr 25	
		db2cfexp 26	
		db2cfimp 28	
		db2cidmg 29	
		db2ckbkp 31	
		db2ckmig 34	
		db2cli 35	
		db2cmd 37	

DEACTIVATE DATABASE 243	dft_mon_lock	dir_type
dec	データベース・マネージャ構成	データベース・マネージャ構成
プリコンパイル / バインド・オプション 172, 466	パラメーター 295	パラメーター 297
decdel	dft_mon_sort	disconnect
プリコンパイル / バインド・オプション 172, 467	データベース・マネージャ構成	プリコンパイル・オプション
DECLARE CURSOR ステートメント	パラメーター 296	468
実行, CLP による 618	dft_mon_stmt	discover
decplusblank 263, 354, 448	データベース・マネージャ構成	データベース・マネージャ構成
decpt 263, 354, 448	パラメーター 296	パラメーター 297
deferred_prepare	dft_mon_table	discover_comm
プリコンパイル・オプション	データベース・マネージャ構成	データベース・マネージャ構成
467	パラメーター 296	パラメーター 297
degree	dft_mon_uow	discover_db
プリコンパイル / バインド・オプション 173, 468	データベース・マネージャ構成	データベース構成パラメーター
delprioritychar 355, 449	パラメーター 296	284
DEREGISTER 245	dft_prefetch_sz	discover_inst
DESCRIBE 247	データベース構成パラメーター	データベース・マネージャ構成
DETACH 250	283	パラメーター 297
dftdbpath	dft_queryopt	dlchktime
データベース・マネージャ構成	データベース構成パラメーター	データベース構成パラメーター
パラメーター 296	283	285
dft_account_str	dft_refresh_age	dldel 264, 355, 449
データベース・マネージャ構成	データベース構成パラメーター	dl_expint
パラメーター 295	283	データベース構成パラメーター
dft_client_adpt	dft_sqlmathwarn	284
データベース・マネージャ構成	データベース構成パラメーター	dl_num_copies
パラメーター 295	284	データベース構成パラメーター
dft_client_comm	diaglevel	284
データベース・マネージャ構成	データベース・マネージャ構成	dl_time_drop
パラメーター 295	パラメーター 296	データベース構成パラメーター
dft_degree	diagpath	284
データベース構成パラメーター	データベース・マネージャ構成	dl_token
283	パラメーター 296	データベース構成パラメーター
dft_extent_sz	dir_cache	284
データベース構成パラメーター	データベース・マネージャ構成	dl_upper
283	パラメーター 296	データベース構成パラメーター
dft_loadrec_ses	dir_obj_name	284
データベース構成パラメーター	データベース構成パラメーター	dos_rqrioblk
283	284	データベース・マネージャ構成
dft_mon_bufpool	データベース・マネージャ構成	パラメーター 297
データベース・マネージャ構成	パラメーター 296	DRDA トレース 42
パラメーター 295	dir_path_name	drda_heap_sz
	データベース・マネージャ構成	データベース・マネージャ構成
	パラメーター 296	パラメーター 297

dumpfile 443
dynamicrules
 プリコンパイル / バインド・オプション 173, 468

E

ECHO 256
estore_seg_sz
 データベース構成パラメーター 285
explain
 バインド・オプション 174, 469
Explain 表フォーマット・ツール 49
explsnap
 プリコンパイル / バインド・オプション 174, 470
EXPORT 257

F

fastparse 438
fcm_num;buffers
 データベース・マネージャー構成パラメーター 298
fcm_num_anchors
 データベース・マネージャー構成パラメーター 297
fcm_num_connect
 データベース・マネージャー構成パラメーター 298
fcm_num_rqb
 データベース・マネージャー構成パラメーター 298
federated
 データベース・マネージャー構成パラメーター 298
 プリコンパイル / バインド・オプション 175, 470
FETCH ステートメント
 実行、CLP による 618
fileserv
 データベース・マネージャー構成パラメーター 298
First Steps 53
FORCE APPLICATION 267
forcein 355, 450

funcpath
 プリコンパイル / バインド・オプション 175, 470

G

generatedignore 348, 438
generatedmissing 348, 438
generatedoverride 439
generic
 プリコンパイル / バインド・オプション 175, 471
GET ADMIN
 CONFIGURATION 269
GET AUTHORIZATIONS 272
GET CLI CONFIGURATION 274
GET CONNECTION STATE 277
GET DATABASE
 CONFIGURATION 279
GET DATABASE MANAGER
 CONFIGURATION 292
GET DATABASE MANAGER
 MONITOR SWITCHES 307
GET INSTANCE 309
GET MONITOR SWITCHES 310
GET SNAPSHOT 312
 UPDATE MONITOR SWITCHES
 への影響 613
grant
 バインド・オプション 176
grant_group
 バインド・オプション 176
grant_user
 バインド・オプション 176

H

HELP 330
HTML
 サンプル・プログラム 643

I

identityignore 348, 439
identitymissing 348, 439
identityoverride 439

implieddecimal 350, 443
IMPORT 332
indexfreespace 440
indexixf 356
indexrec
 データベース構成パラメーター 285
 データベース・マネージャー構成パラメーター 298
indexschema 356
indexsort
 データベース構成パラメーター 285
initdari_jvm
 データベース・マネージャー構成パラメーター 298
INITIALIZE TAPE 359
insert
 プリコンパイル / バインド・オプション 176, 471
intra_parallel
 データベース・マネージャー構成パラメーター 298
INVOKE STORED
 PROCEDURE 360
IPX/SPX インターネットワーク・アドレスの獲得 64
IPX/SPX ノード
 アンカタログ 595
ipx_socket
 データベース・マネージャー構成パラメーター 299
ISOLATION
 CHANGE ISOLATION LEVEL コマンド 232
isolation
 プリコンパイル / バインド・オプション 176, 472
J
java_heap_sz
 データベース・マネージャー構成パラメーター 299

jdk11_path
データベース・マネージャー構成
パラメーター 299

K

keepblanks 355, 449
keepdari
データベース・マネージャー構成
パラメーター 299

L

langlevel
プリコンパイル・オプション
472
LDAP 環境の構成 68
level
プリコンパイル・オプション
474
LIST ACTIVE DATABASES 362
LIST APPLICATIONS 364
LIST COMMAND OPTIONS 367
LIST DATABASE DIRECTORY 369
LIST DATALINKS
MANAGERS 373
LIST DCS APPLICATIONS 374
LIST DCS DIRECTORY 378
LIST DRDA INDOUBT
TRANSACTIONS 380
LIST HISTORY 382
LIST INDOUBT
TRANSACTIONS 386
LIST NODE DIRECTORY 391
LIST NODEGROUPS 394
LIST NODES 397
LIST ODBC DATA SOURCES 398
LIST PACKAGES 400
LIST PACKAGES/TABLES 400
LIST TABLES 400
LIST TABLESPACE
CONTAINERS 403
LIST TABLESPACES 405
LOAD 411
一時ファイル 425
LOAD QUERY 453

LOAD (sqlload) 411
lobsinfile 263, 348, 440
locklist
データベース構成パラメーター
285
locktimeout
データベース構成パラメーター
286
logbufsz
データベース構成パラメーター
286
logfilsiz
データベース構成パラメーター
286
loghead
データベース構成パラメーター
286
logpath
データベース構成パラメーター
286
logprimary
データベース構成パラメーター
286
logretain
データベース構成パラメーター
286
logsecond
データベース構成パラメーター
286
log_retain_status
データベース構成パラメーター
286
longerror
プリコンパイル・オプション
474

M

maxagents
データベース・マネージャー構成
パラメーター 300
maxappl
データベース構成パラメーター
286

maxcagents
データベース・マネージャー構成
パラメーター 300
maxdari
データベース・マネージャー構成
パラメーター 300
maxfilop
データベース構成パラメーター
286
maxlocks
データベース構成パラメーター
287
maxtotfilop
データベース・マネージャー構成
パラメーター 300
max_connretries
データベース・マネージャー構成
パラメーター 299
max_coordagents
データベース・マネージャー構成
パラメーター 299
MAX_IDLEAGENTS 302
max_logicagents
データベース・マネージャー構成
パラメーター 300
max_rt_degree
データベース・マネージャー構成
パラメーター 300
max_time_diff
データベース・マネージャー構成
パラメーター 300
messages
プリコンパイル / バインド・オブ
ション 177, 474
MIGRATE DATABASE 455
mincommit
データベース構成パラメーター
287
min_priv_mem
データベース・マネージャー構成
パラメーター 300
mon_heap_sz
データベース・マネージャー構成
パラメーター 300

multipage_alloc
データベース構成パラメーター
287

N

NetBIOS ノード
アンカatalog 595
Netscape ブラウザー
インストール 648
newlogpath
データベース構成パラメーター
287
nname
データベース・マネージャー構成
パラメーター 301
nochecklengths 353, 356, 446, 450
nodefaults 349
nodetype
データベース・マネージャー構成
パラメーター 301
nodoubleled 264, 355, 449
noeofchar 350, 445
noheader 440
nolinemacro
プリコンパイル・オプション
475
norowwarnings 440
notifylevel
データベース・マネージャー構成
パラメーター 301
no_type_id 348
NULL スtring
使用、Blankを設定するときの
135
nullindchar 353, 446
numdb
データベース・マネージャー構成
パラメーター 302
numsegs
データベース構成パラメーター
288
num_db_backups
データベース構成パラメーター
287

num_estore_segs
データベース構成パラメーター
287
num_freqvalues
データベース構成パラメーター
287
num_initagents
データベース・マネージャー構成
パラメーター 301
num_initdaris
データベース・マネージャー構成
パラメーター 301
num_iocleaners
データベース構成パラメーター
287
num_ioservers
データベース構成パラメーター
287
num_poolagents
データベース・マネージャー構成
パラメーター 301
num_quantiles
データベース構成パラメーター
288

O

objectname
データベース・マネージャー構成
パラメーター 302
OPEN ステートメント
実行、CLP による 618
optlevel
プリコンパイル・オプション
475
output
プリコンパイル・オプション
475
owner
プリコンパイル / バインド・オブ
ション 177, 475

P

package
プリコンパイル・オプション
476
packeddecimal 447
pagefreespace 440
password
CONNECT による変更 620
pckcachesz
データベース構成パラメーター
288
PDF 645
PDF 資料の印刷 645
PRECOMPILE PROGRAM 457
PREP コマンド (PRECOMPILE
PROGRAM) 457
preprocessor
プリコンパイル・オプション
476
priv_mem_thresh
データベース・マネージャー構成
パラメーター 302
PRUNE HISTORY/LOGFILE 487

Q

qualifier
プリコンパイル / バインド・オブ
ション 177, 476
QUERY CLIENT 489
queryopt
プリコンパイル / バインド・オブ
ション 177, 476
query_heap_sz
データベース・マネージャー構成
パラメーター 302
QUIESCE TABLESPACES FOR
TABLE 491
QUIT 495

R

REBIND 496
reclen 353, 447
RECONCILE 500

rec_his_retentn
 データベース構成パラメーター
 288

REDISTRIBUTE NODEGROUP 505

REFRESH LDAP 509

REGISTER 511

release
 データベース構成パラメーター
 288
 データベース・マネージャー構成
 パラメーター 302
 プリコンパイル / バインド・オプション 178, 476

REORGANIZE TABLE 517

REORGCHK 520

RESET ADMIN
 CONFIGURATION 528

RESET DATABASE
 CONFIGURATION 531

RESET DATABASE MANAGER
 CONFIGURATION 533

RESET MONITOR 535

RESTART DATABASE 537

restbufsz
 データベース・マネージャー構成
 パラメーター 302

RESTORE DATABASE 539

restore_pending
 データベース構成パラメーター
 288

resync_interval
 データベース・マネージャー構成
 パラメーター 302

REWIND TAPE 549

ROLLFORWARD DATABASE 550

rollfwd_pending
 データベース構成パラメーター
 288

route_obj_name
 データベース・マネージャー構成
 パラメーター 303

rqrioblk
 データベース・マネージャー構成
 パラメーター 303

RUNSTATS 561

S

SELECT ステートメント
 解決、WHERE 文節内の未確定シ
 ンボルの 140
 実行、CLP による 619
 EXPORT コマンド内の 259

seqdetect
 データベース構成パラメーター
 289

SET CLIENT 566

SET RUNTIME DEGREE 570

SET TABLESPACE
 CONTAINERS 572

SET TAPE POSITION 575

sheapthresh
 データベース・マネージャー構成
 パラメーター 303

SIGNALRM 信号 581
 データベース・マネージャーの開
 始 581

SIGNALRM 信号、データベース・マ
 ネージャーの開始 581

SIGINT 信号、データベース・マネ
 ージャーの開始 581

SmartGuides
 ウィザード 650

SOCKS ノード 227

softmax
 データベース構成パラメーター
 289

sortheap
 データベース構成パラメーター
 289

spm_log_file_sz
 データベース・マネージャー構成
 パラメーター 303

spm_log_path
 データベース・マネージャー構成
 パラメーター 303

spm_max_resync
 データベース・マネージャー構成
 パラメーター 303

spm_name
 データベース・マネージャー構成
 パラメーター 303

SQL NULL 値
 コマンド行プロセッサ表記
 141

SQL ステートメント
 アクセス、ヘルプへの 123
 コマンド行プロセッサ (CLP)
 による実行 617

SQL92 準拠 SQL ステートメント・
 プロセッサ 106

sqlca
 プリコンパイル・オプション
 477

SQLDA 構造
 使用するサーバー・プロシージャ
 ーの呼び出しに関する制限事項
 360

sqlerror
 プリコンパイル / バインド・オブ
 ション 178, 477

sqlflag
 プリコンパイル・オプション
 478

sqlrules
 プリコンパイル・オプション
 478

sqluload - ロード 411

sqlwarn
 プリコンパイル / バインド・オブ
 ション 178, 479

ss_logon
 データベース・マネージャー構成
 パラメーター 304

START DATABASE
 MANAGER 576

start_stop_time
 データベース・マネージャー構成
 パラメーター 304

stat_heap_sz
 データベース構成パラメーター
 289

stmtheap
 データベース構成パラメーター
 289

STOP DATABASE MANAGER 582

strdel
 プリコンパイル / バインド・オプション 179, 479

striptblanks 353, 447

striptnulls 353, 447

svcname
 データベース・マネージャ構成
 パラメーター 304

syncpoint
 プリコンパイル・オプション
 480

syntax
 プリコンパイル・オプション
 480

SYSADM_GROUP
 データベース・マネージャ構成
 パラメーター 304

sysctrl_group
 データベース・マネージャ構成
 パラメーター 304

sysmaint_group
 データベース・マネージャ構成
 パラメーター 304

T

target
 プリコンパイル・オプション
 480

TCP/IP ノード
 アンカタログ 595

TERMINATE 586
 注意、使用上の 586

territory
 データベース構成パラメーター
 289

text
 プリコンパイル / バインド・オプション 179, 481

timeformat 351, 444

timestampformat 352, 445

tm_database
 データベース・マネージャ構成
 パラメーター 304

totalreespace 440

tpname
 データベース・マネージャ構成
 パラメーター 304

tp_mon_name
 データベース・マネージャ構成
 パラメーター 304

tracefile 42

transform group
 プリコンパイル / バインド・オプション 179, 481

trust_allclnts
 データベース・マネージャ構成
 パラメーター 305

trust_clntauth
 データベース・マネージャ構成
 パラメーター 305

TSM アーカイブ・イメージによる作業 5

tsm_mgmtclass
 データベース構成パラメーター
 289

tsm_nodename
 データベース構成パラメーター
 289

tsm_owner
 データベース構成パラメーター
 289

tsm_password
 データベース構成パラメーター
 290

U

udf_mem_sz
 データベース・マネージャ構成
 パラメーター 305

UNCATALOG DATABASE 587

UNCATALOG DCS
 DATABASE 589

UNCATALOG LDAP
 DATABASE 591

UNCATALOG LDAP NODE 593

UNCATALOG NODE 595

UNCATALOG ODBC DATA
 SOURCE 597

UPDATE ADMIN
 CONFIGURATION 599

UPDATE CLI
 CONFIGURATION 602

UPDATE COMMAND
 OPTIONS 604

UPDATE DATABASE
 CONFIGURATION 606

UPDATE DATABASE MANAGER
 CONFIGURATION 608

UPDATE HISTORY FILE 615

UPDATE LDAP NODE 610

UPDATE MONITOR
 SWITCHES 613

usedefaults 349, 441

userexit
 データベース構成パラメーター
 290

user_exit_status
 データベース構成パラメーター
 290

util_heap_sz
 データベース構成パラメーター
 290

V

V5 セマンティクスへの固有索引変換
 の準備 117

validate
 プリコンパイル / バインド・オプション 180, 482

version
 プリコンパイル・オプション
 483

W

wchartype
 プリコンパイル・オプション
 483

WHERE 文節
 解決、SELECT ステートメント内
 の未確定シンボルの 140

Windows NT フェールオーバー・ユ
 ーティリティのセットアップ 82

Z

zoneddecimal 448

[特殊文字]

!, シェル・コマンド 122

¥, 行継続文字 135

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

SC88-8518-00



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