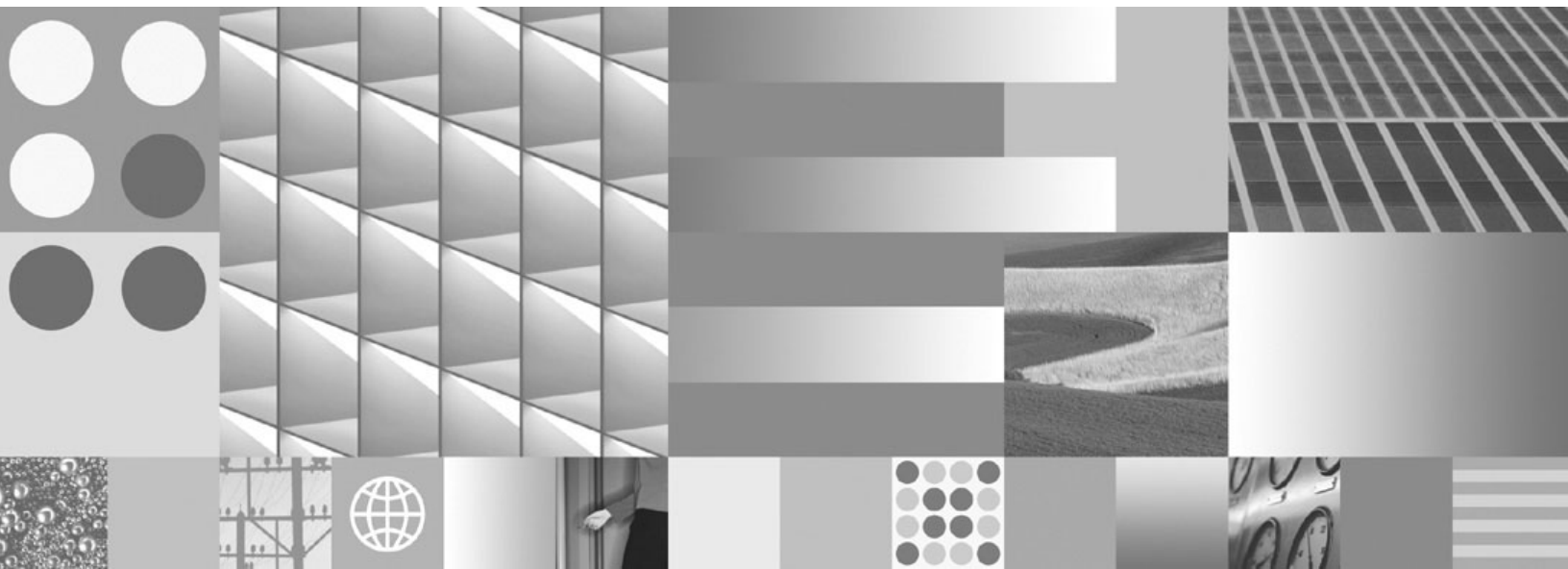


新機能





新機能

#### ご注意

本書および本書で紹介する製品をご使用になる前に、209 ページの『付録 C. 特記事項』に記載されている情報をお読みください。

#### 当版に関する特記事項

本書には、IBM の専有情報が含まれています。その情報は、使用許諾条件に基づき提供され、著作権により保護されています。本書に記載される情報には、いかなる製品の保証も含まれていません。また、本書で提供されるいかなる記述も、製品保証として解釈すべきではありません。

IBM 資料は、オンラインでご注文いただくことも、ご自分の国または地域の IBM 担当員を通してお求めいただくこともできます。

- オンラインで資料を注文するには、[www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order) にある IBM Publications Center をご利用ください。
- ご自分の国または地域の IBM 担当員を見つけるには、[www.ibm.com/planetwide](http://www.ibm.com/planetwide) にある IBM Directory of Worldwide Contacts をお調べください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

IBM 発行のマニュアルに関する情報のページ

<http://www.ibm.com/jp/manuals/>

こちらから、日本語版および英語版のオンライン・ライブラリーをご利用いただけます。また、マニュアルに関するご意見やご感想を、上記ページよりお送りください。今後の参考にさせていただきます。

(URL は、変更になる場合があります)

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

原 典： SC23-5869-00  
DB2 Version 9.5 for Linux, UNIX, and Windows  
What's New

発 行： 日本アイ・ビー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

---

# 目次

本書について	ix
本書の対象読者	ix
本書の構成	ix
<b>第 1 部 新規フィーチャーおよび機能</b>	<b>1</b>
<b>第 1 章 DB2 バージョン 9.5 のハイライト</b>	<b>3</b>
データベースではなくビジネスの管理	3
パフォーマンスとスケーラビリティの向上	7
サービスとしての情報	9
改良されたセキュリティおよび回復力	12
高可用性とデータ・リカバリー	14
さらにアジャイルな (機敏な) 開発	16
<b>第 2 章 DB2 Connect の機能拡張のサマリー</b>	<b>19</b>
DB2 Connect バージョン 9.5 の使用に影響する DB2 バージョン 9.1 フィックスパックの変更	21
<b>第 3 章 製品パッケージ化の機能拡張</b>	<b>23</b>
新しい DB2 クライアント製品ではデプロイメントが単純化されている (Windows)	23
コンポーネント名の変更	23
<b>第 4 章 管理の容易性の機能拡張</b>	<b>25</b>
データ・サーバーの管理および開発用インターフェースの追加	25
リアルタイム統計収集により確実に最新の統計が最適化に使用される	26
単純化されたマルチスレッド化アーキテクチャーにより総所有コスト (TCO) が減っている	27
複数のパーティション間のデータベース構成が単純化された	28
メモリー構成が単純化された	28
データ圧縮が拡張された	30
AUTOMATIC に設定して自動構成にできる構成パラメーターが増えた	30
データ再配分の機能拡張により容量が増えるコストとロード・バランシングのタスクが減っている	32
db2look コマンドはさらに多くのデータベース・オブジェクト用の DDL を生成する	36
SQL を介した DB2 管理コマンドへの拡張されたアクセス	36
拡張されたモニター機能	37
より柔軟で効率的になったライセンス交付のモニター	37
表スペースのスペース使用効率が向上している	38
ロック・タイムアウト診断が拡張された	38
統計プロファイルをリセットするための RUNSTATS ユーティリティ・オプションが追加された	39
<b>第 5 章 ワークロード管理の機能拡張</b>	<b>41</b>
ワークロード管理の改善による制御の向上	41
<b>第 6 章 セキュリティの向上</b>	<b>47</b>
トラステッド・コンテキストによるセキュリティの向上	47
監査機能のパフォーマンスと管理の機能拡張	49
ロールを使用した特権の管理の簡素化	51
ラベル・ベースのアクセス制御 (LBAC) の機能拡張によりセキュリティが向上している	52
<b>第 7 章 パフォーマンスの向上</b>	<b>53</b>
LOB 列を参照する照会の高速化	53
オプティミスティック並行性制御と更新検出の機能拡張により拡張が容易なロックの代替策が提供されている	54

索引の据え置きクリーンアップ・オプションを使用したより高速な MDC ロールアウト削除	55
索引構築の並列処理がデフォルトで使用可能にされる	56
拡張された OLAP 関数	56
照会最適マイザーが拡張されている	57
NO FILE SYSTEM CACHING デフォルトによるファイル・システムのキャッシュ・メモリーの削減	57
DB2 Spatial Extender の照会パフォーマンスの改善	58
EXPLAIN 可能ステートメントの追加	59

## 第 8 章 pureXML の機能拡張 . . . . . 61

XQuery 更新式を使用すると XML 文書の部分を変更できる	61
pureXML 用ロード・ユーティリティーのサポートの追加	62
pureXML アプリケーション処理のパフォーマンスが改善された	62
チェック制約機能が拡張された	62
トリガー処理で XML 文書の自動妥当性検査がサポートされている	63
XSLT サポートによる XML データから他の形式へのトランスフォームの使用可能化	63
SQL/XML および XQuery パラメーターを渡す際の柔軟性が増している	64
非 Unicode データベースに XML データを保管できる	64
パフォーマンス向上のために小規模な XML 文書が基本表の行に格納できる	65
XML 文書の妥当性検査の再実行を必要とせずに XML スキーマを更新できる	66
XQuery の upper-case および lower-case 関数はロケールをサポートする	66
XQuery 関数は日付と時間から構成要素を抽出し、日付と時間を調整する	67
XQuery キャスト可能式が値のキャストのテストをサポートする	67
発行関数が使いやすくなっている	67
アノテーション付き XML スキーマ分解での挿入の順序と再帰的スキーマの登録のサポート	68

## 第 9 章 アプリケーション開発の機能拡張 . . . . . 69

ID の長さ制限の増加	69
DB2 インストールに統合された PHP 拡張 (Linux、AIX、および Windows)	70
DB2 インストールに統合された Ruby on Rails フレームワーク・サポート (Linux、AIX、および Windows)	71
pureXML およびマルチバイト文字をサポートする Perl ドライバー	72
拡張された Visual Studio 2005 用の IBM Database Add-in	72
グローバル変数による SQL ステートメント間のデータ共有の改善	74
SET 変数は動的に準備できる実行可能ステートメントとなった	75
配列のサポートによるアプリケーション移植性の向上	76
10 進浮動小数点データ・タイプにより 10 進数データの正確性とパフォーマンスが向上している	76
JDBC および SQLJ のサポートの機能拡張	77
JDBC 2.0 および JDBC 3.0 のサポートが拡張されている	78
JDBC 4.0 サポートが追加された	84
IBM Data Server Provider for .NET は IBM Informix Dynamic Server、IBM UniData、および IBM UniVerse をサポートする	89
新しい DB2 サンプル・プログラムが追加された	90
CLIENT APPLNAME 特殊レジスターは CLP により自動的に設定される	92
DB2 Developer Workbench が名前変更されて拡張された	93
新しいスカラー関数でアプリケーションの移植が単純化された	95
追加された新しいビット単位のスカラー関数	96

## 第 10 章 高可用性、バックアップ、ロギング、およびリカバリーの機能拡張 . . . . . 97

新規システム・ストアード・プロシージャにより自動保守ポリシー構成が単純化される	97
新しい DB2 拡張コピー・サービス (ACS) API によりストレージ・ハードウェアとの統合が使用可能になっている	98
リカバリー・オブジェクトの除去が自動化されてリカバリー・オブジェクト管理が単純化されている	99
新しい DB2 高可用性インスタンス構成ユーティリティーによりクラスターの構成と管理が単純化されている	100
重複ログ制御ファイルによりデータベース・リカバリーの回復力が向上している	101
HADR のピア・ウィンドウにより、連鎖または複数の障害発生時のデータ損失のリスクを削減する	101
単一システム・ビュー・バックアップで即時に複数のデータベース・パーティションをバックアップおよびリストアできる	102
最小リカバリー時間へのロールフォワードが使用可能になった	103

スナップショット・バックアップによって高速化されたデータのバックアップおよびリストア	104
クラスター管理ソフトウェアの統合が使用可能	105
<b>第 11 章 インストール、マイグレーション、およびフィックスパック機能拡張</b>	<b>107</b>
IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component が DB2 インストール済み環境に統合される (Linux および AIX)	107
追加ツールのインストールを DB2 インストールから立ち上げられる	108
共通フィックスパックによりサーバー製品の更新が単純化されている	108
自動化されたフィックスパックのインストール後タスク (Linux および UNIX)	108
非ルート・ユーザーが DB2 製品のインストールと構成を実行できる (Linux および UNIX)	109
応答ファイルの新規キーワードの追加	109
非 DB2 インスタンス・マージ・モジュールが追加された	110
スタンドアロン・インスタンス・タイプによりサポートされているプラットフォーム間の整合性が向上している (Linux および UNIX)	111
Windows Vista のデプロイメントと使用がさらに簡単になった	111
<b>第 12 章 フェデレーションの機能強化</b>	<b>113</b>
アプリケーション開発がフェデレーション用に拡張された	113
セキュリティーがフェデレーション用に拡張された	114
構成がフェデレーション用に拡張された	115
<b>第 13 章 レプリケーションの機能拡張</b>	<b>117</b>
新規 CCD ターゲット・タイプにより UOW 表と CD 表の結合を回避する	117
DECFLOAT データ・タイプがレプリケーション用にサポートされる	117
<b>第 14 章 各国語の機能拡張</b>	<b>119</b>
言語認識照合によりデータの並び順にさらに多くのオプションが提供される	119
Unicode スtring・リテラルにより任意の Unicode 文字を指定できる	119
スカラー関数の文字ベースの処理は可変文字サイズをサポートする	120
Big5-HKSCS-Unicode 変換表による、Unicode データベース内の HKSCS データ保管のサポートの拡大	120
UPPER (UCASE) および LOWER (LCASE) スカラー関数はロケールをサポートする	121
<b>第 15 章 トラブルシューティングと問題判別の機能拡張</b>	<b>123</b>
データ収集ツールで予期しないエラーを追跡する	123
ロギング機能に SQL 管理ルーチンが追加された	124
ストレージ・キーによりメモリー・アクセス問題が検出される	124
オンライン・データ整合性検査が改善されている	124
索引データの整合性の問題の許容度が高くなっている	125
予期しないエラー時のデータベース回復力が改善されている	125
<b>第 2 部 変更事項</b>	<b>127</b>
<b>第 16 章 変更された機能</b>	<b>129</b>
管理の変更のサマリー	129
新規データベースのデフォルト・コード・ページは Unicode	129
今後、データベース監査は SECADM 権限を必要とするようになった	129
データ・コンプレッション・ディクショナリーが自動的に作成される	130
表書き込みイベント・モニターのターゲット表が変更された	131
いくつかのシステム・カタログ・ビューと組み込みルーチンの追加および変更	132
メモリー・ビジュアライザーによるメモリー最大消費量の表示	136
バックアップ・イメージの読み取りおよび書き込み特権の変更	137
DB2 ランチパッド上の「マイグレーション」ボタンの移動 (Windows)	138
表索引サイズの増加	138
表の切り捨てによって動的ステートメント・キャッシュが無効になる	139
REFRESH TABLE および SET INTEGRITY ステートメントの ALLOW NO ACCESS オプションについて並行性が改善された	139

データベース・セットアップ変更のサマリー	140
いくつかの構成パラメーターが変更された	140
いくつかのレジストリー変数および環境変数の変更	147
並行入出力および直接入出力はデフォルトで使用可能になる (AIX、Linux、Solaris、および Windows)	152
拡張セキュリティでは、ユーザーが DB2ADMNS または DB2USERS グループに属する必要がある (Windows Vista)	153
構成およびランタイム・データ・ファイルのデフォルト・ロケーションが変更された (Windows)	154
フィックスパックのインストール後に手動のステップの必要がない (Linux および UNIX)	154
いくつかの構成パラメーターは単純化されたメモリー構成によって影響を受ける	155
Information Integrator 製品 ID 値の変更	155
アプリケーション開発の変更のサマリー	156
変更された Java ルーチン用のデフォルト JDBC ドライバー	156
ResultSetMetaData が IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 に対して異なる値を返す	157
自動的に生成されるキーでのバッチ更新が SQLException の原因となる	158
unfenced ルーチン、unfenced ラッパー・ライブラリー、およびセキュリティ・プラグインはスレッド・セーフである必要がある (Linux および UNIX)	158
ID の長さ制限の増加	159
列およびアプリケーション・バッファーに、より大きいデフォルト値が必要	161
一部の CLI/ODBC アプリケーションは消費メモリーが多くなる可能性がある	161
db2Load および db2Import パラメーターが変更され、より長い ID をサポートするようになった	162
ID が長すぎる場合にエラーと警告がより早く返される	162
ダウン・レベルのユーティリティーおよび API はより長くなった ID を正しく扱うことができない可能性がある	163
非修飾 SYSFUN 関数が SYSIBM エラー・メッセージを返すことがある	163
特殊レジスターが長くなった	164
CLP およびシステム・コマンドの変更点のサマリー	164
コマンド行プロセッサ (CLP) の出力が変更された	164
REDISTRIBUTE DATABASE PARTITION GROUP コマンドが変更された	165
複数のデータベース・パーティションを同時にバックアップするバックアップ操作	166
db2audit コマンドが変更された	167
db2ckmig コマンドが変更された	169
db2mtrk コマンドが変更された	169
カスタマイズされたコールアウト・スクリプトが検索対象になる (Linux および UNIX)	170
OS プロセスおよびスレッドのリストが変更された (Linux および UNIX)	170
<b>第 17 章 推奨されない機能</b>	<b>173</b>
レジストリー変数および環境変数のいくつかは推奨されない	173
推奨されなくなった GET AUTHORIZATIONS コマンド	176
推奨されなくなった sqluadav API	176
推奨されなくなったいくつかのモニター・エレメント	177
ログ制御ファイル SQLOGCTL.LFH の名前変更とコピー	179
推奨されなくなった IMPORT コマンド・オプション CREATE および REPLACE_CREATE	180
推奨されなくなった XML Extender	181
推奨されなくなった静的データ・ストリームのスナップショット出力	181
推奨されなくなった Web Object Runtime Framework (WORF)	182
推奨されなくなった db2Import および db2Load API の piActionString データ構造	182
Network Information Services サポートは推奨されない (Linux および UNIX)	183
<b>第 18 章 廃止された機能</b>	<b>185</b>
拡張ストレージ (ESTORE) フィーチャー・サポートの廃止	185
アドレス・ウィンドウ操作拡張 (AWE) フィーチャー・サポートの廃止 (Windows)	186
db2icrt、db2ilist、および db2iupdt の -w オプションの廃止 (Linux および UNIX)	186
DB2 Web ツール・サポートの廃止	187
いくつかのレジストリー変数および環境変数の廃止	187
db2undgp コマンドの廃止	189
db2licm コマンドの -n オプションの廃止	189



CLISchema CLI キーワードの廃止 . . . . .	189
<b>第 19 章 DB2 バージョン 9.5 の使用に影響する DB2 バージョン 9.1 フィックスパック の変更 . . . . .</b>	<b>191</b>
<hr/>	
<b>第 3 部 付録 . . . . .</b>	<b>193</b>
<b>付録 A. ファイル・システム・キャッシング構成 . . . . .</b>	<b>195</b>
<b>付録 B. DB2 技術情報の概説 . . . . .</b>	<b>199</b>
DB2 テクニカル・ライブラリー (ハードコピーまたは PDF 形式) . . . . .	200
DB2 の印刷資料の注文方法 . . . . .	202
コマンド行プロセッサから SQL 状態ヘルプを表示する . . . . .	203
異なるバージョンの DB2 インフォメーション・センターへのアクセス . . . . .	203
DB2 インフォメーション・センターにおける特定の言語でのトピックの表示 . . . . .	204
コンピューターまたはイントラネット・サーバーにインストールされた DB2 インフォメーション・センターの更 新 . . . . .	205
DB2 チュートリアル . . . . .	207
DB2 トラブルシューティング情報 . . . . .	207
ご利用条件 . . . . .	208
<b>付録 C. 特記事項 . . . . .</b>	<b>209</b>
<b>索引 . . . . .</b>	<b>213</b>



---

## 本書について

本書には、DB2® Database for Linux®, UNIX®, and Windows® および DB2 Connect™ 製品の、バージョン 9.5 リリースに含まれる新機能および変更された機能について説明します。

---

## 本書の対象読者

本書は、DB2 Version 9.5 for Linux, UNIX, and Windows および DB2 Connect バージョン 9.5 で使用可能な機能拡張、およびそれらの製品のバージョン 9.1 とバージョン 9.5 との間の相違を素早く見つける必要のある、データベース管理者、アプリケーション・プログラマー、その他の DB2 データベース・ユーザーを対象としています。

本書は概説情報を提供するものであり、説明されているフィーチャーを使用するための詳細な指示は含んでいません。詳しい情報を知るには、提供されている参照資料を使用してください。

バージョン 9.5 で導入されたフィーチャーおよび機能拡張については、1 ページの『第 1 部 新規フィーチャーおよび機能』を参照してください。

バージョン 9.5 で変更された、推奨されなくなった、または廃止された機能については、127 ページの『第 2 部 変更事項』を参照してください。この情報は、バージョン 9.5 を使用する前に知る必要のある大切な変更内容を示しています。

DB2 Connect についての情報は、19 ページの『第 2 章 DB2 Connect の機能拡張のサマリー』を参照してください。

バージョン 9.1 ユーザーの場合、本書の他のセクションでは説明されていない、フィックスパック 3 (およびそれ以前のフィックスパック) で導入されてバージョン 9.5 にも適用されている変更内容のリストを、191 ページの『第 19 章 DB2 バージョン 9.5 の使用に影響する DB2 バージョン 9.1 フィックスパックの変更』で参照してください。

---

## 本書の構成

以下のトピックが含まれています。

### パート 1: 新しいフィーチャーおよび機能

#### 3 ページの『第 1 章 DB2 バージョン 9.5 のハイライト』

この章では、DB2 バージョン 9.5 に含まれる最も重要な新しいフィーチャーおよび機能拡張についての概要を示します。

#### 19 ページの『第 2 章 DB2 Connect の機能拡張のサマリー』

この章では、DB2 Connect の機能性に影響を与える DB2 バージョン 9.5 の機能拡張および変更点について説明します。

**23 ページの『第 3 章 製品パッケージ化の機能拡張』**

この章では、バージョン 9.5 で導入される製品パッケージの変更点について説明します。

**25 ページの『第 4 章 管理の容易性の機能拡張』**

この章では、データベースを管理する時間を削減するために役立つ新しいフィーチャーおよび機能拡張について説明します。

**41 ページの『第 5 章 ワークロード管理の機能拡張』**

この章では、以前のリリースで提供された既存のワークロード管理機能を拡張する、新しいワークロード管理フィーチャーについて説明します。

**47 ページの『第 6 章 セキュリティーの向上』**

この章では、機密データの保護および管理に役立つ新しいフィーチャーおよび機能拡張について説明します。

**53 ページの『第 7 章 パフォーマンスの向上』**

この章では、更新データにアクセスする際の最高パフォーマンスの実現に寄与する新しいフィーチャーおよび機能拡張について説明します。

**61 ページの『第 8 章 pureXML の機能拡張』**

この章では、新しい pureXML™ のフィーチャーおよび機能拡張について説明します。

**69 ページの『第 9 章 アプリケーション開発の機能拡張』**

この章では、アプリケーション開発を簡素化し、アプリケーションの移植性を向上し、さらにアプリケーション・デプロイメントを容易にする新しいフィーチャーおよび機能拡張について説明します。

**97 ページの『第 10 章 高可用性、バックアップ、ロギング、およびリカバリーの機能拡張』**

この章では、ユーザーがデータを常に使用できるようにするために役立つ新しいフィーチャーおよび機能拡張について説明します。

**107 ページの『第 11 章 インストール、マイグレーション、およびフィックスパック機能拡張』**

この章では、DB2 製品のデプロイをより高速に、また製品の保守をより簡単にする新しいフィーチャーおよび機能拡張について説明します。

**113 ページの『第 12 章 フェデレーションの機能強化』**

この章では、フェデレーテッド・データベースの新しいフィーチャーおよび機能拡張について説明します。

**117 ページの『第 13 章 レプリケーションの機能拡張』**

この章では、SQL レプリケーションの新しいフィーチャーおよび機能拡張について説明します。

**119 ページの『第 14 章 各国語の機能拡張』**

この章では、複数の各国語を扱うデータおよびデータベース・アプリケーションでの作業をより簡単にする新しいフィーチャーおよび機能拡張について説明します。

**123 ページの『第 15 章 トラブルシューティングと問題判別の機能拡張』**

この章では、問題が発生した場合に診断情報を生成するために使用できる新しいフィーチャーおよび機能拡張について説明します。

## パート 2: 変更事項

### 129 ページの『第 16 章 変更された機能』

この章では、データベースのセットアップ、データベース管理、アプリケーション開発、および CLP コマンドとシステム・コマンドに関連した変更を含む、既存の DB2 機能に対する変更点について説明します。

### 173 ページの『第 17 章 推奨されない機能』

この章では、推奨されない機能をリストします。これは、以前はサポートされていたもののすでに推奨されなくなっており、将来のリリースで除去される可能性がある特定の機能またはフィーチャーです。

### 185 ページの『第 18 章 廃止された機能』

この章では、バージョン 9.5 でサポートされていないフィーチャーおよび機能をリストします。

### 191 ページの『第 19 章 DB2 バージョン 9.5 の使用に影響する DB2 バージョン 9.1 フィックスパックの変更』

この章では、本書の他のセクションでは説明されていない、バージョン 9.1 フィックスパック 3 (およびそれ以前のフィックスパック) の一部として追加または変更されてバージョン 9.5 にも適用されるフィーチャーおよび機能について説明します。

## パート 3: 付録

### ファイル・システム・キャッシング構成

この付録では、バージョン 9.5 でサポートされるファイル・システム・キャッシング構成についての補足情報を示します。

### DB2 技術情報の概説

この付録では、DB2 データベース・システムのための最新資料にアクセスして使用方法を説明します。

### 特記事項

この付録では、DB2 データベース製品およびその資料の使用に関連した法律上の要件および制約について説明します。



---

## 第 1 部 新規フィーチャーおよび機能

このセクションでは、DB2 バージョン 9.5 for Linux, UNIX, and Windows で使用可能な新規フィーチャーおよび機能を説明しています。

### 3 ページの『第 1 章 DB2 バージョン 9.5 のハイライト』

この章では、DB2 バージョン 9.5 に含まれる最も重要な新しいフィーチャーおよび機能拡張についての概要を示します。

### 19 ページの『第 2 章 DB2 Connect の機能拡張のサマリー』

この章では、DB2 Connect の機能性に影響を与える DB2 バージョン 9.5 の機能拡張および変更点について説明します。

### 23 ページの『第 3 章 製品パッケージ化の機能拡張』

この章では、バージョン 9.5 で導入される製品パッケージの変更点について説明します。

### 25 ページの『第 4 章 管理の容易性の機能拡張』

この章では、データベースを管理する時間を削減するために役立つ新しいフィーチャーおよび機能拡張について説明します。

### 41 ページの『第 5 章 ワークロード管理の機能拡張』

この章では、以前のリリースで提供された既存のワークロード管理機能を拡張する、新しいワークロード管理フィーチャーについて説明します。

### 47 ページの『第 6 章 セキュリティーの向上』

この章では、機密データの保護および管理に役立つ新しいフィーチャーおよび機能拡張について説明します。

### 53 ページの『第 7 章 パフォーマンスの向上』

この章では、更新データにアクセスする際の最高パフォーマンスの実現に寄与する新しいフィーチャーおよび機能拡張について説明します。

### 61 ページの『第 8 章 pureXML の機能拡張』

この章では、新しい pureXML のフィーチャーおよび機能拡張について説明します。

### 69 ページの『第 9 章 アプリケーション開発の機能拡張』

この章では、アプリケーション開発を簡素化し、アプリケーションの移植性を向上し、さらにアプリケーション・デプロイメントを容易にする新しいフィーチャーおよび機能拡張について説明します。

### 97 ページの『第 10 章 高可用性、バックアップ、ロギング、およびリカバリーの機能拡張』

この章では、ユーザーがデータを常に使用できるようにするために役立つ新しいフィーチャーおよび機能拡張について説明します。

### 107 ページの『第 11 章 インストール、マイグレーション、およびフィックスパック機能拡張』

この章では、DB2 製品のデプロイをより高速に、また製品の保守をより簡単にする新しいフィーチャーおよび機能拡張について説明します。

**113 ページの『第 12 章 フェデレーションの機能強化』**

この章では、フェデレーテッド・データベースの新しいフィーチャーおよび機能拡張について説明します。

**117 ページの『第 13 章 レプリケーションの機能拡張』**

この章では、SQL レプリケーションの新しいフィーチャーおよび機能拡張について説明します。

**119 ページの『第 14 章 各国語の機能拡張』**

この章では、複数の各国語を扱うデータおよびデータベース・アプリケーションでの作業をより簡単にする新しいフィーチャーおよび機能拡張について説明します。

**123 ページの『第 15 章 トラブルシューティングと問題判別の機能拡張』**

この章では、問題が発生した場合に診断情報を生成するために使用できる新しいフィーチャーおよび機能拡張について説明します。



---

## 第 1 章 DB2 バージョン 9.5 のハイライト

DB2 バージョン 9.5 for Linux, UNIX, and Windows は、組織全体から収集されたビジネス・データを統合すること、コストを削減すること、ビジネス上の価値を創出すること、企業の貴重な情報資産のためにセキュアで弾力的なシステムを提供することなど、ビジネスのさまざまな要件と取り組むための重要な新しいフィーチャーおよび機能拡張を提供します。

---

### データベースではなくビジネスの管理

バージョン 9.5 では、データ・サーバーの管理と調整に必要な時間を削減するオートノミック機能拡張と、アプリケーションをより短時間でセットアップしてデプロイできるインストール機能拡張が導入されています。

バージョン 9.5 には、以下の主要な管理の容易性フィーチャーが組み込まれています。

#### データ・コンプレッションの機能拡張

データ・コンプレッション・ディクショナリーは、COMPRESS 属性を定義した表でのデータ取り込み操作時に自動的に作成できます。このディクショナリーをいつ作成するかについて考慮する必要はなくなり、圧縮を使用するパッケージ・アプリケーションの作成はさらに簡単になりました。詳しくは、30 ページの『データ圧縮が拡張された』を参照してください。

#### 新規ワークロード管理機能

バージョン 9.5 では、新規の総合セットのワークロード管理フィーチャーにより、データ・サーバーのワークロードを識別、管理、およびモニターすることができます。このフィーチャーは、本当の意味で DB2 データ・サーバーに統合された初めてのワークロード管理ソリューションを提供します。ID アサーションのサポートにより、複数層アプリケーション環境内の個々のユーザーまたはグループにワークロード管理を提供することができます。

詳しくは、41 ページの『ワークロード管理の改善による制御の向上』 および 47 ページの『トラステッド・コンテキストによるセキュリティーの向上』を参照してください。

#### マルチスレッド・アーキテクチャーの拡張機能

DB2 データ・サーバーは、すべてのプラットフォーム上でマルチスレッド・アーキテクチャーを使用するようになりました。これによりパフォーマンスは向上し、構成と最適化は単純化されます。さらに、メモリー管理の単純化により、エージェント・レベルのほとんどの構成パラメーターは削除され、残ったものは自動化されます。すべてのオペレーティング・システムにわたって一貫性のあるスレッド化アーキテクチャーをこのように使用することにより、データ・サーバーの総合的な複雑さと保守は削減されます。バージョン 9.5 より前には、IBM® はマルチスレッド・アーキテクチャーを Windows オペレーティング・システム上だけで提供していました。バージョン 9.5 では、他のオペレーティング・システム上でもマルチスレッド・

アーキテクチャーを活用することができます。詳しくは、27ページの『単純化されたマルチスレッド化アーキテクチャーにより総所有コスト (TCO) が減っている』を参照してください。

#### デプロイメントの改善

デプロイメントの改善により、DB2 データ・サーバーのインストールと保守のプロセスは単純化されています。バージョン 9.5 には、以下の主要なフィーチャーが組み込まれています。

- 新規の IBM Data Server Driver for ODBC, CLI, and .NET client は、Windows プラットフォーム上でのアプリケーション・デプロイメントを単純化します。このクライアントは、占有スペースが小さく、独立系ソフトウェア・ベンダー (ISV) によって再配布され、大企業にありがちな一括デプロイメントのシナリオでアプリケーション配布に使用されるように設計されています。詳しくは、23ページの『新しい DB2 クライアント製品ではデプロイメントが単純化されている (Windows)』を参照してください。
- フィックスパックの適用後に必要とされた以前の 2 つの手動ステップの db2iupdt コマンドと dasupdt コマンドの実行は、自動化されました。さらに、最初の接続時にバインディングが自動的に行われます。詳しくは、108ページの『自動化されたフィックスパックのインストール後タスク (Linux および UNIX)』を参照してください。
- 非 root ユーザーは、Linux および UNIX オペレーティング・システム上で管理タスクを実行できるようになりました。非 root ユーザーが実行できるタスクには、フィックスパックのインストール、適用、またはロールバック、インスタンスの構成、新規フィーチャーの追加、およびアンインストールがあります。詳しくは、109ページの『非ルート・ユーザーが DB2 製品のインストールと構成を実行できる (Linux および UNIX)』を参照してください。

#### パーティション・データベース・システムの管理の簡易化

バージョン 9.5 で使用可能な機能拡張により、パーティション・データベース・システムは管理しやすくなりました。バージョン 9.5 には、以下のパーティション・データベース・システムのフィーチャーが組み込まれています。

- 複数のパーティションにまたがるすべてのデータベース構成要素の単一ビューがあります。この新機能を使用すると、データベースがある任意のパーティションから単一の SQL ステートメントまたは単一の管理コマンドを実行して、すべてのデータベース・パーティションにわたってデータベース構成を更新またはリセットすることができます。詳しくは、28ページの『複数のパーティション間のデータベース構成が単純化された』を参照してください。
- BACKUP DATABASE コマンドは、マルチパーティション・データベースの全パーティションを一度でバックアップできるようになりました。詳しくは、102ページの『単一システム・ビュー・バックアップで即時に複数のデータベース・パーティションをバックアップおよびリストアできる』を参照してください。

#### 自動ストレージ管理の機能拡張

自動ストレージは、ディスクおよびファイル・システムをまたいでデータベ

ースのサイズを自動的に大きくします。これによって、データベース管理スペースのパフォーマンスと柔軟性は利用しながらも、ストレージ・コンテナを管理する必要がなくなります。バージョン 9.5 は、表スペースのサイズを正確な最高水準点 (HWM) まで削減できる、表スペース処理の機能拡張を導入しています。これにより、未使用のスペースを自動的に再利用することができます。詳しくは、38 ページの『表スペースのスペース使用効率が向上している』を参照してください。

#### 追加の自動構成パラメーター

バージョン 9.5 には、インスタンスまたはデータベースの停止と再始動を必要とせずに、データ・サーバーが自動的に処理するさらに多くのチューニング・パラメーターが組み込まれています。新規構成パラメーターについては、140 ページの『いくつかの構成パラメーターが変更された』を参照してください。メモリー割り振りを制御するパラメーターの機能拡張については、30 ページの『AUTOMATIC に設定して自動構成にできる構成パラメーターが増えた』を参照してください。

#### 自動保守の向上

自動保守を利用したいが、プロセスとポリシーに対する厳格な制御を維持する必要がある場合は、自動保守構成情報 (SYSPROC.AUTOMAINT\_GET\_POLICY および SYSPROC.AUTOMAINT\_GET\_POLICYFILE) の収集、および自動保守 (SYSPROC.AUTOMAINT\_SET\_POLICY および SYSPROC.AUTOMAINT\_SET\_POLICYFILE) の構成のための新規システム・ストアード・プロシージャを使用することができます。これらのプロシージャを使用して、以下の領域の自動保守に関する情報を構成および収集することができます。

- 保守時間枠
- 自動バックアップ
- 表および索引の自動再編成
- 自動の表 RUNSTATS 操作

詳しくは、97 ページの『新規システム・ストアード・プロシージャにより自動保守ポリシー構成が単純化される』を参照してください。

#### バージョン 9.1 からの関連機能拡張

バージョン 9.1 では、IBM はデータベースのインストールと保守をさらに簡単にすると、以下の管理の容易性およびインストールの機能拡張を導入しました。

- 適応性のあるセルフチューニング・メモリー割り振り機能を使用することにより、さらに簡単になったメモリー管理。セルフチューニング・メモリーは、ワークロード特性の重要な変更に対して動的に適応する構成を提供します。
- データベース作成時にデフォルトで有効になる自動統計収集。自動統計収集を有効にすると、DB2 データベース製品は自動的に RUNSTATS ユーティリティをバックグラウンドで実行して、正しい統計が確実に収集および保守されるようにします。
- パーティション・データベースの自動ストレージ・サポート。
- 表をドロップして再作成する必要なしに、いくつかの表の属性を変更することができる。

- より多くの表および索引の再編成機能を提供する新しいポリシー・オプション。
- データベース・スキーマをコピーしてモデル・スキーマを作成する機能。モデル・スキーマを確立したら、それをスキーマの新規バージョンを作成するためのテンプレートとして使用できます。
- 新しい管理 SQL ルーチンおよびビュー。管理ルーチンおよびビューは、DB2 データベース製品を SQL を介して管理するための、基本的で使いやすいプログラマチック・インターフェースを提供します。
- DB2 データベース・マネージャーによって自動的に調整できる動的な高速コミュニケーション・マネージャー (FCM) のバッファと新しい構成パラメーター。
- ライセンス・センターおよび db2licm コマンドの使用による製品ライセンス管理の容易化。
- 同一コンピューター上に複数の DB2 バージョンおよびフィックスパックをインストールする機能。
- 対話なしで DB2 データベース製品をセットアップできる、新規の応答ファイル・キーワード。
- DB2 Runtime Client のライセンス交付の変更。無償で配布できるようになりました。

#### 関連概念

パーティションおよびクラスタリングのガイドの『高速コミュニケーション・マネージャー (Linux および UNIX)』

パーティションおよびクラスタリングのガイドの『高速コミュニケーション・マネージャー (Windows)』

DB2 サーバー機能 概説およびインストールの『応答ファイルによるインストールの基礎』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『自動統計収集』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『セルフチューニング・メモリー』

IBM データ・サーバー・クライアント機能 概説およびインストールの『IBM データ・サーバー・クライアントのタイプ』

データベース・パフォーマンスのチューニングの『自動再編成』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『自動ストレージ』

#### 関連タスク

データ移動ユーティリティのガイドおよびリファレンスの『スキーマのコピー』

#### 関連資料

SQL リファレンス 第 2 巻の『ALTER TABLE ステートメント』

管理ルーチンおよびビューの『サポートされている管理 SQL ルーチンとビュー』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『複数 DB2 コピーの概要』

---

## パフォーマンスとスケーラビリティの向上

バージョン 9.5 では、大量のデータへのアクセスおよびその更新時に最高のパフォーマンスを実現するために役立つ、パフォーマンスおよびスケーラビリティの機能拡張が導入されています。パフォーマンスおよびスケーラビリティにおける向上により、DB2 データ・サーバーは、あらゆる規模の企業に適した、非常に強力なデータ・サーバー・ソリューションであり続けています。

バージョン 9.5 には、以下の主要なパフォーマンスとスケーラビリティのフィーチャーが組み込まれています。

### 照会パフォーマンスと自動統計収集の向上

照会オプティマイザーおよび自動統計収集の機能拡張により、照会の効率とパフォーマンスが向上します。バージョン 9.5 には、以下の機能拡張が含まれています。

- リアルタイムの統計収集。これにより表統計は、最適化して照会を実行する必要があるときにはいつでも必ず使用できます。照会をコンパイラーにサブミットすると、オプティマイザーは、照会をコンパイルして実行する前に、リアルタイム統計を収集する必要があるかどうかを判別します。次いでコンパイラーは、収集された統計を使用して、照会の最適なアクセス・プランを生成します。リアルタイムの自己修正統計の機能強化により、オプティマイザーが照会のための最適なアクセス・プランを生成するために適量の統計を更新することができます。詳しくは、『26 ページの『リアルタイム統計収集により確実に最新の統計が最適化に使用される』』を参照してください。
- 自動的にリフレッシュされるニックネーム統計。ニックネーム統計は、ニックネーム統計 (NNSTAT) ストアド・プロシージャの実行によって最新に保つことができます。最新の統計へのアクセスにより、フェデレーテッド・サーバー上のオプティマイザーは、パフォーマンスを向上させる照会プランを、情報を得た上で選択することができます。詳しくは、『115 ページの『構成がフェデレーション用に拡張された』』を参照してください。
- 複合照会が最適化されています。詳しくは、『57 ページの『照会オプティマイザーが拡張されている』』を参照してください。

### LOB 管理パフォーマンスの向上

LOB 管理の機能拡張により、LOB データを戻す照会のパフォーマンスが向上します。バージョン 9.5 には、以下の機能拡張が含まれています。

- LOB データ・タイプへの参照を含む行データのブロッキング。結果セットに LOB データが含まれている場合、複数行のデータをまとめてブロックして、単一カーソル要求に対する結果セットとしてクライアントに戻すことができます。
- 動的データ・フォーマットのサポート。これによりサーバーは、LOB 値を効率的に戻すことができます。DB2 クライアントは、LOB 値を取得するために連続ストリーミングを使用するときに、動的データ・フォーマットを自動的に利用します。

詳しくは、『53 ページの『LOB 列を参照する照会の高速化』』を参照してください。

## より高速なマルチディメンション・クラスタリング表 (MDC) のロールアウト

MDC 表上でのロールアウト削除の完了後まで、レコード ID (RID) 索引のクリーンアップを据え置くことができるようになりました。RID 索引の据え置きクリーンアップは、ディメンション境界上で実行される削除操作の速度を大幅に向上させます。詳しくは、『55 ページの『索引の据え置きクリーンアップ・オプションを使用したより高速な MDC ロールアウト削除』』を参照してください。

## 並行性の向上

オプティミスティック・ロックのサポートは、データ保全性を保証しながら、ロックが維持される時間を制限することでリソースが使用できない時間を最小化します。オプティミスティック・ロック・プロトコルを使用して、サーバーは行を読み取った後に即時にロックを解放します。後の時点で行が更新されるときに、サーバーはその間に行が変更されていないことを検証します。詳しくは、『54 ページの『オプティミスティック並行性制御と更新検出の機能拡張により拡張が容易なロックの代替策が提供されている』』を参照してください。

## バージョン 9.1 からの関連機能拡張

バージョン 9.1 では、IBM は大量のデータの管理をさらに簡単にする、いくつかのパフォーマンスおよびスケーラビリティの機能拡張を導入しました。バージョン 9.1 で導入されたそれらの機能拡張には、以下のものがあります。

- データ行圧縮を使用して表データ・オブジェクトを圧縮することができる。
- 統計ビューの使用による照会のアクセス・プランの改良。
- カスタマイズされたスクリプトまたはプログラムの使用によるデータ・ロード機能の高速化。
- マテリアライズ照会表に対する照会の実行の拡張。
- より大きな RID。これを使用すると、オブジェクトごとのデータ・ページを多くでき、ページごとのレコード数も多くできます。
- 列を 64 個まで含められ、サイズを最大で 8 KB とすることができる索引キー。

### 関連概念

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『データベース管理スペース』

データベース・パフォーマンスのチューニングの『マテリアライズ照会表』

データベース・パフォーマンスのチューニングの『統計ビュー』

データ移動ユーティリティ ガイドおよびリファレンスの『カスタマイズしたアプリケーション (ユーザー出口) を使用したデータの移動』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『データ行圧縮』

### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『SQL および XML 制限』

---

## サービスとしての情報

バージョン 9.1 では、IBM は pureXML サポートを導入しました。これは DB2 データベース・システムをリレーショナルと XML の混成のデータ・サーバーにトランスフォームします。バージョン 9.5 はこれらの拡張機能に基づき、XML データ処理をさらに柔軟で、高速で、より一層信頼できるものとなるよう、サポートを拡張しています。

バージョン 9.5 には、以下の主要な pureXML フィーチャーが組み込まれています。

### 高速 XML データ・ロード

ハイパフォーマンスの load ユーティリティーにより、大量の XML データを DB2 表に迅速かつ効率的に挿入することができます。詳しくは、62 ページの『pureXML 用ロード・ユーティリティーのサポートの追加』を参照してください。

### pureXML のパフォーマンスの改善

pureXML フィーチャーの機能拡張は、実行時間を削減し、場合によっては XML データを処理するアプリケーションのリソース使用を削減します。パフォーマンス機能拡張には、SQL/XML および XQuery の両方におけるデータ操作の向上、XML データ作成の索引、照会コンパイラと最適化、および XML 文書ナビゲーションがあります。詳しくは、62 ページの『pureXML アプリケーション処理のパフォーマンスが改善された』を参照してください。

### 照会のパフォーマンス向上のためのサブ文書の更新

DB2 XQuery の機能拡張は、XML 文書の構造変更用にさらに多くのサポートを提供します。新規の XQuery 更新式により、新規の XML 文書を作成する代わりに既存の XML 文書の部分を変更できるので、サブ文書更新の効率は向上します。XQuery 更新式により、XML 文書内のノードの削除、挿入、置換、または名前変更を実行できます。詳しくは、61 ページの『XQuery 更新式を使用すると XML 文書の部分を変更できる』を参照してください。

### 保天性フィーチャーの XML サポート

バージョン 9.5 では、以下の保天性フィーチャーの XML サポートを導入しています。

- チェック制約機能により、処理前に情報の整合性を確保するために、XML 列に対する制約がある追加のオプションを指定することができます。詳しくは、62 ページの『チェック制約機能が拡張された』を参照してください。
- トリガー処理は、文書の現在の検証状態に基づき、登録済み XML スキーマに照らした XML 文書の自動検証をサポートします。詳しくは、63 ページの『トリガー処理で XML 文書の自動妥当性検査がサポートされている』を参照してください。

### XML から HTML、プレーン・テキスト、または他の形式への変換の簡易化

eXtensible Stylesheet Language Transformation (XSLT) は、XML をトランスフォームするための最も一般的な方法です。pureXML の機能拡張により、組み込み XSLT サポートによる柔軟な XML のトランスフォーメーション

が可能になります。新規の XSLTRANSFORM 関数は、データベース内のある XML 文書を、HTML、プレーン・テキスト、または他の形式の XML に変換します。詳しくは、63 ページの『XSLT サポートによる XML データから他の形式へのトランスフォームの使用可能化』を参照してください。

#### スキーマ互換性管理および発展サポート

スキーマは時間の経過と共に発展します。バージョン 9.5 で使用可能な機能拡張により、発展バージョンの登録済みスキーマに照らして、以前に挿入した XML 文書と新規 XML 文書の両方を検証することができます。

UPDATE XMLSCHEMA コマンドおよび XSR\_UPDATE ストアド・プロシージャにより、XML スキーマ・リポジトリに登録済みの XML スキーマを変更することができます。詳しくは、66 ページの『XML 文書の妥当性検査の再実行を必要とせずに XML スキーマを更新できる』を参照してください。

#### 非 Unicode データベース・サポート

pureXML のフィーチャーは、非 Unicode データベースで使用可能になりました。新機能ではコード・ページ変換が管理されるため、Unicode データベースが必要ありません。新規の enable\_xmlchar 構成パラメーターは、SQL ストリング・データがクライアントのコード・ページからデータベースのコード・ページに変換され、次いで内部保管のために Unicode に変換される時、文字置換が発生しないようにします。詳しくは、64 ページの『非 Unicode データベースに XML データを保管できる』を参照してください。

#### SQL/XML および XQuery による開発生産性の向上

SQL/XML および XQuery の改善により、XML データに対する堅固で効率的な照会を作成するために、その両方の言語の能力を活用できるようになりました。バージョン 9.5 では、以下の改善が図られています。

- さらに柔軟性を高めるために、SQL/XML および XQuery の両方に対して、パラメーターの受け渡しは単純化されて拡張されました。詳しくは、64 ページの『SQL/XML および XQuery パラメーターを渡す際の柔軟性が増している』を参照してください。
- 新規のパブリッシング関数は、リレーショナル・データと XML とのマッピングに使用可能です。これらの関数は、既存の SQL/XML パブリッシング関数に必要とされるよりも少ないオプションを指定するだけですみます。詳しくは、67 ページの『発行関数が使いやすくなっている』を参照してください。
- 多くの既存の SQL/XML パブリッシング関数の構文は単純化されています。
- XQuery 言語フィーチャーには、タイプ・キャストの使用、upper-case および lower-case 関数の使用時のロケールの指定、日付と時間要素の抽出、および時間帯の調整のためのサポートが組み込まれるようになりました。詳しくは、67 ページの『XQuery キャスト可能式が値のキャストのテストをサポートする』、66 ページの『XQuery の upper-case および lower-case 関数はロケールをサポートする』、および 67 ページの『XQuery 関数は日付と時間から構成要素を抽出し、日付と時間を調整する』を参照してください。



## pureXML の分解の機能拡張

pureXML の分解が拡張され、挿入の順序と再帰的スキーマの登録がサポートされています。

- 新規の XML スキーマの注釈により、分解階層を指定して、XML 文書の内容は決定可能な順序でターゲット表の行に挿入されます。この機能拡張により、XML 文書の分解時に参照整合性制約が確実に順守されます。
- 再帰を含む XML スキーマを XML スキーマ・リポジトリ (XSR) 内に登録して、それを分解に対して使用可能にできるようになりました。

詳しくは、68 ページの『アノテーション付き XML スキーマ分解での挿入の順序と再帰的スキーマの登録のサポート』を参照してください。

## バージョン 9.1 からの関連機能拡張

バージョン 9.1 では、IBM は業界における最も強力な XML データ・サーバーを提供しました。pureXML サポートは、XML を、リレーショナル・データとは異なるネイティブな階層内に格納される新規データ・タイプとして処理します。XML とリレーショナル・データとのシームレスな統合は、アプリケーション開発を加速し、高度に最適化された XML 索引を使用することで検索パフォーマンスを向上させ、SQL と XQuery の両方を XML データの照会に使用できるように柔軟です。

バージョン 9.1 には、以下の pureXML フィーチャーが組み込まれています。

- DB2 データベース・システムとの統合。これには以下の機能に対するサポートが含まれます。
  - 新しい XML データ・タイプ。表の列の中に階層形式のまま整形 XML 文書を保管することをサポートします。
  - XQuery。XML データを照会するための機能プログラミング言語。
  - SQL ステートメントおよび SQL/XML 関数における XML データ・タイプ。
  - XML データの索引付け。
- XML データのアクセスおよび管理のための拡張された新しいツール。以下が含まれます。
  - Developer Workbench。これは XML 関数、XML データ・タイプ、および XML スキーマ登録をサポートします。
  - DB2 コマンド行プロセッサ (CLP)。これは XML データ・タイプをサポートします。
  - Explain 機能および Visual Explain GUI ツール。これは SQL/XML 関数および XQuery ステートメントをサポートします。
- アプリケーション開発のサポート。以下の項目が含まれます。
  - プログラミング言語の XML サポート。これにより、アプリケーションが XML データとリレーショナル・データの両方のアクセスおよび保管を行えるようになります。
  - SQL および外部プロシージャでの XML サポート。これにより、CREATE PROCEDURE パラメーター・シグニチャーにデータ・タイプ XML のパラメーターを含めることで、SQL および外部プロシージャに XML データを渡すことができるようになります。

### 関連概念

データベース・パフォーマンスのチューニングの『[Explain 機能](#)』

Visual Explain チュートリアル『[Visual Explain](#)』

pureXML ガイドの『[XML データ・タイプ](#)』

pureXML ガイドの『[XML データの照会](#)』

pureXML ガイドの『[XML データの索引付けの概要](#)』

#### 関連資料

SQL リファレンス 第 2 巻の『[CREATE PROCEDURE ステートメント](#)』

コマンド・リファレンスの『[コマンド行プロセッサのフィーチャー](#)』

#### 関連情報

XQuery リファレンスの『[XQuery の概要](#)』

---

## 改良されたセキュリティーおよび回復力

バージョン 9.5 は、データに対して安全で弾力性のある環境を提供することに貢献する新規フィーチャーおよび機能拡張を備えています。

IT セキュリティーは、今日の組織の主要な関心事です。機密データを確実に保護することがきわめて重大になっています。システムのセキュリティーを効率的に管理し、セキュリティー環境を迅速に分析し、データへのアクセスをモニターする能力が必要です。バージョン 9.1 で導入された機能拡張の上に構築されたバージョン 9.5 での新しいセキュリティー強化は、機密データがさらに適切に保護されることを保証します。

バージョン 9.5 には、以下の主要なセキュリティー・フィーチャーが含まれます。

### データベースのロールによる単純化されたセキュリティー管理

ロール は、1 つ以上の特権を一緒にグループ化するデータベース・オブジェクトです。セキュリティー管理者 (SECADM 権限を持つ) は、ロールをユーザー、グループ、PUBLIC、他のロール、またはトラステッド・コンテキストに割り当てることができます。ユーザーがロールのメンバーとなる時、ユーザーはそのロールに割り当てられたすべての特権を自動的に取得します。セキュリティー管理者がロール内のユーザーのメンバーシップを取り消すとき、ユーザーはそのロールに割り当てられたすべての特権を自動的に失います。ロールは、セキュリティー管理者が組織の構造を反映する方法でデータベースのアクセスを制御できるようにして (組織内の職務権限に直接マップするロールをデータベース内に作成できます)、特権の管理およびマネージメントを単純化します。詳しくは、51 ページの『[ロールを使用した特権の管理の簡素化](#)』を参照してください。

### トラステッド・コンテキストのサポート

トラステッド・コンテキストは、より高速でよりセキュアな 3 層アプリケーションを作成する方法を提供します。ユーザーの ID は、監査およびセキュリティー目的のために常に保存されます。セキュア接続が必要な場合、トラステッド・コンテキストでは新しい接続を取得する必要がないためにパフォーマンスが向上します。詳しくは、47 ページの『[トラステッド・コンテキストによるセキュリティーの向上](#)』を参照してください。

### ラベル・ベースのアクセス制御 (LBAC) の機能拡張

LBAC によって、個別の行および個別の列に対するアクセスが指定可能とな

るために、データ・アクセスに対する制御がより良好になります。セキュリティー・ラベルおよび免除の管理は単純化されました。これらの資格情報を個別ユーザー・レベルで管理する必要はなくなりました。それらをグループ・レベルまたはロールレベルでも管理することができるようになりました。詳しくは、52 ページの『ラベル・ベースのアクセス制御 (LBAC) の機能拡張によりセキュリティーが向上している』を参照してください。

### 監査機能の機能拡張

バージョン 9.5 の監査機能に対する主な機能拡張によって、監査をより良好に制御できるようになり、パフォーマンスおよび使用の容易さが著しく向上します。これらの機能拡張には、詳細な構成、新しい監査カテゴリー、別個のインスタンスおよびデータベース・ログ、および監査構成をカスタマイズするための新しい方法が含まれます。データベース・レベルで監査を管理することは、セキュリティー管理者 (SECADM 権限を保持する) だけの責務となりました。詳しくは、49 ページの『監査機能のパフォーマンスと管理の機能拡張』を参照してください。

### セキュリティー管理者権限レベルの強化

前に説明されたように、セキュリティー管理者はトラステッド・コンテキスト・オブジェクト、データベースのロール、および監査ポリシーを管理できるようになりました。セキュリティー管理者は、これらのオブジェクトを作成、変更、またはドロップすること、またはこれらのオブジェクトにコメントを付けることができるようになりました。詳しくは、47 ページの『トラステッド・コンテキストによるセキュリティーの向上』、51 ページの『ロールを使用した特権の管理の簡素化』、および 49 ページの『監査機能のパフォーマンスと管理の機能拡張』を参照してください。

## バージョン 9.1 からの関連機能拡張

バージョン 9.1 で、IBM は機密データを保護するために設計されたいくつかのフィーチャーを導入しました。バージョン 9.1 で導入されたこれらのフィーチャーには、以下のものがあります。

- LBAC を使用するデータ・アクセス制御のサポート
- Lightweight Directory Access Protocol (LDAP) を使用する認証およびグループ検索をサポートするセキュリティーのプラグイン・モジュール
- 新規のセキュリティー管理者権限レベル (SECADM)。情報資産へのアクセスに対する制御が向上し、機密データへのアクセスをモニターするための報告機能が向上しています。
- CREATE DATABASE コマンドの新しい **RESTRICTIVE** オプション。データベース特権に対する制御を強化するオプションです。

#### 関連概念

データベース・セキュリティー・ガイドの『セキュリティー管理者権限 (SECADM)』

データベース・セキュリティー・ガイドの『ラベル・ベースのアクセス制御 (LBAC) の概要』

データベース・セキュリティー・ガイドの『LDAP ベースの認証およびグループ参照のサポート』

#### 関連資料

---

## 高可用性とデータ・リカバリー

バージョン 9.5 では、重要なデータベース・アプリケーションをオンラインにして使用可能にしておくためのいくつかの機能拡張が導入されています。改善されたオートノミック・フィーチャー、柔軟性の向上、および停止期間の削減により、最小限のコストでアプリケーションを実行状態にしておけるようになりました。

バージョン 9.5 は、以下の主要な高可用性およびデータ・リカバリー・フィーチャーを組み込んでいます。

### 拡張オートノミック・フィーチャー

- 自動保守構成は単純化されています。4 つの新規システム・ストアード・プロシージャを使用して、自動保守ポリシー情報を収集し、自動保守ポリシーを構成することができます。詳しくは、97 ページの『新規システム・ストアード・プロシージャにより自動保守ポリシー構成が単純化される』を参照してください。
- リカバリー・オブジェクト管理は自動化されました。バックアップ・イメージ、ロード・コピー・イメージ、およびリカバリーに不要な古いログ・ファイルを自動的に削除するよう、DB2 データベース・マネージャーを構成できます。詳しくは、99 ページの『リカバリー・オブジェクトの除去が自動化されてリカバリー・オブジェクト管理が単純化されている』を参照してください。

### より簡単で高速のバックアップおよびリストア操作

- IBM Tivoli® Storage Management (TSM) などのストレージ管理ソフトウェアの統合には、以下の機能拡張が組み込まれています。
  - ストレージ管理ソフトウェア統合機能を使用可能です。新しい DB2 拡張コピー・サービス (ACS) アプリケーション・プログラミング・インターフェース (API) を使用すると、ストレージ・ハードウェアを使用してスナップショット・バックアップ操作を実行できます。詳しくは、98 ページの『新しい DB2 拡張コピー・サービス (ACS) API によりストレージ・ハードウェアとの統合が使用可能になっている』を参照してください。
  - バックアップおよびリストア操作は、スナップショット・バックアップを使用するとはるかに高速です。スナップショット・バックアップまたはリストア操作を実行すると、ストレージ・デバイスはバックアップまたはリストアのデータ・コピーの部分を実行します。データ・コピーの実行にストレージ・デバイスを使用できるので、バックアップおよびリストア操作がずっと高速になります。詳しくは、104 ページの『スナップショット・バックアップによって高速化されたデータのバックアップおよびリストア』を参照してください。
- 新しい単一システム・ビュー (SSV) バックアップを使用して、複数のデータベース・パーティションを即時にバックアップおよびリストアできるようになりました。詳しくは、102 ページの『単一システム・ビュー・バックアップで即時に複数のデータベース・パーティションをバックアップおよびリストアできる』を参照してください。

- 最小リカバリー時間へのロールフォワードは単純化されています。 **TO END OF BACKUP** 節を **ROLLFORWARD** コマンドと共に、または **DB2ROLLFORWARD\_END\_OF\_BACKUP** フラグを **db2Rollforward** API と共に使用して、パーティション・データベース内の全パーティションを最小リカバリー時間までロールフォワードできます。詳しくは、103 ページの『最小リカバリー時間へのロールフォワードが使用可能になった』を参照してください。

#### フェイルオーバーおよびリカバリー機能の向上

- フェイルオーバーは、HADR ピア・ウィンドウの使用によりさらに堅固になりました。新規の **hadr\_peer\_window** データベース構成パラメーターを使用して、DB2 高可用性災害時リカバリー (HADR) のプライマリーがスタンバイ・データベースとの接続を失った場合に、プライマリーとスタンバイ・データベースのペアが、ピア状態にあるかのように動作させることができます。このフィーチャーは、複数または連鎖の障害発生時のデータ損失のリスクを削減することができます。詳しくは、101 ページの『HADR のピア・ウィンドウにより、連鎖または複数の障害発生時のデータ損失のリスクを削減する』を参照してください。
- 重複ロギング制御ファイルの使用により、リカバリーはより回復力のあるものになりました。バージョン 9.1 では、データベース・マネージャーは **SQLOGCTL.LFH** という 1 つのログ制御ファイルを維持していました。バージョン 9.5 では、データベース・マネージャーは 2 つのログ制御ファイルのコピー (**SQLOGCTL.LFH.1** と **SQLOGCTL.LFH.2**) を維持します。ログ制御ファイルのコピーが 2 つあるので、障害発生時にデータが失われるリスクが減っています。詳しくは、101 ページの『重複ログ制御ファイルによりデータベース・リカバリーの回復力が向上している』を参照してください。

#### 単純化されたクラスター環境管理

- IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component バージョン 2.2 は、IBM Data Server on Linux and AIX® にバンドルされました。詳しくは、107 ページの『IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component が DB2 インストール済み環境に統合される (Linux および AIX)』を参照してください。
- クラスター管理ソフトウェア統合が使用可能です。新しい DB2 クラスター・マネージャーのアプリケーション・プログラミング・インターフェース (API) によって、クラスター環境を構成するための DB2 高可用性インスタンス構成ユーティリティー (**db2haicu**) などの IBM Data Server クラスター構成ツールを使用できます。詳しくは、105 ページの『クラスター管理ソフトウェアの統合が使用可能』を参照してください。

#### パーティション・データベースのより高速なオフライン再配分

バージョン 9.5 フィックスパック 1 から、以下の **REDISTRIBUTE DATABASE PARTITION GROUP** コマンドの改善により、システム容量の拡張シナリオは、より管理しやすく効率的になります。

- 新規コマンド・オプション (**COMPACT** および **STATISTICS USE PROFILE**) は、ユーザビリティとデータ再配布処理の制御を向上しています。

- REDISTRIBUTE コマンドは、表の再編成、索引の保守、空きディスク・スペースの再利用、およびデータベース統計の収集などのさまざまな操作を実行できるようになりました。REDISTRIBUTE コマンド内におけるこれらの操作の統合により、データベース・マネージャーが実行する表スキャンの数は削減されるので、パフォーマンスは向上します。内部アーキテクチャーの機能拡張も、エンドツーエンドの容量拡張操作の総合的なパフォーマンスに貢献します。
- データ移動アクティビティーのためのアクティブ・ログ・スペース所要量はなくなりました。つまりこれは、再配分ユーティリティーはアクティブ・ログのスペースがほとんどなくても実行可能であり、一回の再配分操作を複数の細分化された再配分操作に分割する必要がなくなったことを意味します。

詳しくは、32 ページの『データ再配分の機能拡張により容量が増えるコストとロード・バランシングのタスクが減っている』を参照してください。

---

## さらにアジャイルな (機敏な) 開発

バージョン 9.5 では、データベース・アプリケーション開発を単純化し、アプリケーションのポータビリティを改善し、アプリケーション・デプロイメントを容易にする新規フィーチャーおよび機能拡張を提供します。

バージョン 9.5 には、以下の主要なアプリケーション開発フィーチャーおよび機能拡張が含まれます。

### 10 進浮動小数点データ・タイプのサポート

バージョン 9.5 では DECFLOAT が導入されています。これは 10 進浮動小数点データ・タイプで、正確な 10 進値を扱うビジネス・アプリケーション (金融アプリケーションなど) にとって有用です。DECFLOAT は、DECIMAL データ・タイプの正確性と FLOAT データ・タイプのパフォーマンス上の長所の一部を合わせ持つため、通貨値を操作するアプリケーションで役立ちます。詳しくは、76 ページの『10 進浮動小数点データ・タイプにより 10 進数データの正確性とパフォーマンスが向上している』を参照してください。

### アプリケーション・ポータビリティの改善

バージョン 9.5 で改善されたアプリケーション・ポータビリティは、既存のコードとの互換性を向上させて、他のベンダーによるデータベースを使用するアプリケーションのマイグレーションを簡単にします。バージョン 9.5 には、以下の機能拡張が含まれています。

- プロシージャー内およびプロシージャーを呼び出すアプリケーション内での ARRAY データ・タイプのサポート。詳しくは、76 ページの『配列のサポートによるアプリケーション移植性の向上』を参照してください。
- グローバル変数のサポート。グローバル変数は、SQL ステートメントによってアクセスおよび変更可能な、名前付きメモリー変数です。バージョン 9.5 は、特定のセッションに関連付けられて、そのセッションに固有の値を含む、作成済みセッション・グローバル変数をサポートします。詳しくは、74 ページの『グローバル変数による SQL ステートメント間のデータ共有の改善』を参照してください。

- より大きな ID のサポート。多くの ID の最大長は、128 バイトに増加しました。詳しくは、69 ページの『ID の長さ制限の増加』を参照してください。
- 他のベンダーによる SQL ダイアレクトのサポート。バージョン 9.5 には、DECODE、NVL、LEAST、および GREATEST 関数が許容されています。詳しくは、95 ページの『新しいスカラー関数でアプリケーションの移植が単純化された』を参照してください。

### JDBC および SQLJ の機能拡張

バージョン 9.5 には、JDBC 4.0、JDBC 3.0、および以前の仕様内の機能に対するサポートが含まれます。詳しくは、78 ページの『JDBC 2.0 および JDBC 3.0 のサポートが拡張されている』 および 84 ページの『JDBC 4.0 サポートが追加された』を参照してください。

### Perl、PHP、および Ruby アプリケーション開発の機能拡張

Perl、Hypertext Preprocessor (PHP) 拡張、および Ruby on Rails フレームワークに対する改善されたサポートは、アプリケーション開発を単純化して、DB2 データベースに保管されたデータへのアクセスを改善します。バージョン 9.5 には、以下の機能拡張が含まれています。

- DB2 Perl ドライバーは、pureXML およびマルチバイト・ロケールをサポートするようになりました。これらの機能拡張は、XML データの保管と検索のため、および文字セット間での変換のために以前は必要とされたいくつかのプログラミング・ロジックを除去して、アプリケーション開発を単純化します。詳しくは、72 ページの『pureXML およびマルチバイト文字をサポートする Perl ドライバー』を参照してください。
- Ruby on Rails のドライバーおよびバイナリーは、プラットフォームのサブセットでは DB2 インストールの一部になり、素早いデプロイメントが可能になりました。ドライバーとバイナリーとを別々にダウンロードする必要はなくなりました。詳しくは、71 ページの『DB2 インストールに統合された Ruby on Rails フレームワーク・サポート (Linux、AIX、および Windows)』を参照してください。
- 以下の PHP 拡張は、プラットフォームのサブセット上で、DB2 インストールの一部として使用可能です。
  - IBM\_DB2、これは DB2 コール・レベル・インターフェース (CLI) ライブラリーを使用して、DB2 データベースに保管されたデータへの直接アクセスを提供する既存の拡張です。
  - PDO\_IBM、これは PHP Data Objects (PDO) インターフェースにより、DB2 データベースへのアクセスを提供する新規の拡張です。

詳しくは、70 ページの『DB2 インストールに統合された PHP 拡張 (Linux、AIX、および Windows)』を参照してください。

### バージョン 9.1 からの関連機能拡張

バージョン 9.1 で、IBM はさらにアジャイルな開発をサポートするいくつかのフィーチャーおよび改善を導入しました。バージョン 9.1 で導入されたそれらのフィーチャーおよび改善には、以下のものがあります。

- 以下のフィーチャーを含む、pureXML サポート。
  - pureXML のアプリケーション開発サポート

- XQuery 言語のサポート
- SQL ステートメントおよび SQL/XML 関数での XML のサポート
- SQL および外部プロシージャーでの XML タイプのサポート
- アノテーション付き XML スキーマ分解
- DB2 Driver for JDBC and SQLJ の拡張。これには以下のフィーチャーが含まれます。
  - ほとんどの JDBC メソッドに相当する関数を実行する SQLJ ステートメントのサポート
  - 多くの新規データ・タイプのサポート
  - DB2 for z/OS<sup>®</sup> データベース・サーバーへのトラステッド接続をサポートする新しい DB2 専用のメソッド。
  - 異機種混合プールおよび接続の再利用
- Developer Workbench。これには以下のフィーチャーが含まれます。
  - 統合されたストアド・プロシージャー・デバッグ機能
  - SQLJ アプリケーション開発のためのサポート
  - XML 関数のサポート
  - プロジェクトの共有を可能にする変更管理システムのサポート

#### 関連概念

pureXML ガイドの『アノテーション付き XML スキーマ分解』

pureXML ガイドの『XML データ・タイプ』

pureXML ガイドの『XML データの照会』

pureXML ガイドの『XML データの索引付けの概要』

データベース・アプリケーション 開発の基礎の『JDBC および SQLJ のサポートされるドライバー』

#### 関連資料

SQL リファレンス 第 2 巻の『CREATE PROCEDURE ステートメント』

#### 関連情報

XQuery リファレンスの『XQuery の概要』



---

## 第 2 章 DB2 Connect の機能拡張のサマリー

DB2 Connect は、e-business 用の IBM メインフレーム・データベース、および Linux、UNIX、および Windows オペレーティング・システム上で実行する他のアプリケーションへの高速で堅固な接続を提供します。バージョン 9.5 には、DB2 Connect の機能と能力に影響を与える機能拡張および変更点があります。

IBM の、DB2 for i5/OS™、DB2 for z/OS、および DB2 Server for VSE & VM は、世界有数の組織での最重要なデータを管理するシステムとして、引き続き選ばれています。これらのホストおよび i5/OS データベース製品はデータを管理しますが、このデータを Linux、UNIX、および Windows オペレーティング・システム上で実行しているアプリケーションと統合することが、大いに求められています。

DB2 Connect には、DB2 Connect Personal Edition、およびいくつかの DB2 Connect サーバー製品を含む、複数の接続ソリューションがあります。DB2 Connect サーバーは、複数のデスクトップ・クライアントおよび Web アプリケーションからホストまたは System i システム上で実行している DB2 データベース・サーバーへの接続を、集中させて管理する接続サーバーです。

DB2 Connect サーバーにより、ローカルおよびリモートのクライアント・アプリケーションは、DB2 データベースおよびホスト・システムを、以下を使用して作成、更新、および管理できます。

- 構造化照会言語 (SQL)
- 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)
- Open Database Connectivity (ODBC)
- Java™ Database Connectivity (JDBC)
- Structured Query Language for Java (SQLJ)
- DB2 コール・レベル・インターフェース (CLI)
- Microsoft® ActiveX Data Objects .NET (ADO .NET)

バージョン 9.5 には、DB2 Connect の機能と能力に影響を与える、以下のような機能拡張および変更点があります。

### 製品パッケージ化の機能拡張

- 23 ページの『新しい DB2 クライアント製品ではデプロイメントが単純化されている (Windows)』
- 23 ページの『コンポーネント名の変更』

### アプリケーション開発の機能拡張

- 70 ページの『DB2 インストールに統合された PHP 拡張 (Linux、AIX、および Windows)』
- 71 ページの『DB2 インストールに統合された Ruby on Rails フレームワーク・サポート (Linux、AIX、および Windows)』
- 72 ページの『pureXML およびマルチバイト文字をサポートする Perl ドライバー』

- 72 ページの『拡張された Visual Studio 2005 用の IBM Database Add-in』
- 76 ページの『10 進浮動小数点データ・タイプにより 10 進数データの正確性とパフォーマンスが向上している』
- 77 ページの『JDBC および SQLJ のサポートの機能拡張』
- 78 ページの『JDBC 2.0 および JDBC 3.0 のサポートが拡張されている』
- 84 ページの『JDBC 4.0 サポートが追加された』
- 89 ページの『IBM Data Server Provider for .NET は IBM Informix Dynamic Server、IBM UniData、および IBM UniVerse をサポートする』
- 92 ページの『CLIENT APPLNAME 特殊レジスターは CLP により自動的に設定される』

### フェデレーションの機能強化

- 113 ページの『アプリケーション開発がフェデレーション用に拡張された』
- 114 ページの『セキュリティーがフェデレーション用に拡張された』
- 115 ページの『構成がフェデレーション用に拡張された』

### レプリケーションの機能拡張

- 117 ページの『新規 CCD ターゲット・タイプにより UOW 表と CD 表の結合を回避する』
- 117 ページの『DECFLOAT データ・タイプがレプリケーション用にサポートされる』

### インストール、マイグレーション、およびフィックスパック機能拡張

- 108 ページの『自動化されたフィックスパックのインストール後タスク (Linux および UNIX)』
- 108 ページの『追加ツールのインストールを DB2 インストールから立ち上げられる』
- 109 ページの『非ルート・ユーザーが DB2 製品のインストールと構成を実行できる (Linux および UNIX)』
- 108 ページの『共通フィックスパックによりサーバー製品の更新が単純化されている』
- 109 ページの『応答ファイルの新規キーワードの追加』
- 110 ページの『非 DB2 インスタンス・マージ・モジュールが追加された』
- 107 ページの『IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component が DB2 インストール済み環境に統合される (Linux および AIX)』
- 111 ページの『Windows Vista のデプロイメントと使用がさらに簡単になった』

### トラブルシューティングと問題判別の機能拡張

- 123 ページの『データ収集ツールで予期しないエラーを追跡する』
- 124 ページの『ロギング機能に SQL 管理ルーチンが追加された』
- 124 ページの『ストレージ・キーによりメモリー・アクセス問題が検出される』
- 124 ページの『オンライン・データ整合性検査が改善されている』
- 125 ページの『索引データの整合性の問題の許容度が高くなっている』
- 125 ページの『予期しないエラー時のデータベース回復力が改善されている』

## アプリケーション開発の変更

- 157 ページの『ResultSetMetaData が IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 に対して異なる値を返す』
- 158 ページの『自動的に生成されるキーでのバッチ更新が SQLException の原因となる』
- 161 ページの『列およびアプリケーション・バッファに、より大きいデフォルト値が必要』
- 161 ページの『一部の CLI/ODBC アプリケーションは消費メモリーが多くなることもある』
- 164 ページの『コマンド行プロセッサ (CLP) の出力が変更された』
- 162 ページの『db2Load および db2Import パラメーターが変更され、より長い ID をサポートするようになった』
- 162 ページの『ID が長すぎる場合にエラーと警告がより早く返される』
- 163 ページの『ダウン・レベルのユーティリティーおよび API はより長くなった ID を正しく扱うことができない可能性がある』
- 163 ページの『非修飾 SYSFUN 関数が SYSIBM エラー・メッセージを返すことがある』
- 164 ページの『特殊レジスターが長くなった』

## CLP およびシステム・コマンドの変更点

- 170 ページの『カスタマイズされたコールアウト・スクリプトが検索対象になる (Linux および UNIX)』
- 170 ページの『OS プロセスおよびスレッドのリストが変更された (Linux および UNIX)』
- 191 ページの『第 19 章 DB2 バージョン 9.5 の使用に影響する DB2 バージョン 9.1 フィックスパックの変更』

## 推奨されない機能

- 181 ページの『推奨されなくなった静的データ・ストリームのスナップショット出力』

## 廃止された機能

- 189 ページの『CLISchema CLI キーワードの廃止』

---

## DB2 Connect バージョン 9.5 の使用に影響する DB2 バージョン 9.1 フィックスパックの変更

バージョン 9.1 フィックスパック 3 (およびこれ以前) には、DB2 Connect バージョン 9.5 の使用に影響するフィーチャーおよび機能の変更が含まれています。

### 詳細

バージョン 9.1 フィックスパック 3 以前のフィックスパックを適用しなかった場合、あるいはバージョン 9.1 が使用可能になって以来、ローカルのインフォメーション・センターを更新しなかった場合には、DB2 バージョン 9.5 の使用に影響する可能性のある変更について確認する必要があります。

## 解決方法

DB2 バージョン 9.1 フィックスパックに含まれている技術的な変更点に通じていない場合は、以下のトピックを参照してください。フィックスパックは累積的であるため、以前のフィックスパックの変更および機能性のすべてが含まれています。

### DB2 バージョン 9.1 フィックスパック 1

フィックスパック 1 には、既存の機能に対する以下の変更点が含まれています。

- 連続ストリーミングを使用する JDBC アプリケーションでの LOB または XML 値の使用可能性が変更された
- 製品 ID の修正レベルには英数字が使用できる

フィックスパック 1 には、以下の機能拡張が含まれています。

- DB2Binder ユーティリティに 2 つの新規オプションが含まれる

### DB2 バージョン 9.1 フィックスパック 2

フィックスパック 2 には、フィックスパック 1 の機能と以下の機能拡張が含まれます。

- C および C++ 組み込み SQL アプリケーションにおける BINARY、VARBINARY、および DECFLOAT データ・タイプのサポートが追加された
- .NET Framework 2.0 に対する DB2 .NET Data Provider の機能拡張とサポート
- Visual Studio 2005 用の IBM Database Add-In の機能拡張
- Solaris オペレーティング環境での IBM Software Development Kit (SDK) for Java 5.x のサポートの追加
- Windows Vista サポートの追加 (Windows)

### DB2 バージョン 9.1 フィックスパック 3

フィックスパック 3 には、フィックスパック 2 の機能と以下の機能拡張が含まれます。

- パスワード変更サポートの追加 (Linux)
- JDBC および SQLJ の機能拡張

---

## 第 3 章 製品パッケージ化の機能拡張

IBM のデータ・サーバー製品の継続的な進歩にともない、DB2 コンポーネントのパッケージおよびコンポーネント名が変わりました。

バージョン 9.5 において、IBM は、利用可能な DB2 データベース製品のラインナップを更新し、マーケットのニーズに応える新しいフィーチャーをいくつか追加しました。これらの製品の説明と、関連するライセンス交付情報およびマーケティング情報に関しては、DB2 Database for Linux, UNIX, and Windows のホーム・ページ <http://www.ibm.com/db2/9> を参照してください。

---

### 新しい DB2 クライアント製品ではデプロイメントが単純化されている (Windows)

新しい IBM Data Server Driver for ODBC, CLI, and .NET クライアントでは、ODBC ドライバー、CLI ドライバー、OLE DB ドライバー、または IBM Data Server Provider for .NET を使用する Windows ベースのアプリケーションから DB2 サーバーへのアクセスの提供が容易になりました。

新しい IBM Data Server Driver for ODBC, CLI, and .NET クライアントでは、Windows プラットフォーム上でのアプリケーションのデプロイメントが単純化されます。このクライアントは、占有スペースが小さく、独立系ソフトウェア・ベンダー (ISV) によって再配布され、大企業にありがちな一括デプロイメントのシナリオでアプリケーション配布に使用されるように設計されています。

インストール中のドライバーの登録と構成、およびアンインストール中のドライバーの登録解除は、DB2 インストール・プログラムによって自動的に処理されます。

Linux および UNIX オペレーティング・システムの場合は、引き続き tar 形式で IBM Data Server Driver for ODBC and CLI を入手できます。

#### 関連概念

IBM データ・サーバー・クライアント機能 概説およびインストールの『IBM データ・サーバー・クライアントのセットアップの概要』

コール・レベル・インターフェース ガイドおよびリファレンス 第 1 巻の『IBM Data Server CLI と ODBC ドライバー』

#### 関連タスク

IBM データ・サーバー・クライアント機能 概説およびインストールの『IBM データ・サーバー・クライアントのインストール (Windows)』

ADO.NET および OLE DB アプリケーションの開発の『.NET アプリケーションのデプロイ (Windows)』

---

### コンポーネント名の変更

IBM データ・サーバーの継続的な進歩にともない、関連するコンポーネントおよびコンポーネント名が変わりました。

以下の表は、バージョン 9.5 で名前が変更された製品コンポーネントの一覧です。

表 1. DB2 製品コンポーネントの新しい名前

バージョン 9.1 のコンポーネント名	バージョン 9.5 のコンポーネント名
DB2 クライアント	IBM Data Server Client
DB2 Developer Workbench	データ・サーバー開発者ツール
DB2 Runtime Client	IBM Data Server Runtime Client
IBM DB2 Driver for JDBC and SQLJ	IBM Data Server Driver for JDBC and SQLJ
IBM DB2 Driver for ODBC and CLI	IBM Data Server Driver for ODBC and CLI

#### 関連概念

ADO.NET および OLE DB アプリケーションの開発の『Visual Studio での DB2 統合』

IBM データ・サーバー・クライアント機能 概説およびインストールの『IBM データ・サーバー・クライアントのセットアップの概要』

---

## 第 4 章 管理の容易性の機能拡張

バージョン 9.5 では、一般的なタスクおよび重大なタスクの管理が容易になりました。より多くのタスクが自動化され、一部のタスクは統合され、多くのプロセスはより少ない手順で完了できるようになりました。

このセクションでは、バージョン 9.5 で導入されている、DB2 データ・サーバーを容易に管理するための機能拡張について詳しく説明します。

---

### データ・サーバーの管理および開発用インターフェースの追加

データ・サーバーとデータベースをより効率的かつ効果的に管理およびモニターする助けとして、2 つの新規データ・サーバー・インターフェースを使用できます。

データ・サーバー管理者ツールは、IBM データ・サーバーをモニターするための、用途が広い Web インターフェースを提供します。データベースの健康状態と可用性のモニター、ダッシュボードの表示、アラートの表示、および専門家の推奨による問題のトラブルシューティングを行うことができます。さらに、Q レプリケーションとイベント・パブリッシングのモニター、レプリケーション・ヘルス・レポートの生成、および基本的なレプリケーション操作の実行も行えます。データ・サーバー管理者ツールは、AIX、Linux (x86) および Linux (AMD64)、Windows XP および Windows 2003 (32 ビットおよび 64 ビット) の各プラットフォームにインストールできます。

データ・サーバー開発者ツールは、バージョン 9.1 に同梱の DB2 Developer Workbench に基づいた、多機能で堅固なユーザー・インターフェースで、データベース設計、開発、デプロイメント、および管理タスクを行う助けになります。それを使用すると、ルーチンの開発およびテスト、データ処理を中心とする Web サービスのデプロイ、SQL および XQuery 照会の作成と実行、およびデータベースまたはメモリー内のデータにアクセスする Java アプリケーションの開発を行うことができます。加えて、データベース・オブジェクトの作成と変更、特権の管理などのような基本的なデータベース管理タスクも実行できます。データ・サーバー開発者ツールは、Linux (x86) プラットフォームおよび Windows (32 ビットおよび 64 ビット) プラットフォームにインストールできます。詳しくは、93 ページの『DB2 Developer Workbench が名前変更されて拡張された』を参照してください。

コントロール・センターおよび関連ツール (コマンド・エディター、タスク・センターなど) は、データベース設計および作動データベース管理のための汎用ソリューションを提供するために現在でも使用できます。それらを使用すると、データベース・オブジェクトの作成および変更、データベースでのデータの出し入れ、ルーチンの保守の実行、データベース・リカバリー計画のセットアップと管理、データベース・スクリプトの管理、照会の最適化、およびアドバイザー・ベースのウィザードを使用したデータベース設計の最適化を行うことができます。

#### 関連概念

93 ページの『DB2 Developer Workbench が名前変更されて拡張された』

---

## リアルタイム統計収集により確実に最新の統計が最適化に使用される

バージョン 9.5 はリアルタイム統計収集を導入しているため、照会の最適化と実行に統計が必要な時点で必ず自動的に表統計が収集されます。リアルタイム自動統計は、新しい動的構成パラメーター `auto_stmt_stats` によって使用可能になります。

正確な最新のデータベース統計を使用すると、良質な照会実行計画を作成するのに役立ち、照会処理時間を短縮できます。バージョン 9.5 より前では、統計は手動で収集するか、周期的に自動化される統計収集によって収集することができました。

DB2 UDB バージョン 8.2 では自動統計収集が導入されました。この機能には、表のモニターと、表アクティビティーの量が多いために統計が変更された表に対する周期的な統計収集が含まれています。このバックグラウンド・プロセスは、固定された時間間隔で表アクティビティーの評価を実行しました。したがって、データが変更された時刻と新しい統計が収集された時刻の間にギャップが生じることがありました。

リアルタイム統計収集は、自動統計収集中のギャップを小さくします。コンパイラーに照会をサブミットすると、オプティマイザーは参照されている表の統計が正確かどうかを判別します。統計がないか、最後に統計が収集されてから表に大幅な変更が加えられている場合は、おそらくステートメント・コンパイル中に再び統計が収集されます。デフォルトでは、ステートメント・コンパイル時に統計を収集する時間は 5 秒に制限されます。統計の収集に 5 秒より長くかかる場合は、代わりにバックグラウンド要求が生成されます。この制限は、最適化プロファイルを使用して構成できます。場合によっては、データ・マネージャーおよび索引マネージャーが保守するメタデータに基づいて、ステートメント・コンパイル時に統計を作成できます。表に変更を加えた際に、統計を更新する即時アクションは必要ないが、表に対する大幅な変更である場合には、バックグラウンド要求が生成され、可能な限り即時に統計が収集されます。

### 関連概念

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『自動統計収集』

データベース・パフォーマンスのチューニングの『オプティマイザー・プロファイルとガイドラインの概要』

データベース・パフォーマンスのチューニングの『RTS 要求』

### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『catalogcache\_sz - カタログ・キャッシュ・サイズ構成パラメーター』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『auto\_maint - 自動保守構成パラメーター』



## 単純化されたマルチスレッド化アーキテクチャーにより総所有コスト (TCO) が減っている

現バージョンでは、すべてのプラットフォームで DB2 データ・サーバーにマルチスレッド化アーキテクチャーが備えられています。バージョン 9.5 より前では、UNIX および Linux オペレーティング・システム上で実行している DB2 データ・サーバーはプロセス・ベースのモデルを使用していました。このモデルでは、各エージェントが独自のプロセス中で実行されていました。

マルチスレッド化アーキテクチャーに移動することにより、以下の利点が得られます。

- 以下のフィーチャーと機能拡張により、ユーザビリティが拡張され、総所有コスト (TCO) が減ります。
  - すべてのプラットフォームで、動的セルフチューニング・メモリーが完全に使用可能になりました。
  - メモリー構成パラメーターがさらに自動的および動的になり、メモリーの構成が単純化されています。詳しくは、28 ページの『メモリー構成が単純化された』を参照してください。
  - エージェントおよびプロセス・モデルの構成が単純化されています。これらの機能拡張によって、定期的な DBA 介入によりプロセス・モデル関連パラメーターを調整する必要がなくなり、構成に必要な時間や労力が少なくなります。新しい値を有効にするために、DB2 インスタンスをシャットダウンして再始動する必要はありません。
  - データベース・マネージャーは、新しい動的構成パラメーターを使用して、手操作による介入を行わずに自動的にシステムを調整できます。
  - 普通は、スレッド間を切り替えるコンテキストの方がプロセス間を切り替えるより速いので (アプリケーションとプラットフォームによって異なる)、パフォーマンスが向上します。
- ファイル・ハンドルなどのリソースを共有すると、同じデータベース・ファイル上で作動するすべてのエージェントが同じファイル・ハンドルを共有するので、効率が上がり、システム・リソースの節約になります。
- メモリー占有スペースが減ります。
- すべてのオペレーティング・システム間でスレッド化アーキテクチャーが整合するので、データ・サーバーの複雑さが緩和され、保守が軽減されます。

### 関連概念

データベース・パフォーマンスのチューニングの『DB2 プロセス・モデル』  
136 ページの『メモリー・ビジュアライザーによるメモリー最大消費量の表示』  
28 ページの『複数のパーティション間のデータベース構成が単純化された』  
28 ページの『メモリー構成が単純化された』  
140 ページの『いくつかの構成パラメーターが変更された』

---

## 複数のパーティション間のデータベース構成が単純化された

バージョン 9.5 は、複数パーティション間のすべてのデータベース構成エレメントの単一ビューを提供します。個々のデータベース・パーティションに対して UPDATE DATABASE CONFIGURATION または RESET DATABASE CONFIGURATION コマンドを呼び出さずに、すべてのデータベース・パーティション間でデータベース構成の更新やリセットを行うことができます。db2\_all コマンドを使用する必要はなくなりました。

データベースのあるいずれかのパーティションから 1 つの SQL ステートメントまたは 1 つの管理コマンドのみ発行して、パーティション間のデータベース構成を更新できます。つまり、デフォルトでは、データベース構成を更新したりリセットしたりする動作が、ローカル・データベース・パーティション上からすべてのデータベース・パーティション上に変更されています。

バージョン 9.5 は、コマンド・スクリプトおよびアプリケーションの後方互換性を保つために、以下の方法を提供しています。

- 新しいレジストリー変数 **DB2\_UPDDBCFG\_SINGLE\_DBPARTITION**。このレジストリー変数を TRUE に設定してください。この代替方法は、ADMIN\_CMD プロシージャによって行う UPDATE DATABASE CONFIGURATION 要求や RESET DATABASE CONFIGURATION 要求には適用されないことに注意してください。
- UPDATE DATABASE CONFIGURATION コマンド、RESET DATABASE CONFIGURATION コマンド、および ADMIN\_CMD プロシージャの新しいオプション **dbpartitionnum**。

特定のパーティション上のデータベース構成の更新またはリセットを行うには、**dbpartitionnum** オプションと、UPDATE DATABASE CONFIGURATION または RESET DATABASE CONFIGURATION コマンドを指定します。

### 関連資料

コマンド・リファレンスの『RESET DATABASE CONFIGURATION コマンド』

コマンド・リファレンスの『UPDATE DATABASE CONFIGURATION コマンド』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『システム環境変数』

管理ルーチンおよびビューの『ADMIN\_CMD プロシージャ - 管理コマンドの実行』

---

## メモリー構成が単純化された

旧リリースでは、ほとんどのパフォーマンス関連メモリー・パラメーター用にセルフチューニング・メモリーを使用可能にできました。しかし、DB2 データ・サーバーに必要なその他のメモリー・ヒープは依然として構成する必要がありました。現リリースでは、ほとんどのメモリー関連の構成パラメーターに対するデフォルトの AUTOMATIC 設定により、この構成タスクが単純化されています。

バージョン 9.5 のセルフチューニング・メモリーに対する機能拡張には、以下の利点があります。

- 1 つのパラメーター **instance\_memory** を使用して、データベース・マネージャーが専用および共有メモリー・ヒープから割り振ることができるすべてのメモリーを指定できます。新しい **appl\_memory** 構成パラメーターを使用して、アプリケーション要求にサービスを提供するために DB2 データベース・エージェントが割り振るアプリケーション・メモリーの最大量を制御できます。デフォルトでは、この値は **AUTOMATIC** に設定されます。この場合、データベース・パーティションによって割り振られる合計メモリー量が **instance\_memory** の限度内であれば、アプリケーション・メモリー要求は許可されます。
- ファンクション・メモリー専用のパラメーターを手動で調整する必要はありません。
- データベース・マネージャーの専用および共有メモリー・ヒープで現在消費されている合計メモリーの量を照会できます (メモリー・ビジュアライザーを使用)。**db2mtrk** コマンドを使用してヒープの使用量をモニターしたり、**ADMIN\_GET\_DBP\_MEM\_USAGE** 表関数を使用してメモリー消費量全体を照会したりすることもできます。
- 単純化されたアプリケーション・メモリー・モデルを使用する方が、必要な時点でアプリケーションの構成や調整を簡単に行えます。
- デフォルトの DB2 構成に必要な調整は非常に少なくなり、これは新しいインスタンスによって即時に得られる利点です。

#### 関連概念

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『セルフチューニング・メモリー』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『セルフチューニング・メモリーの操作に関する詳細および制限』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『セルフチューニング・メモリーの概要』

136 ページの『メモリー・ビジュアライザーによるメモリー最大消費量の表示』

155 ページの『いくつかの構成パラメーターは単純化されたメモリー構成によって影響を受ける』

140 ページの『いくつかの構成パラメーターが変更された』

#### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『instance\_memory - 「インスタンス・メモリー」構成パラメーター』

コマンド・リファレンスの『db2mtrk - メモリー・トラッカー・コマンド』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『appl\_memory - アプリケーション・メモリー構成パラメーター』

管理ルーチンおよびビューの『ADMIN\_GET\_DBP\_MEM\_USAGE 表関数 - インスタンスの合計メモリー消費量の取得』

---

## データ圧縮が拡張された

ディクショナリー自動作成 (ADC) を使用できるようになりました。ADC により、オフラインの表の再編成を行う必然性が減り、再編成を行わなくても良い場合があります。データ圧縮用に使用できる表にデータを追加する際に、初めてしきい値 (デフォルトは約 1 から 2 MB) を超えると、データ・コンプレッション・ディクショナリーが自動的に作成されます。

このリリースでは、COMPRESS 属性を定義した表に対するデータの設定操作中に、データ・コンプレッション・ディクショナリーを自動的に作成できます。表にまだデータ・コンプレッション・ディクショナリーがない場合は、ADC でディクショナリーを作成して表に挿入できます。ディクショナリーの作成後に表に移動されるデータは圧縮されます。

データ・コンプレッション・ディクショナリーが作成される可能性のあるデータ表の設定操作には、INSERT、IMPORT INSERT、LOAD INSERT、および REDISTRIBUTE DATABASE PARTITION GROUP があります。LOAD REPLACE ユーティリティーには、KEEPDICTIONARY および RESETDICTIONARY キーワードを使用して、明示的にデータ・コンプレッション・ディクショナリーの管理を実行する機能が備えられました。既存のコンプレッション・ディクショナリーを維持できますが、既存の場合でも新しいコンプレッション・ディクショナリーを作成できるようになりました。

### 関連概念

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『表のスペース圧縮』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『(コンプレッション) ディクショナリーの自動作成 (ADC)』

---

## AUTOMATIC に設定して自動構成にできる構成パラメーターが増えた

バージョン 9.5 では、さらに多くの構成パラメーターを AUTOMATIC 設定に設定して、より多くの構成パラメーターをインスタンスまたはデータベースを停止および再始動する必要なく動的に構成できます。これらの構成パラメーターの設定によって、データベース・マネージャーはより多くのデータベース・チューニングを処理すること、およびシステム・ワークロード内の変更に対して自動的に反応することができます。

31 ページの表 2 は、AUTOMATIC 設定が許可されて、動的に構成可能な構成パラメーターをリストしています。AUTOMATIC 設定の意味はパラメーターごとに異なります。一般に、AUTOMATIC 設定は、データベース構成パラメーターがシステム・リソースに基づいて自動的に調整されることを意味します。これらのパラメーターは、新しいデータベースまたはインスタンスの作成中に、そしていくつかのデータベースおよびインスタンスのマイグレーション・シナリオで、デフォルトで使用可能になります。

表 2. 動的に構成可能で、AUTOMATIC に設定できる、追加の構成パラメーター

構成パラメーター名	説明	AUTOMATIC 設定の意味
<b>applheapsz</b>	<p>以前のリリースでは、これはアプリケーションのために作業中の個々のデータベース・エージェントごとに、消費可能なアプリケーション・メモリーの量を設定するものでした。</p> <p>バージョン 9.5 では、これはアプリケーション全体で消費できるアプリケーション・メモリーの合計量を設定します。同じアプリケーションに対して複数のエージェントが作業している場合、<b>applheapsz</b> はそれらすべてのエージェントによって消費されるアプリケーション・メモリーの合計を表します。</p> <p>DPF、コンセントレーター、または SMP 構成では、以前のリリースで使用されていた <b>applheapsz</b> 値は、AUTOMATIC 設定が使用されているのではない限り、以前のリリースと類似のワークロードにおいては増やす必要が生じる可能性があります。</p>	<p>AUTOMATIC 設定は、<b>appl_memory</b> 限度または <b>instance_memory</b> 限度のいずれかに達するまで、アプリケーション・ヒープ・サイズが必要に応じて増加できるようにします。</p>
<b>database_memory</b>	<p>これは、データベース共有メモリー領域用に予約されるメモリーの量を指定します。</p>	<p>AUTOMATIC 設定は、セルフチューニングを使用できるようにします。使用可能にされると、メモリー・チューナーはデータベース全体のメモリー所要量を判別し、現在のデータベース要件に応じて、データベース共有メモリーに割り振られるメモリー量を増加または減少させます。</p>
<b>dbheap</b>	<p>データベース・ヒープが使用する最大メモリーを決定します。</p>	<p>AUTOMATIC 設定は、<b>database_memory</b> 限度または <b>instance_memory</b> 限度のいずれかに達するまで、データベース・ヒープが必要に応じて増加できるようにします。</p>

表 2. 動的に構成可能で、AUTOMATIC に設定できる、追加の構成パラメーター (続き)

構成パラメーター名	説明	AUTOMATIC 設定の意味
<b>instance_memory</b>	これは、データベース・パーティションに割り振り可能なメモリの最大量を指定します。	AUTOMATIC 設定により、データベース・パーティションが活動化される時点 (db2start) で実際の値を計算できます。
<b>mon_heap_sz</b>	これは、データベース・システム・モニター・データに割り振るメモリ量を決定します。	AUTOMATIC 設定は、 <b>instance_memory</b> 限度に達するまで、モニター・ヒープが必要に応じて増加できるようにします。
<b>stat_heap_sz</b>	RUNSTATS コマンドを使用して統計を収集するときに使用されるヒープの最大サイズを示します。	AUTOMATIC 設定は、 <b>appl_memory</b> 限度または <b>instance_memory</b> 限度のいずれかに達するまで、統計ヒープ・サイズが必要に応じて増加できるようにします。
<b>stmtheap</b>	これは、SQL または XQuery ステートメントのコンパイル中に SQL または XQuery コンパイラーのワークスペースとして使用される、ステートメント・ヒープのサイズを指定します。	AUTOMATIC 設定は、 <b>appl_memory</b> 限度または <b>instance_memory</b> 限度のいずれかに達するまで、ステートメント・ヒープが必要に応じて増加できるようにします。

## database\_memory 構成パラメーターのために AUTOMATIC 設定を使用する例

**database\_memory** 構成パラメーターを AUTOMATIC 現行のデータベース要求が多く、かつシステム上に十分な空きメモリーがあるなら、データベース共有メモリーによってより多くのメモリーが消費されます。データベースのメモリー所要量が低下する場合、またはシステム上の空きメモリー量が過度に低下する場合は、データベース共有メモリーの一部が解放されます。

### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『構成パラメーターのサマリー』

## データ再配分の機能拡張により容量が増えるコストとロード・バランシングのタスクが減っている

バージョン 9.5 フィックスパック 1 からは、強化されたデータ再配分サービス提供ユーティリティーにより、データ再配分のパフォーマンスとユーザビリティが劇的に向上しています。このユーティリティーは、データ再配分に関連したすべてのタスクを実行でき、効率も向上しています。

容量が増えるシナリオで、ロード・バランシングまたはパフォーマンス調整中に、データ再配分を行うと、貴重な保守時間枠の時間、相当量の計画時間、不相应なログ・スペースや余分のコンテナ・スペースが必要になることがあります。

バージョン 9.5 フィックスパック 1 の前は、表の再編成や統計の収集など一部のデータ再配分タスクを別々に実行する必要がありました。現リリースでは、新しい `REDISTRIBUTE DATABASE PARTITION GROUP` コマンドを発行して一緒に実行できます。このコマンドは、これらのタスクを実行する方法とタイミングを制御するオプションを提供します。これらのタスクを組み合わせて自動化すると (組み合わせで自動化しないと手動タスクになる)、エラーが起きにくくなり、速度が上がり、効率も上がり、操作を制御できる割合が増えます。新しい `REDISTRIBUTE DATABASE PARTITION GROUP` コマンドを発行して実行できるタスクの例を以下に示します。

- データベース・パーティション・グループへのパーティションの追加
- データベース・パーティション・グループからのパーティションのドロップ
- 新しいパーティション・マップに基づいた、ターゲット・パーティションへのデータの移動
- フラグメント化の影響を大きく減らせる表データの再編成。すべての表に関するスキンのパフォーマンスを改善し、マルチディメンション・クラスター化されていない表のストレージ要件を減らせます。
- 索引の作成
- 統計の収集

旧リリースではデータは内部の挿入および削除操作によって移動されましたが、現リリースでは代わりにバルク・データが移動されます。そのため、表のスキニングやアクセスを行わなければならない回数が減り、パフォーマンスが向上しています。

挿入および削除操作ごとにログ・レコードが必要ではなくなりました。つまり、データ再配分の実行時にシステム中の大量のアクティブ・ログ・スペースやログ・アーカイブ・スペースを管理する必要がなくなりました。旧リリースでは、アクティブ・ログ・スペースやストレージ要件が大きいために 1 つのデータ再配分操作を複数の小さな再配分タスクに切断せざるを得なくなると、エンドツーエンドのデータ再配分操作を完了するのに必要な時間が長くなることがあったので、このような場合にこの新機能は特に利点があります。

`REDISTRIBUTE DATABASE PARTITION GROUP` コマンド・オプションでサポートされる以下の新しい動作により、データ再配分の制御をさらに細分化できます。

- データベース・パーティションの追加または除去

`REDISTRIBUTE` コマンドを使用して、データ再配分を実行中にデータベース・パーティションの追加や除去を行えるようになりました。旧リリースでは、データベース・パーティションの追加や除去を行うには、別個に `ALTER NODEGROUP` コマンドを発行する必要がありました。

- データ再配分処理の制御

新しい `TABLE` オプションを使用して、データ再配分操作の一部として、表が処理される順序を指定できます。例えば確実に、重要な表が最初に処理され、可能な限り即時にオンライン (読み取り専用) に戻るが、重要性の低い表は後で処理さ

れるようにすることができます。 **STOP** オプションを使用して将来の指定された時刻より後の追加の表の処理を休止することによって表に対するデータ再配分を一時停止し、後から **CONTINUE** オプションを使用してデータ再配分を完了することができるようになりました。

**注:** コマンドによる表の再配分の最中に停止イベントが起きた場合は、データ再配分ユーティリティーは指定された時刻で停止しません。停止時刻がチェックされるのは、表の前回の再配分から次の再配分までの間に限られます。

#### • 索引の保守

デフォルトでは、表集合のデータ再配分の実行時に、索引を更新できる場合は更新されます。さまざまな方法を使用して索引を更新できます。表に対する有効な索引を更新する方法を指定するには、以下のいずれかの新しいコマンド・オプションを使用します。

##### – INDEXING MODE AUTOSELECT

- これはデフォルトの動作です。このオプションを指定すると、データベース・マネージャーは索引の更新に関する最善の方法を選択します。

##### – INDEXING MODE INCREMENTAL

- このオプションは、索引が増分保守されるように指定します。

##### – INDEXING MODE REBUILD

- このオプションは、索引が完全に再構築されるように指定します。

##### – INDEXING MODE DEFERRED

- このオプションは、索引保守が据え置かれるように指定します。索引は無効とマークされます。表に対して他のユーティリティー操作が実行され、これらの操作の完了後にすべての索引が再作成されるようにしたいならば、このオプションは有用です。

#### • 再配分中のスペースの解放

データ再配分の実行と同時に表の再編成を実行して、ディスク・スペースを解放できます。この表の再編成により、再配分操作にパフォーマンス・コストが追加されることはありません。スペースの圧縮を実行するかどうかを指定するには、**COMPACT ON** または **COMPACT OFF** を指定します。

クラスタリング索引のある表の場合、**COMPACT ON** オプションを指定すると、クラスタリング索引は無視されます。

MDC 表の場合、**COMPACT ON** オプションを指定すると、個々のパーティション中のデータが再編成されます。しかし、再配分後の合計サイズは変わっていないように見えます。

#### • 統計の保守

統計プロファイルのある表に対してデータ再配分を実行すると、同時に表統計を収集してから、表の処理の完了後にこの統計を使用して DB2 カタログ中の表統計を更新できます。統計プロファイル中で指定されているすべての統計が収集されます。

統計を収集して更新するよう指定するには、新しいデフォルト・オプション **STATISTICS USE PROFILE** を指定します。



索引統計は、**INDEXING MODE REBUILD** オプションを指定する場合のみ収集されます。

統計プロファイルがない場合に、**INDEXING MODE REBUILD** を指定すると、統計は収集されません。

- **複数の表の並行再配分**

複数の表のデータを並行して再配分できます。データが並行して再配分される表の数を指定するには、**PARALLEL TABLE** オプションに値を指定します。この数の表が並行して再配分されます。このオプションを指定しない場合は、データベース・マネージャーが並行して再配分される表の最適な数を判別します。

CPU、ディスク、およびネットワークの使用率が低い場合、このオプションを使用すると、これらの 3 つのリソースの使用率を改善できます。CPU、ディスク・スペース、およびネットワークの使用率が低い場合に、**PARALLEL TABLE** オプションを指定していない際には、CPU、ディスク、およびネットワークの使用率が十分に大きくなるまで値を大きくしながらこのオプションを使用してみてください。再配分操作によりこれらのいずれかのリソースの使用率がすでに最大に達している場合は、表の並列処理を増やしても、パフォーマンスが向上する見込みはなく、低下する可能性があります。

- **メモリーの消費量**

データ再配分を実行する際には、ユーティリティー・ヒープ・メモリーが使用されます。各表のデータ再配分に使用されるユーティリティー・ヒープ・メモリーの 4 KB ページの数を指定するには、新しいオプション **DATA BUFFER data\_buffer\_sz** を指定します。このオプションを使用して、データ再配分操作のパフォーマンスを調整してください。

このオプションを指定しない場合は、デフォルトの動作として、表の処理の開始時に表ごとに使用できるユーティリティー・ヒープ・メモリーの 50% を使用します。

- **クラッシュ・リカバリーまたはロールフォワードとデータ再配分**

**REDISTRIBUTE DATABASE PARTITION GROUP** コマンドは、再配分される表に関連した表スペースをバックアップ・ペンディング状態にして、この表スペースのバックアップが取られるまで確実にこの表スペース中のデータに変更を加えることができないようにします。そして、影響を受けるすべての表スペースが、表スペース・レベルのバックアップかデータベース・レベルのバックアップにより確実にバックアップされるようにします。再配分操作によるロールフォワードにより、再配分されたすべての表が無効とマークされることになるので、再配分操作の完了時に個々の影響を受ける表スペースかデータベース全体のバックアップを取ることは重要です。**REDISTRIBUTE DATABASE PARTITION GROUP** コマンドは、ロールフォワード・リカバリー可能ではありません。詳しくは、**REDISTRIBUTE DATABASE PARTITION GROUP** のトピックを参照してください。

データ再分配タスクを整備して 1 つにしたユーティリティー、新しい **REDISTRIBUTE DATABASE PARTITION GROUP** コマンド・オプション、および新しい再配分ユーティリティーの動作は、データ再配分に要する時間を短縮し、エ

ラーを起きにくくし、システム・リソースの消費量を減らすのに役立ちます。結果として、データ再配分を実行する際の合計コストが減るので、他のビジネス・オペレーションのために時間とリソースが解放されます。

#### 関連概念

データベース・パフォーマンスのチューニングの『統計プロファイルを使用した統計の収集』

#### 関連資料

パーティションおよびクラスタリングのガイドの『REDISTRIBUTE DATABASE PARTITION GROUP コマンド』

---

## db2look コマンドはさらに多くのデータベース・オブジェクト用の DDL を生成する

データベース・オブジェクトの理解、移動、または複製の支援に使用できる db2look コマンドは、さらに多くのデータベース・オブジェクト用の DDL を生成します。

ロール、ワークロード管理オブジェクト、監査オブジェクト、トラステッド・コンテキスト、配列タイプ、およびグローバル変数などの、新規バージョン 9.5 オブジェクト用の DDL の生成に加え、db2look コマンドには、以下の機能拡張が含まれています。

- パーティション表用の表スペース DDL は、db2look **-d dbname -t tablename -l** を使用して生成することができます。
- 統計ビューの UPDATE 統計は、db2look **-d dbname -t tablename -m** を使用して生成することができます。
- 誤ったコマンド行パラメーターを指定したか、または **-t** オプションの使用時に、存在しない表名を指定した場合は、エラーが生成されます。
- データベース・オブジェクト用の DDL 生成の順序は、**-e** オプションを使用すると改善されます。

#### 関連資料

データ移動ユーティリティのガイドおよびリファレンスの『db2look - DB2 統計および DDL 抽出ツール・コマンド』

---

## SQL を介した DB2 管理コマンドへの拡張されたアクセス

バージョン 8 で導入された SQL 管理ルーチンは、バージョン 9.5 で拡張されて、より多くの管理タスクを含むようになりました。バージョン 9.5 では、新しい管理ビューも追加されました。

SQL 管理ルーチンおよびビューは、SQL を介して DB2 機能を使用するための、基本的で使いやすいプログラマチック・インターフェースを提供します。それらには、表の再編成、モニター・データのキャプチャーと取得、現行接続のアプリケーション ID の取得といった、さまざまな管理タスクを実行するための、組み込みビュー、表関数、プロシージャ、およびスカラー関数の集合が含まれています。

これらのルーチンおよびビューは、SQL ベースのアプリケーション、コマンド行、またはコマンド・スクリプトから呼び出すことができます。

新しい管理ビュー、ルーチン、およびプロシージャーに加えて、バージョン 9.5 には以下が含まれています。

- 拡張ワークロード管理機能
- データベースをモニターするための拡張されたサポート

既存の管理ルーチンに対するサポートを拡張するために、バージョン 9 のいくつかのルーチンが、バージョン 9.5 では新規のより包括的なルーチンまたはビューに置き換えられました。

新規のものを含む、サポートされるすべての管理 SQL ルーチンおよびビューのリストは、「管理ルーチンおよびビュー」の『サポートされる管理 SQL ルーチンおよび管理ビュー』を参照してください。

---

## 拡張されたモニター機能

バージョン 9.5 には、さまざまなプロセスの状況の確認、潜在的な問題点の監視、問題発生時の情報収集、またはパフォーマンスを改善するための構成設定値の調整に役立つ新しいモニター・エレメントが含まれています。

システム活動を追跡するための 80 を超えるモニター・エレメントが追加されました。これらの新しいエレメントのいくつかを使用して、リアルタイム統計や DB2 ワークロード管理フィーチャーなど、新しいバージョン 9.5 機能をモニターできます。その他のエレメントは、追加の表スペース活動のモニターなど、既存のモニター機能を拡張します。

以前のリリースからマイグレーションする場合で、表書き込みイベント・モニターを作成した場合は、新しいエレメントを使用するようにそれらを再作成する必要があります。詳しくは、「マイグレーション・ガイド」の『表書き込みイベント・モニターの再作成』を参照してください。

### 関連資料

システム・モニター ガイドおよびリファレンスの『データベース・システム・モニターのエレメント』

システム・モニター ガイドおよびリファレンスの『表スペース・アクティビティに関するモニター・エレメント』

ワークロード・マネージャー ガイドおよびリファレンスの『ワークロード管理に関するモニター・エレメント』

システム・モニター ガイドおよびリファレンスの『リアルタイム統計に関するモニター・エレメント』

システム・モニター ガイドおよびリファレンスの『オンライン分析処理 (OLAP) に関するモニター・エレメント』

---

## より柔軟で効率的になったライセンス交付のモニター

新しい表関数および管理ビューを使用すると、ライセンス交付情報の収集が容易になります。これらを使用すると、レポートを作成して、インストール済み DB2 製品およびフィーチャーのライセンス交付を管理できます。

ENV\_FEATURE\_INFO 管理ビューおよび ENV\_GET\_FEATURE\_INFO 表関数は、ライセンスが必須となるすべての使用可能なフィーチャーに関する情報を返します。フィーチャーごとに、フィーチャーの有効なライセンスがインストールされているかどうかに関する情報があります。

#### 関連資料

コマンド・リファレンスの『db2licm - ライセンス管理ツール・コマンド』

DB2 インストールおよび管理 概説 (Linux および Windows 版)の『DB2 ライセンス・ファイル』

管理ルーチンおよびビューの『ENV\_FEATURE\_INFO 管理ビュー - DB2 フィーチャーのライセンス情報を戻す』

---

## 表スペースのスペース使用効率が向上している

ALTER TABLESPACE ステートメントの新しいオプションを使用して、未使用の表スペースを再利用できます。バックアップとリストアに使用するストレージ・スペースを少なくすると、結果として時間を節約できます。

以下の表スペース処理の機能拡張により、ストレージ・スペースの使用量は、データベースに入っている情報の量にとって確実に最適になります。

- ALTER TABLESPACE ステートメントは、自動ストレージで管理される表スペースに関する REDUCE オプションをサポートします。
- バージョン 9.5 では、判別される最高水準点 (HWM) の値の正確性が向上しています。そのため、判別後に REDUCE、RESIZE、および DROP オプションを使用して、使用されているスペースの量をさらに正確に表すように表スペースのサイズを変更できます。続いて、解放されたスペースを他の表スペースが使用できます。

これらのステートメントによって減らせる表スペースのサイズは HWM より上の値に限られ、HWM より下の値にはできません。

#### 関連資料

SQL リファレンス 第 2 巻の『ALTER TABLESPACE ステートメント』

---

## ロック・タイムアウト診断が拡張された

バージョン 9.5 には、複雑な DB2 環境におけるロック・タイムアウト状況の診断を単純化して、それを訂正するのに必要な時間を削減するロック・タイムアウト・レポート機能が含まれています。

ロック・タイムアウト・レポート機能を使用して、ロック・タイムアウトの発生時にそれに関する情報をログに記録できます。情報は、ロック要求側 (ロック・タイムアウト・エラーを受け取ったアプリケーション) および現行ロック所有者の両方に対してキャプチャーされます。ログに記録された情報は、ロック・タイムアウトの原因になったロック競合に関係した主なアプリケーション、ロック・タイムアウト時にこれらのアプリケーションが実行していたことに関する詳細、および競合を起こしたロックに関する詳細を示します。検出されたロック・タイムアウトごとに、テキスト・レポートが書き込まれ、ファイルに保管されます。

**DB2\_CAPTURE\_LOCKTIMEOUT** レジストリー変数を設定して、ロック・タイムアウト・レポート機能を動的にオン/オフにできます。

#### 関連概念

データベース・パフォーマンスのチューニングの『ロック・タイムアウト・レポート』

#### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『汎用レジストリー変数』

データベース・パフォーマンスのチューニングの『ロック・タイムアウト・レポート・ファイル』

---

## 統計プロファイルのリセットするための **RUNSTATS** ユーティリティー・オプションが追加された

**RUNSTATS** ユーティリティーの新しい **UNSET PROFILE** オプションによって、統計プロファイルのリセットできます。

**RUNSTATS** ユーティリティーは統計プロファイルを登録および使用するオプションを提供します。統計プロファイルは、特定の表でどの統計が収集されるかを指定するオプションの集合です。

統計プロファイルのデフォルト設定に戻すには、**UNSET PROFILE** オプションを指定して **RUNSTATS** コマンドを発行します。別の方法として、**db2Runstats API** を呼び出して、**iRunstatsFlags** パラメーターを **DB2RUNSTATS\_UNSET\_PROFILE** に設定することもできます。

#### 関連概念

データベース・パフォーマンスのチューニングの『統計プロファイルを使用した統計の収集』

#### 関連資料

コマンド・リファレンスの『**RUNSTATS** コマンド』

管理 API リファレンスの『**db2Runstats API** - 表および索引の統計情報の更新』



---

## 第 5 章 ワークロード管理の機能拡張

バージョン 9.5 のフィーチャーは、以前のリリースで提供されたワークロード管理機能を拡張します。

このセクションでは、バージョン 9.5 のワークロード管理機能拡張について詳しく説明します。

---

### ワークロード管理の改善による制御の向上

バージョン 9.5 に包括的なワークロード管理が統合され、システムの実行状況をより深く洞察し、リソースおよびパフォーマンスに対するよりきめの細かい制御を行うことができます。

バージョン 9.5 のワークロード管理を使用すると、作業をクラスに分割し、データ・サーバーが同じシステム上のさまざまなユーザーおよびアプリケーションをサポートするように調整できます。トランザクション処理およびデータ・マイニングのために別個のデータベースを作成する必要はありません。

DB2 ワークロード管理フィーチャーを使用すると、以下のタスクを実行できます。

- ワークロード定義を使用して作業を自動的に識別し、ワークロードをサービス・クラスに割り当て、さらにリソースを各サービス・クラスに割り振ることにより、管理可能な論理グループに作業を分割できます。詳細なワークロード・プロファイルおよびパフォーマンス情報をキャプチャーして、ワークロードおよびサービス・クラス定義を洗練するために活用できます。
- コスト、時間、および並行性のしきい値を使用して実行を制御できます。これにより、他に影響を与えるような好ましくない照会を制御できるようになり、サービス・レベルの合意 (SLA) の目標を満たす助けとなります。しきい値を使用すると、システムは自動的に悪い状態に反応するか、またはそれが発生する前に予測することができます。トランザクションをスムーズに実行し続けると同時に、長時間実行されている照会および複雑な照会の影響を制御することができます。
- ユーザーに最新の状況情報を提供できるように、作業の処理のすべての段階をたどることができます。

DB2 のサービス・クラスを AIX Workload Manager (WLM) のサービス・クラスに結合することにより、AIX オペレーティング・システム的能力を活用できます。例えば、AIX WLM は、動的に CPU 共有を調整したり、他のサービス・クラスの予備の CPU 共有を使用したりすることで、データ・サーバーを最大限活用できます。

### DB2 ワークロード管理の概説

以下の図は、データ・サーバーに送信された複数の要求が特定のワークロードに対して評価され、割り当てられた後、該当するサービス・クラスで実行される方法を示しています。定義したワークロードに一致させることができない要求はデフォルトのワークロードに割り当てられ、これはデフォルトのサービス・クラスで実行さ

れます。

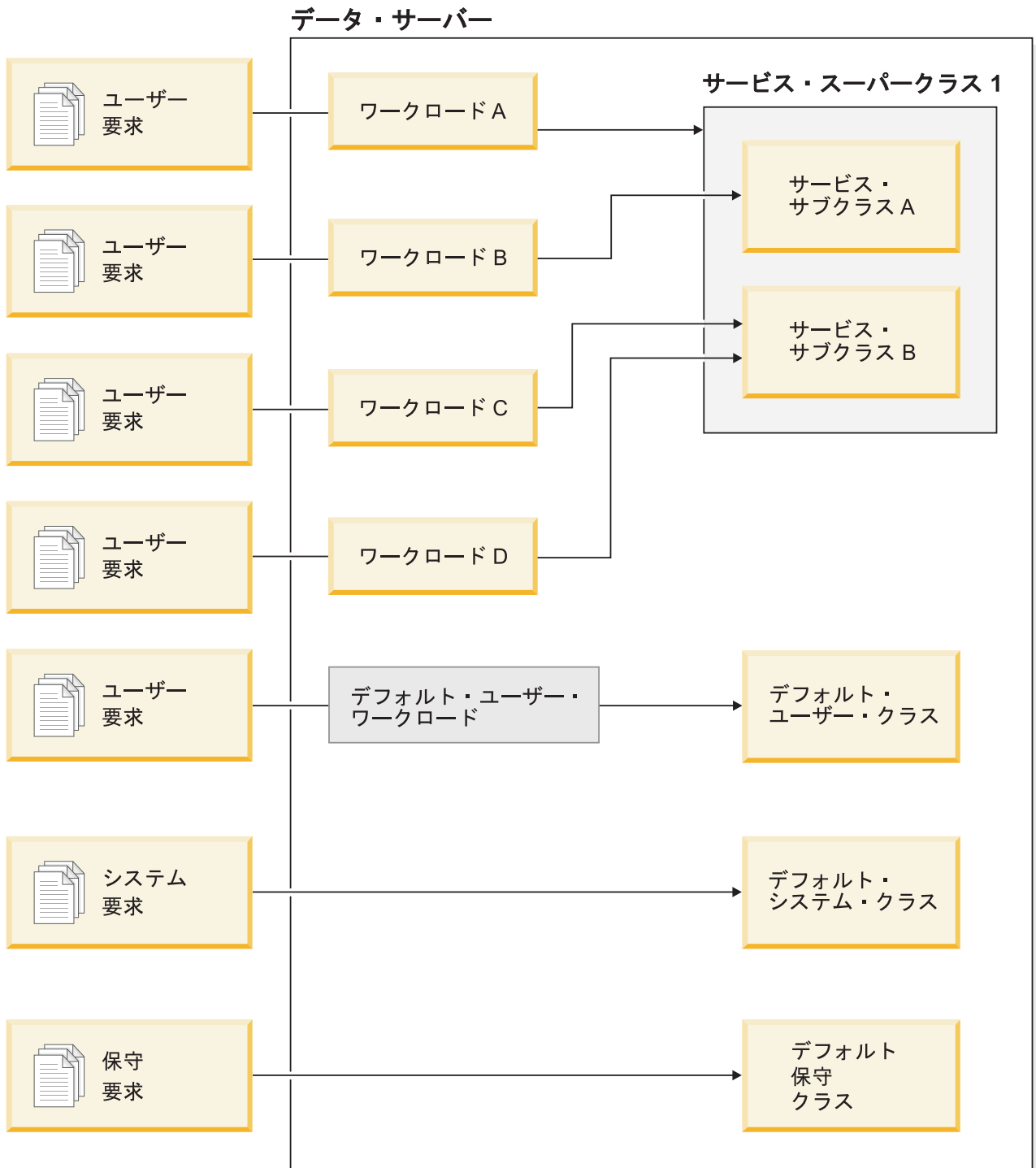


図1. サービス・クラスおよびワークロード

## DB2 ワークロード管理を使用する利点

システム・リソースに対する、よりきめの細かい制御

今日ではデータベース・アクティビティーの量は増大する一方であるため、CPU、I/O、メモリーなどのシステム・リソースの競合は、業務目標を達成



する上でますます障害となっています。機能拡張された DB2 ワークロード管理フィーチャーは、データベース・アクティビティーの定義済みセットを識別して、目標を満たすために必要な適切なリソースを割り当てることができる独自の実行環境にそれらを分離するために役立ちます。環境またはサービス・クラスの中で、優先度の高い作業でより多くのクリティカル・リソースを消費できるようにし、優先度の低い作業との競合を制御または除去できるように、システム・リソースを明示的に管理できます。

バージョン 9.5 では、サポートされるすべてのプラットフォームでリソース制御を使用できますが、AIX オペレーティング・システム上のデータ・サーバーは、WLM により提供されるリソース管理機能を直接活用できます。DB2 のサービス・クラスを AIX WLM のリソース・グループにリンクできます。これにより、データ・サーバーのワークロード管理機能とオペレーティング・システム自体のワークロード管理機能を結合させた、さらに動的なレベルのリソース管理を実現できます。

### システムの予測可能性と安定度の増大

ほとんどの業務には、データ・サーバーのワークロードに影響を与えるような、アクティビティーと要求の量が増加するピーク期間があります。ピーク期間は、ほとんどのユーザーがシステムを使用する日中の場合もありますし、すべての詳細な月次報告書の期限である月末である場合もあります。これらのピーク時の間、通常は予測可能な時間で実行されるワークロードの応答時間が、予測不能になる場合があります。ユーザーが使用できるリソースの量を制限するための制御を設定していないために、データ・サーバーで想定外の作業を行うユーザーにより意図しないピーク期間が発生する場合があります。例えば、ユーザーは、サーバーによる多くの処理を必要とする SQL ステートメント、またはデカルト結合などの複雑な結合を含む SELECT ステートメントを無意識に実行してしまう場合があります。

DB2 ワークロード管理を使用すると、適切なリソース割り振り、アクティビティーの優先順位付け、および効率的な方法で作業を処理するためのキューイング・オプションを事前に決定できるようにすることにより、ワークロードのピークを平滑化できます。これらの指示を定義した後、データ・サーバーはそれらを使用してリソースを割り振り、作業を優先順位付けします。例えば、過度に多くのデータベース・リソースを使用し、システムで実行中の他の照会に悪い影響を与え、さらにデータベース・システム全体に影響を与える可能性があるような、他に影響を与えるような好ましくない照会から作業を分離することができます。しきい値を使用すると、実行時間、SYSTEM TEMPORARY 表スペース使用量などのいくつかの異なる特性を使用してシステム内の許容できる照会動作を定義でき、予想どおりに実行されていない照会に対して取るアクションを定義できます。これらのアクションには、照会に関する詳細な情報を収集する機能および自動的にその照会を取り消す機能が含まれます。

### ワークロードが混合した環境での調整されたパフォーマンス要件

同じデータ・サーバーに混在するワークロードは同じリソースを共有しますが、パフォーマンス要件は異なる場合があります。例えば、バッチ・ワークロードは、データ・サーバーの処理が比較的少ない夜間によく実行されるので、日中に実行される日時報告書のジョブに悪影響を与えません。

DB2 ワークロード管理は、ワークロードに効果的に優先順位を付けられるようにし、リソースを最も必要とされる場所に振り向けることにより、混在したワークロードのパフォーマンスに着目する上で役立ちます。カスタマイズされた制御とリソース割り振りフィーチャーを使用することにより、システムにおけるデータ・サーバー・アクティビティの全体的なスループットを最大化できます。

さらに、具体的な尺度および漠然とした尺度を使用して、データ・サーバーのパフォーマンスを測定することもできます。具体的な尺度の例には、特定のセットのアクティビティを完了するために必要な時間の長さを示すデータベース統計、および単純な照会またはより複雑なジョブ（データをウェアハウスにロードするバッチ・ジョブなど）を完了するために必要な個々の時間の長さがあります。漠然とした方法には、データ・サーバーの応答時間についてのユーザーの印象および満足度があるかもしれません。

パフォーマンスを最適化するため、ワークロード管理モニター・フィーチャーを使用して、データ・サーバーで実行中の作業に関する集約情報および特定時点での情報の両方を取得することができます。必要な時間フレーム内で完了しない作業のタイプがある場合、モニター・データを使用して、状況を理解し、構成を変更する上で役立てることができます。例えば、追加のリソースをサービス・クラスに割り当てるか、あるいは特定のタイプの作業のためのリソース使用制御を割り当てるようにするかもしれません。これらの変更を行った後、システムの動作をモニターして、応答時間を満たすために実施した変更が有効であったこと、他の予期しない動作が発生していないことを確認できるかもしれません。ワークロード管理は反復プロセスです。ビジネス要件を満たす結果が得られるまで構成を洗練できます。

### 容易になった SLA 目標の管理とモニター

SLA とは、2 つのグループ間で期待することを定義するグループ間の正式な契約で、これには、サービス、優先順位、責任などの項目の目標が含まれます。SLA 目標は、応答時間の目標を使用して定式化されることがよくあります。例えば、特定の人事報告書を平均 5 分以内で実行する必要があるかもしれません。他の例として、現金レジスターから在庫システムへの更新が常に 2 秒未満で完了する必要があるという要件や、午前 9 時までには日時販売報告書を使用できるように、午前 8 時より前に実行されるバッチ・ジョブを使用してデータをロードする必要があるという要件があります。

以前は、SLA 目標に対する実際のパフォーマンスを追跡するには、統計を記録するためにカスタム・コーディングと、データを抽出してパフォーマンスの尺度を計算するための人手による手順を必要としました。しかし、DB2 ワークロード管理では、SLA で定義された目標に対する結果を測定するタスクを簡素化するための、一連の堅固なモニター機能が提供されています。個々のすべてのアクティビティに関するデータを収集しなくても平均応答時間をキャプチャーするための集約された手段を使用できます。例えば、現金レジスターでの販売が表に記録される度に応答時間値を収集する必要はありません。このような項目は 1 日に何十万も発生する場合があります。代わりに、目標が満たされているかを調べるために必要になったときにこれらの統計を収集できるようになりました。

DB2 ワークロード管理では、ヒストグラムを使用して測定しているアクティビティの分散に関する情報を保持します。応答時間に不整合がある（ま

たは非常に整合している) ことを示している可能性がある、特に高い (または低い) 標準偏差を調べ、応答時間が SLA 目標を満たしているかを調べるためにヒストグラムのデータを分析できます。

### 同じシステム上の複数の顧客グループに対する SLA 目標のカスタマイズ

データ・サーバーのリソースを共有する複数の顧客の SLA 目標は、必ずしも同じであるとは限りません。例えば、1 つのアプリケーションは 3 つの異なる部門により使用されているかもしれません。その 1 つの部門は平均 2 秒未満の応答時間を希望しており、他の 2 つの部門は 5 秒の応答時間で満足するかもしれません。

DB2 ワークロード管理を使用すると、同じデータベースを共有する顧客に対するカスタマイズされた SLA を独立してサポートできるようにするための固有の環境を提供できます。これらの実行環境を使用することにより、異なるタイプの顧客ごとにサービス・クラスを使用するデータ・サーバー上のアクティビティを分離することができます。例えば、グループごとにワークロードをセットアップした後、それをよりリソースが少ない別のサービス・クラスに割り当てることができます。それらのサービス・クラスをセットアップすると、顧客ごとに SLA 目標が満たされていることを確認するために、容易に集約アクティビティ統計を収集およびモニターできます。受け取ったサービスのレベルに基づいて各顧客に課金できます。

### 同じデータ・サーバー上でのアプリケーションとビジネス単位の統合の簡素化

ハードウェアおよびオペレーティング・システムがより多くのトランザクションを処理できるようになったため、より少ない台数のデータ・サーバー上で業務単位とアプリケーションを統合することにより、運用コストを削減するための多くの可能性が生まれます。DB2 ワークロード管理を使用すると、アプリケーションと業務単位が同じサーバーを使用するものの、同じ要件を持たず、財源が別であるような環境を管理できます。

例えば、給与計算部門用のデータ・サーバーが人事部門用のデータ・サーバーと組み合わせられているとします。給与計算グループには、毎週の給与を支払い、従業員の出費を追跡し、年末に税金の領収書を送信する作業の標準セットがあります。人事部門は通常のトレンド分析を行います。この部門は従業員データへの臨時的アクセスを必要とする問題を処理する必要があるため、より多くの臨時アクティビティがあります。各グループには独自の目標と優先順位のセットがあり、各部門の財源は異なります。同じデータ・サーバー上の各グループが、必ず許可されたリソースのみを使用するように、グループごとに別個の実行環境を作成できます。各環境に、その特定のグループに固有の作業が含まれるように定義でき、そのグループに対して、財源のレベルに対応したリソースの割り振りや優先順位の設定を行うことができます。

### 他のデータ・サーバーから移植されたデータベース・アクティビティの独立性

統合は、アプリケーションとデータが別のデータ・サーバー・システムから DB2 データ・サーバーに移植された場合にもよく発生します。DB2 ワークロード管理を使用して実行環境を設定し、移植されたアプリケーションが必要リソースを持てるようにする上で役立てることができます。また、この作業すべてをサービス・クラスの中で分離するとこれらのアクティビティ

一のモニターは単純になるため、古いデータ・サーバーと新しい DB2 データ・サーバーの間のパフォーマンスの比較が容易になり、新しいサーバーに移る決定を検証する上で役立ちます。

### データベース・アクティビティの動的モニター

すべてのデータベース・アクティビティは、DB2 サービス・クラスで実行される DB2 ワークロードに割り当てられます。バージョン 9.5 では、いくつかの表関数を使用して、ワークロードまたはサービス・クラスで見つかったアクティビティの状態および内容に関する情報を照会できます。この情報により、システム上で現在実行中の作業、パーティション間における分散状況、およびデータ・サーバー上でのリソース競合を引き起こす可能性がある特定のアクティビティがあるかについて、はっきり理解することができます。

### 拡張されたチャージバック機能

DB2 ワークロード管理のサービス・クラスから AIX WLM のサービス・クラスにマッピングすることにより、オペレーティング・システムのレベルで CPU リソースを制御できるだけでなく、サービス・クラスのレベルで CPU 使用量をモニターすることもできます。これにより、消費した CPU リソースの量に基づいて適切に業務単位に課金するための、より多くのオプションを活用できます。

サービス・クラスを使用すると、Query Patroller およびガバナーから新規ワークロード管理機能へのリソース管理の移動は簡単です。DB2 ワークロード管理を使用して管理を開始する作業のタイプを識別すると、各タイプの作業が実行されるサービス・クラスを定義し、各タイプの作業をそれに指定されたサービス・クラスにマップすることができます。この作業を管理するために Query Patroller を必要としなくなりました。DB2 ワークロード管理には、すべてのデータベース・パーティションにわたりアクティビティのライフ・サイクルを使用してデータベース・アクティビティをモニターおよび制御できるという、付加的な利点があります。

#### 関連概念

ワークロード・マネージャー ガイドおよびリファレンスの『ワークロード』

ワークロード・マネージャー ガイドおよびリファレンスの『サービス・クラス』

ワークロード・マネージャー ガイドおよびリファレンスの『DB2 ワークロード管理と AIX Workload Manager の統合』

ワークロード・マネージャー ガイドおよびリファレンスの『しきい値』

ワークロード・マネージャー ガイドおよびリファレンスの『ワークロード管理の概念の紹介』

ワークロード・マネージャー ガイドおよびリファレンスの『作業アクション・セット、作業アクション、作業クラス・セット、および作業クラス』

---

## 第 6 章 セキュリティーの向上

バージョン 9.5 のセキュリティーの向上には、トラステッド・コンテキストおよびロールのサポートと、監査およびラベル・ベースのアクセス制御の改善が含まれます。

このセクションでは、バージョン 9.5 のセキュリティーの向上について詳しく説明します。

---

### トラステッド・コンテキストによるセキュリティーの向上

トラステッド・コンテキストを使用すると、制限された機密情報を扱う特権を使用するときにより多くの制御を行えるようになり、中間層サーバーまたはアプリケーションがエンド・ユーザーの ID をデータベース・サーバーに示すことができます。

データベース接続は、接続属性が DB2 サーバーで定義されたトラステッド・コンテキストの属性と一致する場合、トラステッド接続と見なされます。信頼関係は、以下の属性のセットに基づいています。

- システム許可 ID: データベース接続を確立するユーザーを表します
- IP アドレス (またはドメイン・ネーム): データベース接続を確立したホストを表します
- データ・ストリーム暗号化: データベース・サーバーとデータベース・クライアントの間のデータ通信のための暗号化設定がある場合にはそれを表します

#### トラステッド・コンテキストを使用するとどのようにセキュリティーが向上するか

3 層アプリケーション・モデルは、クライアント・アプリケーションとデータベース・サーバーの間に中間層を置くことにより、標準的な 2 層クライアント/サーバー・モデルを拡張します。このモデルは、Web ベースのテクノロジーや Java 2 Enterprise Edition (J2EE) プラットフォームの登場により、近年特に大きな人気を得ています。3 層アプリケーション・モデルをサポートするソフトウェア・プロダクトの一例として、IBM WebSphere® Application Server (WAS) があります。

3 層アプリケーション・モデルでは、クライアント・アプリケーションを実行するユーザーの認証、およびデータベース・サーバーとの相互作用の管理は、中間層が処理します。従来の方法では、データベース・サーバーとのすべての相互作用は、データベース・サーバーに対して中間層を識別するユーザー ID と資格情報の組み合わせを使用して、その中間層により確立されたデータベース接続を介して行われていました。つまり、データベース・サーバーは、中間層のユーザー ID に関連付けられたデータベース特権を使用して、すべてのデータベース・アクセスで行う必要がある許可検査および監査を行っていました。これには、ユーザーの代わりに中間層により実行されるアクセスも含まれます。

3 層アプリケーション・モデルには多くの利点がありますが、データベース・サーバーとのすべての相互作用 (例えば、ユーザー要求) を中間層の許可 ID を使用して行うようにすると、以下のようないくつかのセキュリティー上の問題が生じます。

- **ユーザー ID の消失**

企業によっては、アクセス制御の目的で、データベースにアクセスしている実際のユーザーの ID を知りたい場合があります。

- **ユーザーの説明責任の減少**

監査による説明責任は、データベース・セキュリティーにおける基本原則です。ユーザーの ID が不明であると、中間層の固有の目的のために中間層により実行されるトランザクションと、ユーザーのために中間層により実行されるトランザクションを区別することが難しくなります。

- **中間層の許可 ID に特権を付与し過ぎる**

中間層の許可 ID には、すべてのユーザーからのすべての要求を実行するために必要なすべての特権が含まれていなければなりません。これにより、特定の情報への不必要なアクセス権限をユーザーに与えてしまうというセキュリティー問題が生じます。

- **弱いセキュリティー**

前述の特権の問題に加えて、現在の産業界の慣行では、接続するために中間層により使用される許可 ID には、ユーザー要求によりアクセスされる可能性があるすべてのリソースへの特権を付与する必要があります。中間層の許可 ID の暗号漏えいが発生すると、それらすべてのリソースは公開されてしまいます。

これらのセキュリティー上の問題は、実際のユーザーの ID およびデータベース特権が、そのユーザーに代わって中間層により実行されるデータベース要求で使用されるようなメカニズムが必要であることを示しています。この目標を達成するための最も簡単な方法は、中間層がユーザーの ID とパスワードを使用して新規接続を確立した後、ユーザーの要求をその接続を介して送信するというものです。この方法は単純ですが、以下のようないくつかの欠点があります。

- 特定の中間層では不適當。多くの中間層サーバーには、接続を確立するために必要なユーザー認証資格情報がありません。
- パフォーマンス上のオーバーヘッド。新しい物理接続を作成し、データベース・サーバーでユーザーを再認証することに関連した、パフォーマンス上の明らかなオーバーヘッドがあります。
- 保守上のオーバーヘッド。2 つのユーザー定義 (1 つは中間層上、もう 1 つはサーバー上) を持つことによる保守上のオーバーヘッドがあります。この状況では、異なる場所にあるパスワードを変更することが必要です。

トラステッド・コンテキスト機能は、この問題を解決します。セキュリティー管理者 (SECADM 権限の保有者) は、データベースと中間層の間の信頼関係を定義するトラステッド・コンテキスト・オブジェクトをデータベースに作成できます。その後、中間層ではデータベースへの明示的トラステッド接続を確立できますが、この接続では、接続の現行ユーザー ID を、認証のあるなしに関係なく別のユーザー ID に切り替える機能が中間層に付与されます。トラステッド・コンテキストは、ユーザーの ID アサーション問題を解決するだけでなく、別の利点もあります。それは、データベース・ユーザーが特権を使用できるようになる時期を制御する機能で

す。ユーザーが特権を使用できる時期を制御できないと、全体的なセキュリティーの低下につながります。例えば、特権が当初の意図とは異なる目的で使用される可能性があります。セキュリティー管理者は、1 つ以上の特権を 1 つのデータベースのロールに割り当て、そのロールをトラステッド・コンテキスト・オブジェクトに割り当てることができるようになりました。そのトラステッド・コンテキストの定義と一致するトラステッド・データベース接続 (明示的または暗黙的) のみがそのロールに関連付けられた特権を利用できます。

#### 関連概念

データベース・セキュリティー・ガイドの『トラステッド・コンテキストおよびトラステッド接続』

#### 関連タスク

データベース・セキュリティー・ガイドの『明示的トラステッド接続の確立とユーザー ID の切り替え』

#### 関連資料

SQL リファレンス 第 2 巻の『CREATE TRUSTED CONTEXT ステートメント』

---

## 監査機能のパフォーマンスと管理の機能拡張

監査ユーティリティーは、事前定義されてモニター対象となっている一連のデータベース・イベントの監査レコードの証跡を生成します。バージョン 9.5 では、監査機能が大幅に機能拡張されています。

バージョン 9.5 の DB2 監査機能の機能拡張には、きめの細かい構成、新しい監査区分、インスタンスとデータベースの別個のログ、および監査構成をカスタマイズするための新しい方法があります。監査対象となるデータベース・オブジェクトを厳密に制御できるようになったため、調べる必要がないデータベース・オブジェクトで発生するイベントを監査する必要がなくなりました。そのため、監査のパフォーマンス (およびそのパフォーマンスが他のデータベース操作に与える影響) は、かなり改善されました。

データベース・レベルで監査を管理する責務は、セキュリティー管理者だけが担当するようになりました。

バージョン 9.5 では、以下の監査機能の機能拡張が含まれています。

- 監査ポリシーと呼ばれる新規データベース・オブジェクトを使用してデータベース内の監査構成を制御できます。

データベース内の表、ユーザー、グループ、およびロールに至るまでの特定のオブジェクトが独自の監査構成を持てるように、個々のデータベースでも監査構成を持つことができます。必要な情報に容易にアクセスできるだけでなく、この機能拡張によりディスクに書き込むデータは少なくて済むため、パフォーマンスも改善されています。

- SQL ステートメントの監査が容易になり、生成される出力が少なくなりました。

新しい監査区分である EXECUTE を使用すると、実行中の SQL ステートメントのみを監査できます。以前は、この詳細をキャプチャーするために CONTEXT イベントを監査する必要がありました。

- 監査ログがデータベースごとに存在します。

V9.5 から、インスタンスに 1 つの監査ログ、データベースごとに 1 つの監査ログがあります。このフィーチャーにより、監査レビューが簡単になります。

- 監査ログにカスタマイズ可能なパスが含まれるようになりました。

監査ログのパスを制御できると、監査ログを大容量の高速なディスクに置くことができ、データベース・パーティション (DPF) のインストール済み環境では、オプションでノードごとにディスクを分けることも可能です。このフィーチャーによって、オフラインで監査ログをアーカイブすることができ、必要になるときまでデータを抽出する必要はありません。

- 監査ログをアーカイブできます。

監査ログをアーカイブすると、現在の監査ログはアーカイブ・ディレクトリーに移され、サーバーは新しいアクティブ監査ログの書き込みを開始します。監査ログからデータベースの表にデータを抽出するときは、アクティブ監査ログからではなく、アーカイブ・ログからとなります。これにより、アクティブ監査ログのロックが原因となるパフォーマンスの低下を防ぎます。

- セキュリティー管理者 (SECADM 権限の保有者) が各データベースの監査を管理するようになりました。

セキュリティー管理者だけがデータベースの監査の構成を制御します。システム管理者 (SYSADM 権限の保有者) はこの権限を持たなくなりました。セキュリティー管理者は、監査ログの操作、ARCHIVE コマンドの発行、および表へのログ・ファイルの抽出のための十分なアクセス権限も持ちます。

- 各区分の新しい情報を監査することができます。

CURRENT CLIENT 特殊レジスターを使用すると、クライアントのユーザー ID、アカウントिंग・ストリング、ワークステーション名、およびアプリケーション名の値をアプリケーション内部に設定して、これらの値が監査データに記録されるようにすることができます。

ローカルおよびグローバル・トランザクション ID を監査データに記録できます。これにより、監査ログとトランザクション・ログの間の相関が容易になります。

#### 関連概念

データベース・セキュリティー・ガイドの『監査ポリシー』

データベース・セキュリティー・ガイドの『SQL ステートメントを監査するための EXECUTE 区分』

データベース・セキュリティー・ガイドの『監査ログの保管と分析』

#### 関連資料

SQL リファレンス 第 2 巻の『CREATE AUDIT POLICY ステートメント』

SQL リファレンス 第 2 巻の『AUDIT ステートメント』



## ロールを使用した特権の管理の簡素化

ロールは、特権の管理を簡素化します。つまり、グループと同等の機能は提供されませんが、同じ制約事項は設けられません。

ロールは、1 つ以上の特権と一緒にグループ化するデータベース・オブジェクトです。ロールを、GRANT ステートメントを使用してユーザー、グループ、PUBLIC、またはその他のロールに割り当てることもできますし、CREATE TRUSTED CONTEXT または ALTER TRUSTED CONTEXT ステートメントを使用してトラステッド・コンテキストにロールを割り当てることもできます。ワークロード定義内で、SESSION\_USER ROLE 接続属性用のロールを指定することができます。

ロールには、以下のようないくつかの利点があります。

- 組織の構造を映し出すような方法で、データベースへのアクセスを制御することができます (組織における役職や担当業務に直接対応したロールをデータベース内に作成できます)。
- ユーザーには、その役職や担当業務に応じたロールに対するメンバーシップを付与できます。ユーザーの仕事の責務が変わったら、容易にユーザーに新しいロールへのアクセス権限を付与し、古いロールへのアクセス権限を取り消すことができます。
- 特権の割り当てが簡素化されます。特定の役職や担当業務に該当する個々のユーザーに一連の同じ特権を付与するのではなく、その役職や担当業務に応じたロールに対してこの一連の特権を付与してから、その役職や担当業務に該当する各ユーザーにそのロールを付与することができます。
- ロールの特権を更新した場合、そのロールが付与されたすべてのユーザーが更新を受け取ります。つまり、個人ごとに各ユーザーの特権を更新する必要はありません。
- ビュー、トリガー、マテリアライズ照会表 (MQT)、静的 SQL、および SQL ルーチンの作成時には、ロールに対して付与された特権および権限が常に使用されます。この場合、グループに付与された特権および権限は (直接でも間接にでも) 使用されません。

その理由は、グループはサード・パーティー・ソフトウェア (例えば、オペレーティング・システム) によって管理されるので、DB2 データベース・システムは、グループ内のメンバーシップがいつ変更になったかを判別できないからです。ロールはデータベース内部で管理されるので、DB2 データベース・システムは、許可がいつ変更されたかを判別して、それに応じたアクションをとることができます。グループに付与されたロールは、外部で管理されるため、使用されません。

- ユーザーに割り当てられたすべてのロールは、ユーザーが接続を確立したときに有効になるので、ロールに付与されたすべての特権と権限も、ユーザーが接続するときに有効となります。明示的にロールを有効または無効にすることはできません。
- セキュリティー管理者は、ロールの管理を他人に委任することができます。

### 関連概念

データベース・セキュリティー・ガイドの『ロール』

### 関連資料

## ラベル・ベースのアクセス制御 (LBAC) の機能拡張によりセキュリティが向上している

ラベル・ベースのアクセス制御 (LBAC) の機能拡張が導入されており、グループだけでなくロールに対してセキュリティ・ラベルと免除を付与できます。また、セキュリティ・ラベル・コンポーネントに新しいエレメントを追加したり、セキュリティ・ポリシーに変更を加えて動作を変更したりセキュリティ・ラベルにコンポーネントを追加したりできるようになりました。

LBAC の機能拡張は以下のとおりです。

- 新しい ALTER SECURITY LABEL COMPONENT ステートメントを使用して、新しいエレメントをセキュリティ・ラベル・コンポーネントに追加できます。
- 新しい ALTER SECURITY POLICY ステートメントを使用して、セキュリティ・ポリシーを変更できます。定義済みのセキュリティ・ポリシーにコンポーネントを追加できます。ポリシーの更新中は表がポリシーを使用中であってはなりません。また、このステートメントを使用して、セキュリティ・ラベルと免除の付与を使用可能または使用不可にしたり、セキュリティ・ポリシーに関する書き込み許可のエラー動作を変更したりできます。
- GRANT SECURITY LABEL ステートメントを使用して、ロールおよびグループにセキュリティ・ラベルを付与できるようになりました。REVOKE SECURITY LABEL ステートメントを使用してロールおよびグループからセキュリティ・ラベルを取り消すことができるようになりました。
- GRANT EXEMPTION ステートメントを使用して、ロールおよびグループに免除を付与できるようになりました。REVOKE EXEMPTION ステートメントを使用してロールおよびグループから免除を取り消すことができるようになりました。

### 関連概念

データベース・セキュリティ・ガイドの『ラベル・ベースのアクセス制御 (LBAC) の概要』

データベース・セキュリティ・ガイドの『LBAC セキュリティ・ポリシー』

データベース・セキュリティ・ガイドの『LBAC セキュリティ・ラベル・コンポーネントの概要』

データベース・セキュリティ・ガイドの『LBAC セキュリティ・ラベル』

データベース・セキュリティ・ガイドの『LBAC 規則の免除』

---

## 第 7 章 パフォーマンスの向上

DB2 バージョン 9.5 ではさまざまなパフォーマンスの向上が図られ、時系列、空間データ、およびスライディング照会ウィンドウが関係する複雑な照会の照会時間において劇的な改善が見られます。

このセクションでは、バージョン 9.5 で導入されているパフォーマンスの向上について詳しく説明します。

---

### LOB 列を参照する照会の高速化

バージョン 9.5 では、LOB 列を含む多数の行のデータが含まれる結果セットを返す照会のパフォーマンスが改善されました。

これらのパフォーマンスの改善は、結果セットに LOB データが含まれる場合に、複数行のデータをブロック化してまとめ、クライアントに対して単一のカーソル要求で結果セットを返すことができるように変更されたことによるものです。LOB データ・タイプへの参照を含む行データのブロックングは、すべての環境でサポートされています。以前は、行データのブロックングは、LOB 列を含まない結果セットに制限されていました。

さらに、バージョン 9.5 サーバーは、サーバーが LOB 値を最適の方法で返すことができる、Dynamic Data Format をサポートします。これにより、LOB 列を含むカーソルから行を取り出すためにかかる時間の長さが削減されるはずです。ほとんどの IBM データ・サーバー・クライアントは、バージョン 9.1 フィックスパック 1 以降、Dynamic Data Format フィーチャーをサポートしており、連続ストリーミングを使用して LOB 値を取り出すために自動的にそれを活用しています。

LOB 値の改良された行ブロックングは、以下のように制限されています。

- LOB 列を参照する組み込み SQL アプリケーションの照会では、改善はほとんどあるいはまったくありません。
- 照会の中で LOB 値を出力するユーザー定義関数がある場合、カーソル・ブロックングが無効になります。

このパフォーマンス・フィーチャーは、DB2 for z/OS および DB2 for i5/OS でも使用できます。

#### 関連概念

Java アプリケーションの開発の『IBM Data Server Driver for JDBC and SQLJ を使用したプログレッシブ・ストリーミング』

## オプティミスティック並行性制御と更新検出の機能拡張により拡張が容易なロックの代替策が提供されている

オプティミスティック並行性制御に対する機能拡張により、並行データ・アクセスのデータベース・ロックに対する、拡張がさらに容易になった代替ロックが提供されています。このロックは、値ベースのオプティミスティック・ロックと置き換えられており、パフォーマンスも向上しています。関連した機能拡張により、最新の(日、週、または月単位) データベース更新を検出するメカニズムが提供されています。

### オプティミスティック・ロックおよびオプティミスティック並行性制御

オプティミスティック・ロックにより、他のトランザクションが特定のリソースを使用できない時間を最小化できます。行に変更が加えられる時点をデータベース・マネージャーが判別できるので、データ保全性を確保しながら、ロックを保持する時間を制限できます。データベース・マネージャーは、オプティミスティック並行性制御を使用して、読み取り操作の後で即時に行またはページ・ロックを解除します。

オプティミスティック並行性制御は、初期選択リストで `RID_BIT()` および `ROW CHANGE TOKEN` 式を使用して、検索される更新の述部として元の値を提供できるようにすることによりサポートされます。

### 日、週、および月単位の更新検出

データベース管理者は、特定の時間範囲内の更新量が分かれば、データのレプリケーションを計画したり、監査シナリオを作成したりする場合などに利点があります。

新しい式 `ROW CHANGE` は、`SELECT`、`INSERT`、および `UPDATE` ステートメントと併用し、行に対する最終変更時点を表すトークンまたはタイム・スタンプを戻します。現バージョンでは、アプリケーションには以下のオプションがあります。

- `ROW CHANGE TIMESTAMP` 式を使用して、行の最終変更時点 (または日付の範囲内の変更か日数の変更) を判別します。
- `ROW CHANGE TOKEN` 式を使用して、行の変更シーケンス内の相対点を表す `BIGINT` (多倍長整数) 値としてトークンを戻します。

#### 関連概念

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『オプティミスティック・ロック』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『時間に基づく更新の検出』

#### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『オプティミスティック・ロックの概要』

## 索引の据え置きクリーンアップ・オプションを使用したより高速な MDC ロールアウト削除

レコード ID (RID) 索引のクリーンアップを、マルチディメンション・クラスタリング (MDC) 表でのロールアウト削除が完了した後に実行されるように指定できるようになりました。この索引の据え置きクリーンアップ・オプションにより、特定のロールアウト削除操作の速度がかなり改善されます。

バージョン 8.2.2 以降、ロールアウト削除を活用できるようになりました。これには、ブロック・ベースの削除の高速化とロギングの削減という二重の利点があります。ロールアウトの効率は、MDC 表のどのブロックに削除対象となる行が含まれているかを知ることと、特定のブロック内のすべての行が削除されるという点に基づいています。しかし、バージョン 9.5 より前でのロールアウトのパフォーマンス上の利点は、ご使用の表に含まれる RID 索引の量に依存していました。なぜなら、バージョン 9.5 より前のロールアウトでは、索引ごとに行レベルでの処理とロギングが必要であったためです。

バージョン 9.5 では、RID 索引の据え置きクリーンアップのオプションを備えることにより、ロールアウト削除のパフォーマンスをさらに向上させています。これにより、削除処理中に表の削除中の行を参照する索引キーは除去しなくなったため、ディメンションの境界で発生する削除操作がはるかに高速になります。トランザクションがコミットされ、その後 RID 索引が非同期でクリーンアップされるまで、DB2 はこれらの索引を更新せずにブロックにロールアウトとしてマークを付けます。

ロールアウト削除の据え置きクリーンアップの速度がどの程度改善されたかは、実行する削除の種類によって異なります。表に多数の RID 索引がある場合、または削除のサイズが非常に大きい場合、据え置きクリーンアップの削除時間はかなり削減されます。同様に、特定の表に対していくつかのロールアウト削除を計画している場合、ロールアウトされたすべてのブロックに対して RID 索引のクリーンアップを 1 回だけ実行した方がはるかに高速になります。

新しいフィーチャーを有効にするには 2 つの方法があります。一つは、既存の **DB2\_MDC\_ROLLOUT** レジストリー変数（動的に構成可能）を新規値である **DEFER** に設定することで可能です。あるいは、**SET CURRENT MDC ROLLOUT MODE** ステートメントを使用して、**CURRENT MDC ROLLOUT** 特殊レジスターを **DEFERRED** に設定することで可能です。新規データベース・モニター要素、**BLOCKS\_PENDING\_CLEANUP** を使用すると、クリーンアップを保留している MDC 表のブロック数を判別できます。

デフォルトでの削除の動作は、即時索引クリーンアップでのロールアウトです。追加されたフィーチャーでは、索引の据え置きクリーンアップが必要となる時期を決定できます。**DB2\_MDC\_ROLLOUT** は動的であるため、**DELETE** ステートメントを新しくコンパイルする時には新しい設定値が使用されます。ただし、**CURRENT MDC ROLLOUT MODE** 特殊レジスターを使用した方がロールアウトの動作をより適切に制御できるかもしれません。

### 関連概念

パーティションおよびクラスタリングのガイドの『MDC 表の最適化ストラテジー』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『MDC 表の非同期索引クリーンアップ』

#### 関連資料

SQL リファレンス 第 1 巻の『CURRENT MDC ROLLOUT MODE 特殊レジスター』

SQL リファレンス 第 2 巻の『SET CURRENT MDC ROLLOUT MODE ステートメント』

システム・モニター ガイドおよびリファレンスの『blocks\_pending\_cleanup クリーンアップ・ロールアウト・ブロックの保留：モニター・エレメント』

---

## 索引構築の並列処理がデフォルトで使用可能にされる

バージョン 9.5 では、索引構築の並列処理が、**intra\_parallel** 構成パラメーターの設定にかかわらず使用可能になります。

複数のプロセッサを使用して索引データのスキャンやソートを並列して行うと、CREATE INDEX ステートメントのパフォーマンスを改善できます。索引マネージャーのコンポーネントにより索引の作成を並列化するかどうかが決まります。この決定は、表サイズや複数のプロセッサがあるかどうかなどの多くの考慮事項に基づいて行われます。この決定は、**intra\_parallel** 構成パラメーターによって制御されなくなりました。

#### 関連概念

データ移動ユーティリティ ガイドおよびリファレンスの『並列処理とロード』

パーティションおよびクラスタリングのガイドの『パーティション内並列処理の最適化ストラテジー』

#### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『照会の最大並列処理の度合い構成パラメーター』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『パーティション内並列処理機能の使用可能化構成パラメーター』

---

## 拡張された OLAP 関数

新規ならびに改良された OLAP 関数により、照会に対して、特に大きなスライディング・ウィンドウのデータに対して実行される集約に対して、より良好な制御が可能になります。

標準装備の OLAP 関数は、3 つの方法で改良されました。

- 4 つの新しい関数を使用できます。新しい LEAD、LAG、FIRST\_VALUE、および LAST\_VALUE OLAP 関数によって、表の行が読まれると共に移動する行のウィンドウ (スライディング・ウィンドウ) で実行される集約に対して、より優れた制御が使用可能になります。これらの新しい関数によって、現在行以外の行にある特定の値にアクセスできます。例えば、OLAP 関数ウィンドウが特定の期間中の株式の取り引きを表す場合、これらの新しい OLAP 関数はその期間中の最初と最後の取り引き (おそらく始値と終値) を返すことができます。また、特定の日に

取り引きがなかった株式の終値の 150 日移動平均を計算する場合、金融機関は移動平均の計算に他の日（取り引きがなかった日の前日または翌日のどちらか）の終値を算入することを好む場合があります。これには、計算で使用される欠落 (NULL) 値を補正することにより、計算を平滑化する効果があります。

- 既存の OLAP 関数をスライディング・ウィンドウのデータに適用するときのパフォーマンスが改善されました。これらのパフォーマンスの向上は劇的で、結果として、いくつかの照会は、以前のバージョンと比較して幾百倍も高速に実行されることがあります。
- OLAP メモリーの使用が最適化されました。OLAP 関数によって使用されるメモリーはセルフチューニング・メモリーによって動的にチューニングできるので、OLAP 関数はメモリーによる制約が小さくなりました。これらの最適化により、ずっと大きなウィンドウのデータに対する照会が可能となります。現在では、システム・メモリーよりも、使用可能な一時的スペースのほうが、多くの照会にとって制約要因となっています。

#### 関連資料

SQL リファレンス 第 1 巻の『OLAP 仕様』

---

## 照会オプティマイザーが拡張されている

複合照会が（特に金融セクターで関心が向けられる照会タイプにおいて）最適化されています。

以下のタイプの照会で、かなりのパフォーマンスの向上を得ることができます。

- 単一の SELECT ステートメントに複数の異なる集約が含まれる照会
- ORDER BY および FETCH FIRST n ROWS ONLY セマンティクスが含まれる照会
- GROUP BY 節が指定された MIN および MAX 関数を持つ照会
- 大規模な IN リスト述部がある照会（リスト全体の検索が改善されており、そのような述部の評価のために、さらに多くのストラテジーが照会最適化時に考慮されます）

#### 関連概念

53 ページの『第 7 章 パフォーマンスの向上』

56 ページの『拡張された OLAP 関数』

---

## NO FILE SYSTEM CACHING デフォルトによるファイル・システムのキャッシュ・メモリーの削減

バージョン 9.5 で作成する表スペース・コンテナでは、デフォルトでデータベース・マネージャーは可能な場合には常に Concurrent I/O (CIO) の使用を試行します。CIO がサポートされないシステム構成では、Direct I/O (DIO) または buffered I/O が代わりに使用されます。

CIO および DIO では、メモリー・パフォーマンスが改善されます。これらの設定によりデータベース・マネージャーはファイル・システム・レベルでキャッシングを迂回できるためです。この処理は CPU オーバーヘッドを削減して、データベース・インスタンスがより多くのメモリーを使用できるようにします。CIO、DIO、

またはファイル・システム・キャッシングをサポートするシステム構成については、『ファイル・システム・キャッシング構成』を参照してください。

FILE SYSTEM CACHING および NO FILE SYSTEM CACHING 属性は、入出力操作をファイル・システム・レベルでキャッシュするかどうかを指定します。データベース・マネージャーは、自身のデータ・キャッシングをバッファー・プールを使って管理します。バッファー・プールのサイズが適切に調整されていれば、ファイル・システム・レベルのキャッシングは必要ありません。

新しいデフォルトは、バージョン 9.5 より前に作成した表スペースには適用されません。

セルフチューニング・メモリー・マネージャーを使用可能化すること、および ALTER BUFFERPOOL ステートメントの AUTOMATIC バッファー・プール・サイズ設定と組み合わせることにより、新しいデフォルトは以下の利点を提供します。

- 非バッファー入出力の利点を利用するために、NO FILE SYSTEM CACHING を CREATE TABLESPACE ステートメントに明示的に指定する必要はありません。
- データがバッファー・プール・レベルで自動的にキャッシュされるために、ファイル・システム・キャッシュの使用量が削減します。その結果、ファイル・システム・キャッシュ専用として割り振られるメモリーの量が削減されます。

#### 関連概念

データベース・パフォーマンスのチューニングの『複数のデータベース・バッファー・プールの管理』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『セルフチューニング・メモリー』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『ファイル・システム・キャッシング構成』

#### 関連タスク

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『表スペースの作成』

#### 関連資料

SQL リファレンス 第 2 巻の『ALTER BUFFERPOOL ステートメント』

SQL リファレンス 第 2 巻の『CREATE TABLESPACE ステートメント』

---

## DB2 Spatial Extender の照会パフォーマンスの改善

DB2 Spatial Extender は、地形に関する空間情報を生成および分析し、この情報の基となるデータを保管および管理します。DB2 Spatial Extender の照会パフォーマンスは、劇的に改善されました。

一般的には LOB データとして保管される顧客の位置、携帯電話の電波塔、道路、町、その他のオブジェクトに関する空間データをより効率的に照会できるようになりました。これは、LOB 列を参照する照会において、基となる DB2 データベースのクライアントとサーバー間のデータ・ブロッキングの改善によるものです。このデータをより高速に取り出すことにより、サード・パーティー・ソフトウェアによる空間データ・マップの描画と表示が高速になります。例えば、バージョン 9.5 より前の ESRI ArcGIS を使用して普通の待ち時間のネットワーク・リンクを介してマ



ップを描画すると、2 分以上かかっていた。バージョン 9.5 の ArcGIS を、新しい LOB ブロッキング・サポートを有効にして使用すると、同じマップが 4 秒で表示されます。

#### 関連情報

Spatial Extender and Geodetic Data Management Feature ユーザーズ・ガイドおよびリファレンスの『DB2 Spatial Extender の目的』

---

## EXPLAIN 可能ステートメントの追加

バージョン 9.5 では、REFRESH TABLE および SET INTEGRITY ステートメントが EXPLAIN 可能ステートメント (EXPLAIN 情報を収集し分析するために使用し、パフォーマンス上の問題を診断する) のリストに追加されました。この機能強化により、マテリアライズ照会表 (MQT) の保守が容易になります。

SET INTEGRITY および REFRESH TABLE ステートメントのパフォーマンス上の問題のセルフ診断を行うためにアクセス・プランを入手できるようになりました。

#### 関連資料

SQL リファレンス 第 2 巻の『EXPLAIN ステートメント』

SQL リファレンス 第 2 巻の『REFRESH TABLE ステートメント』

データ移動ユーティリティー ガイドおよびリファレンスの『SET INTEGRITY ステートメント』

SQL リファレンス 第 1 巻の『CURRENT EXPLAIN MODE 特殊レジスター』

SQL リファレンス 第 1 巻の『CURRENT EXPLAIN SNAPSHOT 特殊レジスター』

SQL リファレンス 第 1 巻の『EXPLAIN\_STATEMENT 表』

SQL リファレンス 第 1 巻の『Explain 表』



---

## 第 8 章 pureXML の機能拡張

バージョン 9.5 は、バージョン 9.1 で初めて導入された pureXML サポートを基に構築されており、リレーショナルおよび XML データ・サーバーの混成機能を拡張することにより、XML データ処理がさらに柔軟、高速、かつ信頼性のあるものになっています。

このセクションでは、バージョン 9.5 で導入されている pureXML の機能拡張について詳しく説明します。

### 関連概念

pureXML ガイドの『pureXML の概要』

pureXML ガイドの『pureXML のチュートリアル』

pureXML ガイドの『XML 入出力の概要』

### 関連タスク

国際化対応ガイドの『非 Unicode データベースを Unicode に変換する』

---

## XQuery 更新式を使用すると XML 文書の部分を変更できる

新しい XQuery 更新式を使用すると、新しい XML 文書を構成する代わりに、既存の XML 文書の部分に変更を加えることができます。

例えば、DB2 データベース中の XML 列に XML 文書として保管されている購入注文がある場合は、DB2 XQuery 更新式を使用して、新しい行項目エレメントを購入注文に追加できます。新しい行項目のある購入注文を再作成する必要はありません。更新式は、XML データを別の形式に変換せずに XML 文書に変更を加えることができます。

DB2 XQuery 更新式は、XQuery トランスフォーム式によって作成される XML データのコピーに対して常に作業します。トランスフォーム式は copy 節、modify 節、および return 節の 3 つの節で構成されます。copy 節は XML データのコピーを作成し、modify 節にはコピーに変更を加える XQuery 更新式が含まれ、return 節には変更済みの XML データのコピーをさらに処理できる他の非更新式を含めることができます。

更新式を含む DB2 XQuery 式は、W3C 文書に記述されている XQuery 言語に基づいています。この式は、XQuery および XPath データ・モデル (XDM) のインスタンスを操作します。XDM により、XQuery は、XML 文書またはフラグメントの表面的な構文ではなく、その抽象的な論理構造を操作することができるようになります。XQuery 式の入力 (あれば) は、XDM のインスタンスであり、式の結果も XDM のインスタンスです。XML 文書は、XML 列に保管されるときに XDM に変換されます。

### 関連情報

XQuery リファレンスの『変換式での更新式の使用』

---

## pureXML 用ロード・ユーティリティのサポートの追加

バージョン 9.1 において XML データで使用できるようになったインポートおよびエクスポート・ユーティリティに加えて、ハイパフォーマンス・ロード・ユーティリティを使用して XML 文書を DB2 表に挿入できるようになりました。

ロード・ユーティリティは、大容量のデータを短い時間フレーム内で表に挿入する必要がある場合に特に役立ちます。ロード・ユーティリティはフォーマットしたページをデータベースに直接書き込むため、データのロードはインポートよりはるかに高速になる場合があります。また、ロード操作中に最小限のロギングが実行されるため、データの移動がより効率的になります。XML データをロードする機能により、CURSOR ファイル・タイプからのロードなどのロード固有のオプションや、さまざまなデータ操作フィーチャーを使用することもできます。

### 関連概念

データ移動ユーティリティ ガイドおよびリファレンスの『XML データのロード』

### 関連資料

データ移動ユーティリティ ガイドおよびリファレンスの『インポート・ユーティリティとロード・ユーティリティの相違点』

---

## pureXML アプリケーション処理のパフォーマンスが改善された

pureXML フィーチャーのパフォーマンスが改善されています。XML データを処理するアプリケーションの実行時間が短くなり、場合によってはリソース使用量が減っています。

改善範囲は広く、SQL/XML と XQuery の両方のデータ操作、XML データ作成に関する索引、照会コンパイラーおよび最適化、XML 文書のナビゲーションなどの領域が含まれます。以下に、さらに特定化した改善例を示します。

- 挿入、更新、および削除操作の実行速度が上がりました。
- 照会評価時の索引の ANDing に、XML データの索引とリレーショナル索引の両方を組み込めます。
- UNION ALL ビューの照会で XML データの索引を使用できます。
- 照会評価中に複雑なアクセス・プランに必要な NLJOIN 演算子が少なくなりました。

pureXML のトピックに新しいパフォーマンスのガイドラインが組み込まれています。

### 関連概念

データベース・パフォーマンスのチューニングの『Explain 機能』

---

## チェック制約機能が拡張された

XML 列に対するチェック制約を使用して追加のオプションを指定し、処理の前に情報の整合性を確認できるようになりました。

チェック制約を使用すると、XML 列に対する特定の制約事項を設けることができます。この制約は、XML 列中でのデータの挿入や更新が試行されるたびに強制されます。この制約で指定された基準が true と評価される場合のみ操作が実行されます。

VALIDATED 述部を使用し、オプションで ACCORDING TO XMLSCHEMA 節を指定して 1 つ以上の登録済みの XML スキーマを組み込むと、XML 値を妥当性検査するかしないかを要求できるようになりました。

また、BEFORE トリガーの一部としてタイプ XML の相関名を使用した列制約を指定できるようになりました。

#### 関連概念

pureXML ガイドの『XML 列のチェック制約』

---

## トリガー処理で XML 文書の自動妥当性検査がサポートされている

トリガー処理が、XML 文書の現在の妥当性検査の状態に基づいた、登録済み XML スキーマに対する XML 文書の自動妥当性検査をサポートするようになりました。

XML 列に保管される前の登録済み XML スキーマに対する XML 文書の妥当性検査はオプションですが、データ安全性が不明な場合は行う必要があります。なぜなら、この妥当性検査により、有効な XML 文書のみ挿入や更新が行われることが保証されるからです。

登録済み XML スキーマに対する XML 文書の自動妥当性検査の際に、BEFORE トリガーはタイプ XML の NEW AS 相関名を指定して、SET ステートメントから XMLVALIDATE 関数を呼び出したり、値を NULL に設定したり、タイプ XML の値を未変更のままにしたりできます。

XML スキーマに対する XML 文書の妥当性検査を起動しなければならないかどうかを判別するには、BEFORE トリガーの WHEN 節に検索条件 IS VALIDATED または IS NOT VALIDATED を組み込み、オプションで ACCORDING TO XMLSCHEMA 節を指定して 1 つ以上の XML スキーマを組み込むことによって、XML 文書の妥当性検査の状態をテストできます。

#### 関連概念

pureXML ガイドの『XML データのトリガー処理』

---

## XSLT サポートによる XML データから他の形式へのトランスフォームの使用可能化

新規 XSLTRANSFORM 関数を使用すると、データベースに置かれている XML 文書を HTML、プレーン・テキスト、または他の XML 形式に変換することができます。

XSLT トランスフォーメーションは、XML データを他の形式にトランスフォームするために使用される標準的な方法で、単一のデータ・ソースから複数の出力形式を生成することを可能にします。この機能は、XML Extender で提供される XSLT トランスフォームに似ています。

XSLTRANSFORM は、スタイル・シートを使用して XML を他のデータ・フォーマットに変換します。XML 文書の一部またはすべてを変換し、XPath 照会言語および XSLT の組み込み関数を使用してデータを選択または再配置することができます。

XSLTRANSFORM 関数の重要なフィーチャーは、XSLT パラメーターを実行時に受け入れる機能です。この機能がない場合は、XSLT スタイル・シートの大規模なライブラリー (XML データに対する照会の各変種ごとに 1 つ) を保守するか、新しい照会の種類ごとにスタイル・シートを手動で編集する必要があります。パラメーターの引き渡しにより、スタイル・シートの標準化された動作と、特定の照会のカスタマイズされた要件または固有の要件を分離できます。中核となるスタイル・シートを注意深く設計した後、それらはそのままにしておいて、カスタム動作をパラメーター・ファイルにオフロードすることができます。

#### 関連概念

pureXML ガイドの『XSLT スタイルシートを使用した変換』

---

## SQL/XML および XQuery パラメーターを渡す際の柔軟性が増している

SQL/XML と XQuery の両方で、パラメーターの引き渡しが単純化されて拡張され、柔軟性が増しています。

SQL ステートメントと XQuery 式を結合して発行する際に、SQL ステートメントと XQuery 式の間で以下のようにデータを渡すことができるようになりました。

#### SQL からのパラメーターの引き渡し

- デフォルトでは、XMLQuery スカラー関数、XMLEXISTS 述部、および XMLTable 表関数のパラメーターを明示的に指定する必要はありません。
- デフォルトでは、XMLTable 表関数の列を指定する必要はありません。

#### XQuery から SQL へのパラメーターの引き渡し

- db2-fn:sqlquery 関数を使用して、XQuery 式中のパラメーターを SQL 全選択に渡すことができます。このパラメーターを使用して、XQuery 式によって実行される SQL 全選択に変更を加えたり、XQuery 式に戻されるデータに変更を加えたりできます。

#### 関連概念

pureXML ガイドの『XMLEXISTS および XMLQUERY への定数およびパラメーター・マーカの引き渡し』

pureXML ガイドの『XMLEXISTS、XMLQUERY、または XMLTABLE を使用した列名の簡単な引き渡し』

pureXML ガイドの『XQuery から SQL へのパラメーターの引き渡し』

#### 関連情報

XQuery リファレンスの『sqlquery 関数』

---

## 非 Unicode データベースに XML データを保管できる

非 Unicode データベースに XML データを保管できるようになりました。

DB2 データベース・マネージャーは XML データを内部では Unicode として保管するため、バージョン 9.5 より前は pureXML のフィーチャーは UTF-8 でエンコードされたデータベースのみで使用できました。新機能では、Unicode データベースをもはや必要としないようにコード・ページ変換が管理されます。ただし、XML データは、コード・ページ変換をしないような形式でのみ非 Unicode データベースに挿入してください (BIT DATA、BLOB、XML など)。データに置換文字が入らないようにするため、データベースのコード・ページに含まれているコード・ポイントを含む XML データのみを挿入してください。

新規 `enable_xmlchar` 構成パラメーターを使用すると、SQL ストリング・データがクライアントのコード・ページからデータベースのコード・ページに変換され、その後内部保管のために Unicode に変換される時、文字置換が発生しないようにすることができます。 `enable_xmlchar` を NO に設定すると、XML 構文解析中の文字データ・タイプの使用がブロックされるため、文字置換が発生せず、保管 XML データの健全性が保たれます。デフォルトでは、文字データ・タイプの構文解析が許可されるように、`enable_xmlchar` は YES に設定されています。

非 Unicode XML データベースは、他のデータベースと同様に、コントロール・センターを使用して管理できます。

#### 関連概念

pureXML ガイドの『非 Unicode データベースでの XML の使用』

---

## パフォーマンス向上のために小規模な XML 文書が基本表の行に格納できる

32 KB 以下のサイズの XML 文書には、追加のストレージ選択が可能です。XML 列を表に追加するか、または既存の XML 列を変更する場合、デフォルトの XML ストレージ・オブジェクトではなく、基本表の行にこれらの文書を格納できます。

XML 文書の行の格納は、構造化タイプのインスタンスが表の行にインラインで格納される仕方と似ており、ユーザーの制御下にあります。ユーザーがどのストレージ選択をするかは、ストレージとパフォーマンスの要件に応じて異なります。大規模な文書は必ずデフォルトの XML ストレージ・オブジェクトに格納する必要がありますが、主に小規模な文書を扱っている場合は、基本表の行ストレージには以下の利点があります。

- XML 文書を照会、挿入、更新、および削除する操作におけるパフォーマンスの向上。基本表の行に格納される文書では必要とされる入出力操作が少なくなるからです。
- データ行圧縮も使用している場合、XML 文書のストレージ・スペース要件の削減と入出力効率の向上。

このオプションを使用するには、`INLINE LENGTH` キーワードを `CREATE TABLE` および `ALTER TABLE` ステートメントと共に組み込み、基本表の行に格納する XML 文書の最大サイズを続けて指定します。32 KB を超える大きさの XML 文書を基本表の行ストレージに使用できる XML 表列に格納する場合、サイズ超過の文書はデフォルトの XML ストレージ・オブジェクトに透過的に格納されます。

#### 関連概念

pureXML ガイドの『XML ストレージの概要』

pureXML ガイドの『XML の基本表行保管』

---

## XML 文書の妥当性検査の再実行を必要とせずに XML スキーマを更新できる

XML スキーマ・リポジトリに登録された XML スキーマを、以前に挿入された XML 文書と新規の XML 文書の両方がそのスキーマの改良されたバージョンに対して妥当性検査されるように更新できるようになりました。

UPDATE XMLSCHEMA コマンドおよび XSR\_UPDATE ストアド・プロシージャが追加され、既に登録されている XML スキーマを変更できるようになりました。これは、例えば、オプションの要素または属性を既存のスキーマに追加すると共に、更新されたスキーマを新規および以前に保管された XML インスタンス文書両方の妥当性検査で使用できるようにするような場合に役立ちます。

XML スキーマを改良するための主な要件は、元のスキーマと新規スキーマに互換性があることで、そのため、以前に妥当性検査された XML 文書はそのまま有効となり、元のスキーマのタイプ・アノテーションは変更されません。更新処理中、元のスキーマと新規スキーマの互換性を確認するために検査が行われます。スキーマに互換性がない場合には、更新操作は失敗します。

### 関連概念

pureXML ガイドの『シナリオ: XML スキーマの展開』

### 関連タスク

pureXML ガイドの『XML スキーマの展開』

### 関連資料

pureXML ガイドの『XML スキーマの展開のための互換性要件』

---

## XQuery の upper-case および lower-case 関数はロケールをサポートする

XQuery 関数の fn:upper-case および fn:lower-case は、ロケールを認識した変換を使用して大/小文字を変更できるようになりました。デフォルトでは、fn:upper-case および fn:lower-case は、Unicode 標準に基づいてテキストの大/小文字を変換します。一部の文字では、ロケール・ベースの変換を使用した場合と Unicode 標準を使用した場合とで、大文字と小文字とのマッピングが異なる場合があります。

例えば、トルコ語では、文字 *i* または *I* (ローマ字 *I* の大文字または小文字) を変更する場合、ドットが文字に残ります。トルコ語のロケール `tr_TR` を指定すると、fn:upper-case は、文字 *i* を、ドットが上部に付いたローマ字大文字 *I* に変換します。数字参照は `&#x130;` です。トルコ語のロケールを指定すると、fn:lower-case は、文字 *I* を、ローマ字小文字のドットなし *i* に変換します。数字参照は `&#x131;` です。ロケールを指定しないと、fn:upper-case は文字 *i* を *I* に、fn:lower-case は文字 *I* を *i* に変換します。

### 関連情報

XQuery リファレンスの『小文字関数』

XQuery リファレンスの『upper-case 関数』



---

## XQuery 関数は日付と時間から構成要素を抽出し、日付と時間を調整する

XQuery の日付と時間の構成要素抽出関数を使用して、日付、時間、日時、および期間の各データ・タイプを抽出できるようになりました。XQuery の時間帯調整関数を使用して、日付と時間を指定の時間帯に調整したり、日付、時間、または日時データ・タイプから時間帯の構成要素を除去したりできるようになりました。

例えば、日付抽出関数を使用して、日付データ・タイプから年や月の構成要素を簡単に抽出することができます。時間帯調整関数を使用して、日付または時間をある時間帯から別の時間帯に簡単に変換することができます。

### 関連情報

XQuery リファレンスの『カテゴリ別の関数』

---

## XQuery キャスト可能式が値のキャストのテストをサポートする

XQuery 言語のキャスト可能式により、XQuery データ・タイプへの値のキャストをテストすることができます。

キャスト可能式を XPath 述部として使用して、評価時のエラーを避けることができます。さらに、キャスト可能式を使用して、特定の値の処理時の適切なデータ・タイプを選択することができます。

### 関連情報

XQuery リファレンスの『キャスト可能な式』

---

## 発行関数を使いやすくなっている

リレーショナル・データと XML とのマッピングに新しい発行スカラー関数を使用できるようになりました。これらの関数に必要なオプションは以前の SQL/XML 発行関数より少なく、デフォルトの動作で、ISO SQL/XML 2006 で定義されている多数の規則を活用し、最も一般的に必要なオプションをサポートします。

新しい発行関数は以下のとおりです。

### XMLGROUP

この関数は、表または照会の結果を表す 1 つの最上位エレメントを戻します。デフォルトでは、結果セット中の各行は行サブエレメントにマップされ、各入力式は行サブエレメントのサブエレメントにマップされます。オプションで、各入力式を行サブエレメントの属性にマップできます。

### XMLROW

この関数は、表または照会の結果を表す行エレメントのシーケンスを戻します。デフォルトでは、各入力式は行エレメントのサブエレメントに変換されます。オプションで、各入力式を行エレメントの属性に変換できます。

### XSLTRANSFORM

この関数は、提供されたスタイル・シートに基づいて、XML 文書を HTML、プレーン・テキスト、または他の XML 形式に変換します。XSLT トランスフォーマーメーションは、XML データを他の形式にトランスフォームするために使用される標準的な方法で、単一のデータ・ソースから複数の出力形式を生成することを可能にします。

## 関連資料

SQL リファレンス 第 1 巻の『XMLROW スカラー関数』

SQL リファレンス 第 1 巻の『XMLGROUP 集約関数』

pureXML ガイドの『XSLTRANSFORM スカラー関数』

---

# アノテーション付き XML スキーマ分解での挿入の順序と再帰的スキーマの登録のサポート

pureXML の分解機能に 2 つの新しいフィーチャーが追加され、挿入の順序と再帰的 XML スキーマの登録が制御できるようになりました。

## 挿入の順序のサポート

新しい XML スキーマ・アノテーションを使用すると、決定可能な順序で XML 文書の内容がターゲット表の行に挿入されるように、分解階層を指定できます。バージョン 9.5 より前では、分解プロセスによるデータを行としてターゲット表に挿入する順序を制御する機能はなかったため、ターゲット表の整合性要件を満たすようにデータを挿入することは困難である場合がありました。新しいフィーチャーにより、XML 文書の分解中、リレーショナル・スキーマで定義された参照整合性制約が確実に守られるようになります。

## 再帰的スキーマの登録

バージョン 9.5 より前では、XML インスタンス文書の再帰的セクションを分解する必要がない場合でも、再帰が含まれる XML スキーマを登録することはできませんでした。しかし、再帰が含まれる XML スキーマを XML スキーマ・リポジトリ (XSR) に登録して分解可能な状態にすることができるようになりました。関連する XML インスタンス文書の再帰的セクションは、スカラー値としてターゲット表に分解できません。ただし、適切なスキーマ・アノテーションを使用することにより、再帰的セクションを保管して、後でシリアルライズされたマークアップとして取り出すことができます。

### 関連概念

pureXML ガイドの『アノテーション付き XML スキーマ分解と再帰的 XML 文書』

### 関連資料

pureXML ガイドの『db2-xdb:rowSetOperationOrder 分解アノテーション』

pureXML ガイドの『db2-xdb:order 分解アノテーション』

## 第 9 章 アプリケーション開発の機能拡張

アプリケーション開発の改善には、データベース・アプリケーション開発を簡素化し、アプリケーションの移植性を改善し、さらにアプリケーション・デプロイメントを容易にする新規フィーチャーおよび機能拡張が含まれます。

このセクションでは、バージョン 9.5 で導入されているアプリケーション開発の機能拡張について詳しく説明します。

### ID の長さ制限の増加

ID のサポート拡大により、他の DBMS ベンダー製アプリケーションの移植がさらに容易になります。また、ID を短くする必要がないため、データ定義言語 (DDL) のマイグレーションがさらに容易になるはずです。

最大長が長くなった ID は、以下の表にリストされています。

表 3. バージョン 9.1 および 9.5 での ID の長さの制限

ID 名	バージョン 9.1 での長さ (バイト)	バージョン 9.5 での長さ (バイト)
属性	18	128
許可 ID (Authid)	30	128
列	30	128
制約	18	128
カーソル	18	128
データベース・パーティション・グループ	18	128
イベント・モニター	18	128
グループ	30	128
パッケージ	8	128
スキーマ	30	128
特定の名前	18	128
SQL パス (FUNCPATH BIND オプションおよび CURRENT PATH 特殊レジスタによる指定)	254	2048
ステートメント	18	128
トリガー	18	128
ユーザー定義タイプ	18	128

現在でも SQLDA は、ユーザー定義タイプ (UDT) では 8 バイトのスキーマ名、UDT では 18 バイトの名前、さらに列では 30 バイトの名前に制限されているため、128 バイトの制限は非組み込み SQL のみに適用される点に注意してください。

128 バイトの制限は、データベース・マネージャーによりシステム・カタログに保管される制限を指します。アプリケーション内で ID を示すために使用されるコード・ページはさまざまである場合があるため、アプリケーション側の制限は定義されていません。アプリケーションのコード・ページに関係なく、アプリケーションおよびサーバー両方の側の DB2 ユーティリティーが 128 バイトの制限を使用します。

役立つサンプル・ファイル、checkv9limits が、samples/admin\_scripts にあります。このファイルを使用すると、データベースで、より大きいバージョン 9.5 の制限を使用している可能性がある ID を見つけることができます。

#### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『SQL および XML 制限』

---

## DB2 インストールに統合された PHP 拡張 (Linux、AIX、および Windows)

バージョン 9.5 では、DB2 Data Server Client (以前の DB2 Client) に Hypertext Preprocessor (PHP) 拡張が付属しているため、それらをダウンロードする必要はなくなりました。さらに、バージョン 9.5 は PDO\_IBM と呼ばれる新しい拡張を提供することにより、バージョン 9.1 PHP サポート上に構築されます。

新しい PDO\_IBM 拡張および既存の IBM\_DB2 拡張は、DB2 データベースに保管されたデータに PHP アプリケーションを介して堅固なアクセスができるようにし、PHP アプリケーションを素早く簡単にデプロイできるようにする、小さなフットプリントの拡張です。これらの拡張の間の相違は、以下のとおりです。

#### PDO\_IBM

この新しい拡張は、PHP Data Objects (PDO) インターフェースにより、DB2 データベースへのアクセスを提供します。PDO は、データにアクセスするための汎用のオブジェクト指向インターフェースを提供します。この拡張によって、PDO を使用するために ODBC データ・ソースを作成する必要がなくなりました。PDO\_IBM を使用して、DB2 データベース・マネージャーへのカタログ接続またはダイレクト TCP/IP 接続を使用することにより、データベースに接続できます。

#### IBM\_DB2

この拡張は、DB2 コール・レベル・インターフェース (CLI) ライブラリーを使用して、DB2 データベースに保管されたデータへの直接アクセスを提供します。この拡張のインターフェースは DB2 製品に固有のもので、他の拡張では使用可能でない、いくつかのより高度な DB2 フィーチャーを使用します。IBM\_DB2 拡張は、データベース・メタデータへの広範囲なアクセスを可能にするアプリケーション・プログラミング・インターフェース (API) を提供します。

DB2 インストール・プログラムには、以下のオペレーティング・システム上の PHP 拡張が含まれます。

- AIX
- Linux on x86

- Linux on AMD64 および Linux on EM64T
- Linux on POWER™ (PowerPC® および pSeries®)
- Windows on x86

#### 関連情報

データベース・アプリケーション 開発の基礎の『DB2 のための PHP アプリケーション開発の概要』

---

## DB2 インストールに統合された Ruby on Rails フレームワーク・サポート (Linux、AIX、および Windows)

Ruby on Rails アプリケーション用に、DB2 Web アプリケーションの素早い開発およびデプロイメントが使用可能になり、拡張されています。

Rails アダプター (IBM\_DB) および Ruby ドライバーが IBM によって開発されており、DB2 Database for Linux, UNIX, and Windows、DB2 for i5/OS (DB2 Connect を使用)、DB2 for z/OS (DB2 Connect を使用) を含む、すべての DB2 データ・サーバーに合わせて最適化されています。Ruby on Rails を使用可能にしてサポートを提供しているベンダーは、IBM だけです。

Rails フレームワークでさまざまなデータ・ストアを構成できますが、IBM DB2 データ・サーバーには固有の利点があります。IBM\_DB Rails アダプターおよび Ruby ドライバーを DB2 データ・サーバーと共に使用することにより、pureXML データを操作できるようになります。さらに、Rails フレームワークはモデル・リリースシップ、rake タスク、マイグレーション、scaffolding、および統合テスト環境などのフィーチャーを提供し、これらによって繰り返し、迅速なアプリケーション開発を実現できます。

迅速なアプリケーション開発およびデプロイメントを使用可能にするために、新しい DB2 Ruby ドライバーおよび Rails アダプターが、以下のオペレーティング・システム上の DB2 インストール・ディレクトリーに備わっています。

- AIX (APAR IZ01456 が適用されている必要があります)
- Linux on x86
- Linux on AMD64 および Linux on EM64T
- Linux on POWER (PowerPC および pSeries)
- Windows on x86

DB2 Ruby ドライバーおよび Rails アダプターは、RubyForge Rails Adapter/Driver for IBM Databases Web サイトでも入手可能です。

#### 関連概念

データベース・アプリケーション 開発の基礎の『IBM\_DB Ruby ドライバーおよび Rails アダプター』

---

## pureXML およびマルチバイト文字をサポートする Perl ドライバー

DB2 Perl ドライバーは、pureXML サポートおよびマルチバイト・ロケール・サポートの 2 つの新しい重要なフィーチャーが追加されて機能強化されています。これらの新しいフィーチャーによって、DB2 Perl ドライバーを介してデータに対するより直接的なアクセスが可能になります。

新しいフィーチャーは、アプリケーションとデータベースとの間のより透過的な通信を提供することにより、アプリケーション・ロジックを削減するために役立ちます。これらの新しいフィーチャーに関する詳細は、以下のとおりです。

### pureXML サポート

pureXML サポートにより XML 文書を DB2 データベースに直接挿入できます。XML データをデータベースに挿入すると pureXML パーサーが自動的に実行するので、アプリケーションが XML 文書を構文解析する必要はなくなりました。アプリケーションの外部で文書の構文解析が扱われることにより、アプリケーション・パフォーマンスは向上して、保守作業は減少します。XML で保管されたデータを DB2 Perl ドライバーを使用して取り出すことも同様に容易です。BLOB またはレコードを使用してデータにアクセスできます。

### マルチバイト文字セットのサポート

このフィーチャーは、アプリケーションと DB2 データベースとの間の、より透過的なインターフェースを提供します。Perl アプリケーションは、DB2 データベースと対話する前に文字セットの間の変換を行う必要がなくなりました。変換の必要性がなくなるので、結果としてアプリケーションのフットプリントが小さくなり、保守の必要が縮小してエラーの可能性が小さくなります。

最新の DB2 Perl ドライバーをダウンロードする方法については、<http://www.ibm.com/software/data/db2/perl/> Web サイトを参照してください。

### 関連概念

Perl および PHP アプリケーションの開発の『Perl でのプログラミングに関する考慮事項』

---

## 拡張された Visual Studio 2005 用の IBM Database Add-in

迅速なアプリケーション開発、データベース・スキーマ開発、およびデバッグを行うためのツールを提供する Visual Studio 2005 用の IBM Database Add-In は、バージョン 9.5 でさらに優れたサポートを提供します。

以下の機能拡張があります。

- Visual Studio 2005 用の IBM Database Add-in は、IBM Informix<sup>®</sup> Dynamic Server、IBM UniData<sup>®</sup>、および IBM UniVerse<sup>®</sup> をサポートするようになりました。このサポートにより、DB2、Informix、UniData、または UniVerse 製品のどのような組み合わせで作業するときでも、同じアドイン・ツールを使用することが可能になります。特に、.NET アプリケーションは以下の IBM データ・サーバーにアクセスできるようになりました。
  - DB2 Universal Database<sup>™</sup> for Linux, UNIX, and Windows バージョン 8.1 以降

- DB2 Universal Database for z/OS or OS/390<sup>®</sup> バージョン 6 および 7
- DB2 Universal Database for z/OS バージョン 8 以降
- DB2 Universal Database for iSeries (OS/400<sup>®</sup> バージョン 5 リリース 1 以降を使用)
- IDS バージョン 11.1
- UniData 7.1 以降
- UniVerse 10.2 以降

フィーチャーが使用可能かどうか、およびデータ・タイプの互換性は、使用されるデータ・サーバーに依存します。例えば、Informix Dynamic Server、UniData、および UniVerse データ・サーバーは XML データ・タイプをサポートしていません。

- 新しい IBM Function Designer を使用すると、関数をより簡単に処理できます。デザイナーを使用すると、以下のアクションを実行できます。
  - 関数の作成および変更
  - 関数のロールの作成と変更、およびアクセス権の定義
  - 関数の複製
  - すべての関数のスクリプトの表示または作成
- DB2 サーバーへのデータ接続を定義すると、表タイプに基づいてデータベース表をフィルターに掛けることができます。オプションにより、P (物理) や L (論理) および他のプラットフォーム特定およびデータベース特定の表タイプなどの表タイプを選択できます。
- アドインを使用して、データベース操作 (SQL SELECT および DML ステートメント、XQuery 式、またはストアド・プロシージャの呼び出し) をクライアント・アプリケーションに公開する Web サービスを作成できるようになりました。さらに、アドインを使用して、Web サービスを Web サーバーにデプロイしたり Web サービスを Web サーバーからアンデプロイしたりすることができます。
- XML ツールは、以下の機能をサポートするように拡張されました。
  - IBM XML Schema Mapping Designer 内の XML スキーマに注釈を付ける
    - 実行中の Web サービス・メソッドの結果をマッピングの XML スキーマ・ソースとして選択する
    - マッピング用の表を Server Explorer からデザイナー内のマッピング・エディターにドラッグする
    - 実行中の Web サービス・メソッドの結果を使用して、XML スキーマに注釈を付けた後にマップ・リンクをテストする
  - XML スキーマ・リポジトリ内のアノテーション付き XML スキーマ用に Web サービス・メソッドを生成する
  - クライアント・サイドおよびサーバー・サイドの XML スキーマ妥当性検査コードを生成する
  - XML スキーマの 2 つのバージョンを比較して、それらの相違を検討する
  - クライアント・サイドおよびサーバー・サイドの XSL トランスフォーメーション・コードを生成する

#### 関連概念

## グローバル変数による SQL ステートメント間のデータ共有の改善

バージョン 9.5 では、グローバル変数の概念を取り入れています。これは、SQL ステートメントを介してアクセスおよび変更できる名前付きメモリー変数です。グローバル変数を使用すると、同じセッション (または接続) で実行される異なる SQL ステートメント間でデータを共有できます。しかも、このデータ転送をサポートするためのアプリケーション・ロジックは必要ありません。

あるステートメントの出力引数 (ホスト変数など) から別のステートメントの入力引数に値をコピーするようなステートメントを発行するアプリケーションは必要なくなりました。さらに、データベース・システム自体に含まれる SQL ステートメント (トリガーおよびビューを定義するステートメントなど) は、この共有情報にアクセスできるようになりました。

グローバル変数は、アプリケーションまたは SQL プロシーチャーの内部にサポートするロジックを置かなくて済むように、データ転送用のより複雑な対話式モデルをデータベース自体の内部に実装する上で役立ちます。グローバル変数に関連付けられた定義済み特権により、転送データのセキュリティは、アプリケーション・ロジックを使用した実行方法に委ねられることがないようにします。セキュリティが問題となっている場合には、GRANT および REVOKE ステートメントによってグローバル変数へのアクセス権限を制御できます。

グローバル変数は、静的データ、セッション中にほとんど変更されないデータ、または管理制御を行うデータを保管する上で特に役立ちます。そのようなデータの例には、DBA にアラートを送るために使用されるポケットベル番号や、特定のトリガーが有効になっているか無効になっているかの標識があります。

バージョン 9.5 は、作成されたセッション・グローバル変数をサポートします。セッション・グローバル変数は、特定のセッションに関連付けられていて、そのセッションに固有の値を持っています。作成されたセッション・グローバル変数は、その変数を定義したデータベースに対して実行されるアクティブな SQL ステートメントで使用できます。システム・カタログには、作成されたセッション・グローバル変数と、それらに関連付けられた特権の定義が含まれています。

### 関連資料

SQL リファレンス 第 1 巻の『ID』

SQL リファレンス 第 2 巻の『CREATE VARIABLE ステートメント』

SQL リファレンス 第 2 巻の『GRANT (グローバル変数特権) ステートメント』

SQL リファレンス 第 2 巻の『REVOKE (グローバル変数特権) ステートメント』

SQL リファレンス 第 1 巻の『グローバル変数』



---

## SET 変数は動的に準備できる実行可能ステートメントとなった

バージョン 9.5 では、ホスト変数、バインド変数、グローバル変数、およびローカル変数を、トリガー内や関数内、プロシージャや独立ステートメントとして設定するための一貫した方法を提供する、SET ステートメントが導入しています。SQL ステートメントは動的に準備して実行することができ、実質的に単一行の動的カーソルを不要なものとしします。

SET 変数ステートメントは、値を変数に割り当てます。バージョン 9.5 より前には、このステートメントは、動的コンパウンド・ステートメント、トリガー、SQL 関数、SQL メソッド、SQL プロシージャ、および UPDATE ステートメント内だけで許可されました。

SET 変数ステートメントは、アプリケーション・プログラムに組み込んだり、対話式に発行することが可能となり、動的に準備できる実行可能ステートメントであるので、SET 変数の構文全体は、アプリケーション内および SQL ストアド・プロシージャ内のコマンド行プロセッサによりサポートされます。例えば、SQL ストアド・プロシージャ内で SET (a, b) = (1, 2) などの複数ターゲットの SET 変数ステートメントを使用できるようになりました。

プリコンパイル (prep) 時に入手不可のデータを取得するためにカーソルを使用する必要はなくなりました。バージョン 9.5 より前には、以下の例で示すように、SQL プロシージャで表データをホスト変数にコピーするには、SELECT INTO ステートメントを使用する必要がありました。

```
SELECT c0, c1 INTO :hv1, :hv2 FROM ...
```

ただし、SELECT INTO ステートメントは静的 SQL にのみ有効であり、動的には準備できません。そのため、SELECT ステートメントが準備時に入手不可のデータに依存している場合、過去においては以下のようにする必要がありました。

```
DECLARE vsq1 VARCHAR(254)
DECLARE c0 CURSOR FOR vstmt
DECLARE vstmt STATEMENT

SET vsq1 = 'select statement'
PREPARE vstmt FROM vsq1
OPEN c0
FETCH c0 INTO var
```

現在では、以下のようにすることができます。

```
SET vsq1 = 'SET (?, ?, ?) = (select statement)'
PREPARE vstmt FROM vsq1
EXECUTE vstmt INTO a, b, c USING x, y, z
```

### 関連資料

SQL リファレンス 第 2 巻の『SET 変数ステートメント』

---

## 配列のサポートによるアプリケーション移植性の向上

バージョン 9.5 は、ARRAY コレクション・データ・タイプをサポートします。配列とは、ストアード・プロシージャおよびアプリケーションで操作することはできても、表に保管できない一時値のことです。このフィーチャーを使用すると、既に配列をサポートする他のデータベース・ベンダー製のアプリケーションおよびストアード・プロシージャの移植が容易になります。

配列を使用すると、アプリケーション間およびストアード・プロシージャ間でデータの集合を効率的に受け渡し、リレーショナル表を使用しなくても SQL プロシージャ内で一時的なデータの集合を保管および操作できます。SQL プロシージャ内で使用できる演算子を配列に対して使用すると、データを効率的に保管および検索することができます。

バージョン 9.5 の配列データ・タイプのサポートにより、以下のことが可能になります。

- 配列に基づいたユーザー定義タイプを作成する。例えば、CREATE TYPE INT10 AS INTEGER ARRAY[10] は、最大 10 の整数値を持つ配列のタイプを定義します。
- ストアード・プロシージャおよびアプリケーションでの配列型の変数およびパラメーターを宣言する。
- 配列値を作成および操作する。配列操作の基本的なものには、配列コンストラクター、副索引、エレメント数、およびトリムが含まれます。
- JDBC と CLI アプリケーション間、および SQL と Java ストアード・プロシージャ間で配列をやり取りする。
- 配列を表に変換して (表の行ごとに 1 つの配列エレメント)、配列と SQL の間のインターフェースが容易になるように列を配列に集約する。
- 入出力配列パラメーターを持つプロシージャをコマンド行プロセッサから呼び出す。

### 関連資料

SQL リファレンス 第 1 巻の『ユーザー定義タイプ』

SQL リファレンス 第 2 巻の『CREATE TYPE (配列) ステートメント』

---

## 10 進浮動小数点データ・タイプにより 10 進数データの正確性とパフォーマンスが向上している

バージョン 9.5 では DECFLOAT が導入されています。これは 10 進浮動小数点データ・タイプで、正確な 10 進値を扱うビジネス・アプリケーション (金融アプリケーションなど) にとって有用です。

この種のアプリケーションでは、10 進数データのバイナリーの近似値を提供するバイナリー浮動小数点データ・タイプ (REAL および DOUBLE) は適切ではありません。DECFLOAT は、DECIMAL の正確性と FLOAT のパフォーマンス上の長所の一部を組み合わせたもので、貨幣の値が扱われるアプリケーションにとって利点があります。例えば、\$0.70 の通話料に対する 5% の税金を計算するのに、価格を含む列を REAL と定義すると結果の計算値は 0.734999987483025 になり、列を DECFLOAT(16) と定義すると値は 0.7350 になります。

バージョン 9.5 は、新しい IBM POWER6™ ハードウェア上で DECFLOAT をサポートします。IBM Power 6 では、DB2 バージョン 9.5 は、以前は正規の浮動小数点数演算専用だった 10 進浮動小数点ハードウェア・アクセラレーションを活用します。バージョン 9.5 は、IEEE 標準で指定されている 16 桁と 34 桁の 10 進浮動小数点エンコード方式をインプリメントしています。以下の表は、各エンコード方式の精度、長さ、および範囲を示しています。

表 4. サポートされている 10 進浮動小数点エンコード方式の精度、長さ、および範囲

桁数の精度	長さ (ビット単位)	範囲
16	64	-9.99999999999999 x 10 <sup>384</sup> から -1 x 10 <sup>-383</sup> まで: 1 x 10 <sup>-383</sup> から 9.99999999999999 x 10 <sup>384</sup> まで
34	128	-9.999 x 10 <sup>6144</sup> から -1 x 10 <sup>-6143</sup> まで: 1 x 10 <sup>-6143</sup> から 9.999 x 10 <sup>6144</sup> まで

このデータ・タイプは

ROUND\_HALF\_EVEN、ROUND\_HALF\_UP、ROUND\_DOWN、ROUND\_CEILING、および ROUND\_FLOOR の 5 つの IEEE 指定の丸めモードをサポートします。新しいデータベース構成パラメーター **decflt\_rounding** を使用して丸めモードを指定できます。

#### 関連資料

- SQL リファレンス 第 1 巻の『数値』
- SQL リファレンス 第 1 巻の『COMPARE\_DECFLOAT スカラー関数』
- SQL リファレンス 第 1 巻の『DECFLOAT スカラー関数』
- SQL リファレンス 第 1 巻の『NORMALIZE\_DECFLOAT スカラー関数』
- SQL リファレンス 第 1 巻の『QUANTIZE スカラー関数』
- SQL リファレンス 第 1 巻の『TOTALORDER スカラー関数』
- データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『decflt\_rounding - 10 進浮動小数点丸め構成パラメーター』

## JDBC および SQLJ のサポートの機能拡張

IBM Data Server Driver for JDBC and SQLJには、バージョン 9.5 の大幅な機能拡張が数多く含まれています。

バージョン 9.5 では、2 つのバージョンの IBM Data Server Driver for JDBC and SQLJ を使用できます。つまり JDBC 3.0 およびそれ以前の仕様の関数をサポートするバージョンと、JDBC 4.0 およびそれ以前の仕様の関数をサポートするバージョンです。以下の表は、ドライバーのバージョンがパッケージされている JAR ファイルと、それに対応するサポートのレベルをリストしています。

表 5. IBM Data Server Driver for JDBC and SQLJ のパッケージ

ドライバーのパッケージ名	JDBC サポートのレベル	Java 用 SDK の必須最小レベル
db2jcc.jar	JDBC 3.0 およびそれ以前	1.4.2
db2jcc4.jar	JDBC 4.0 およびそれ以前	6

このセクションでは、バージョン 9.5 で導入されている JDBC および SQLJ のサポートの機能拡張について詳しく説明します。

## JDBC 2.0 および JDBC 3.0 のサポートが拡張されている

JDBC 3.0 およびそれ以前の仕様のサポートを提供する IBM Data Server Driver for JDBC and SQLJ バージョン 3.50 には、多くの機能拡張が含まれています。

### ドライバー名が変更された

新規ドライバー名は IBM Data Server Driver for JDBC and SQLJ です。ただし、ドライバーのバージョン 3.50 では、`java.sql.DatabaseMetaData.getDriverName` メソッドは IBM DB2 JDBC Universal Driver Architecture を戻します。

### ライセンス・ファイルは不要になった

DB2 Database for Linux, UNIX, and Windows、Cloudscape、または Informix Dynamic Server (IDS) データ・ソースのアクセスには、ライセンス・ファイルは不要です。したがって、これらのデータベース・サーバーに接続するときは、`CLASSPATH` に `db2jcc_license_*.jar` ファイルを組み込む必要はなくなりました。

これは DB2 Connect ユーザーには適用されません。

### runJDBCBinder メソッドが代替として DB2Binder ユーティリティに追加されている

以前のバージョンの IBM Data Server Driver for JDBC and SQLJ では、DB2Binder コマンド行インターフェースだけが、IBM Data Server Driver for JDBC and SQLJ によりデータベース・サーバーで使用される DB2 パッケージのバインドに使用できました。runJDBCBinder メソッドは、同じタスクを実行するためのアプリケーション・プログラミング・インターフェースです。

runJDBCBinder メソッドは、BIND オプションと同等の以下のオプションをサポートします。

- **action** (addreplacelddrop); drop は DB2 for z/OS のみでサポートされます
- **blocking** (allnolunambig)
- **dbprotocol** (drdalprivate); dbprotocol は DB2 for z/OS のみでサポートされます
- **keepdynamic** (nolyes)
- **owner**
- **reopt** (nonelalwaysloncelauto)
- **size**
- **optprofile**; optprofile は DB2 Database for Linux, UNIX, and Windows のみでサポートされます

さらに、runJDBCBinder は size オプションをサポートします。size オプションは、各分離レベルおよびカーソル保持可能性についてバインドまたはドロップする内部 JDBC ドライバー・パッケージの数を指定します。

### DB2Binder ユーティリティが拡張されている

新規オプション: DB2Binder ユーティリティは、以下の新規オプションをサポートします。

**-action (drop)**

既存の IBM Data Server Driver for JDBC and SQLJ パッケージがドロップされていることを示します。

**-size (n)**

各分離レベルおよびカーソル保持可能性についてバインドまたはドロップする内部 IBM Data Server Driver for JDBC and SQLJ パッケージの数を指定します。

**-optprofile**

値が CURRENT OPTIMIZATION PROFILE 特殊レジスターで設定されていない場合に、IBM Data Server Driver for JDBC and SQLJ パッケージ内のすべての静的ステートメントに使用される最適化プロファイルを指定します。

**診断の向上:** 以前のリリースでは、DB2Binder ユーティリティは必ず戻りコード 0 を戻しました。現在では DB2Binder ユーティリティは、DB2Binder の処理が成功したかどうかを示す整数値を戻すようになりました。処理が成功しなかった場合、戻り値はエラーの性質を示します。

## XML データ・タイプの暗号化がサポートされる

IBM Data Server Driver for JDBC and SQLJ Type 4 接続では、アプリケーションが securityMechanism プロパティを ENCRYPTED\_USER\_AND\_DATA\_SECURITY または ENCRYPTED\_USER\_PASSWORD\_AND\_DATA\_SECURITY に設定している場合に、XML データのデータ暗号化をサポートするようになりました。

## 連続ストリーミングがサポートされる

IBM Data Server Driver for JDBC and SQLJ は、DB2 Database for Linux, UNIX, and Windows および DB2 for z/OS に接続する場合の LOB の連続ストリーミングをサポートするようになりました。連続ストリーミングはデフォルトでは使用可能になっています。

## DatabaseMetaData メソッドの長 ID がサポートされる

以下の DatabaseMetaData メソッドは、128 バイトの ID に正しい長さを戻すようになりました。

- getMaxColumnNameLength
- getMaxCursorNameLength
- getMaxSchemaNameLength

## 新規トレース・レベルが追加された

Connection または DataSource プロパティの traceLevel の新規値である TRACE\_TRACEPOINTS は、ドライバーの内部トレース・ポイントをトレースするかどうかを決定します。TRACE\_TRACEPOINTS 値を設定する場合、ドライバーの内部トレース・ポイントは、接続上で有効になっている LogWriter に出力されます。他の traceLevel 値と同様、TRACE\_TRACEPOINTS は OR ロジックにより他の値と結合させることができます。

## 動的両方向スクロール・カーソルのための ResultSet および DatabaseMetaData メソッドが追加された

以下の JDBC 2.0 java.sql.ResultSet メソッドがサポートされるようになりました。

### ResultSet.insertRow

挿入行の内容を ResultSet オブジェクトと表とに挿入します。

### ResultSet.moveToInsertRow

カーソルを ResultSet オブジェクトの挿入行に移動させます。

### ResultSet.moveToCurrentRow

挿入行にあるカーソルを ResultSet オブジェクト内の以前のカーソル位置に移動させます。

### ResultSet.rowInserted

ResultSet オブジェクト内の現在行が挿入されたものかどうかを判別します。

以下の JDBC 2.0 java.sql.DatabaseMetaData メソッドがサポートされるようになりました。

### DatabaseMetaData.ownInsertsAreVisible

ResultSet オブジェクトによって基礎表に挿入された行が ResultSet に可視であるかどうかを判別します。

### DatabaseMetaData.othersInsertsAreVisible

他のアプリケーションまたは他の ResultSet オブジェクトによって基礎表に挿入された行が、指定された ResultSet に可視であるかどうかを判別します。

### DatabaseMetaData.insertsAreDetected

ResultSet オブジェクトが挿入行を検出できるかどうかを判別します。

## 配列がサポートされる

バージョン 9.5 では、配列をストアード・プロシージャの入力または出力パラメーターとしてサポートします。JDBC には、それらのストアード・プロシージャを呼び出すクライアント・プログラム内の配列パラメーターを検索または更新するための、対応するサポートがあります。したがって、IBM Data Server Driver for JDBC and SQLJ は、IBM Data Server Driver for JDBC and SQLJ 専用のインターフェースである com.ibm.db2.jcc.DB2Array を導入しています。

出力配列パラメーターの内容は、以下の方法で検索することができます。

- java.lang.Object であれば、DB2Array.getArray メソッドを使用する
- java.sql.ResultSet であれば、DB2Array.getResultSet メソッドを使用する

入力配列パラメーターは、以下の方法で更新することができます。

- PreparedStatement.setArray メソッドの使用
- PreparedStatement.setObject メソッドの使用

## 10 進浮動小数点データ・タイプがサポートされる

バージョン 9.5 は、10 進浮動小数点データの格納用に DECFLOAT SQL タイプをサポートします。IBM Data Server Driver for JDBC and SQLJ を使用するアプリケーションは、バージョン 9.5 データベース内の 10 進浮動小数点データを格納および検索するようになりました。

DB2 Database for Linux, UNIX, and Windows は、CURRENT DECFLOAT ROUNDING MODE 特殊レジスターを使用して、10 進または 10 進浮動小数点値に使用されるデフォルトの丸めモードを指定します。decimalRoundingMode Connection または DataSource プロパティを使用して、特殊レジスターが設定されていない場合はそれを設定することができます。

## クライアント・リルトの代替サーバー情報のリポジトリとしてドメイン・ネーム・システム (DNS) がサポートされる

DB2 Database for Linux, UNIX, and Windows サーバーへの接続中のクライアント・リルトでは、代替サーバー情報のリポジトリとして、JNDI ディレクトリーの代わりに DNS を使用できます。

複数の IP アドレスを DNS 項目内に指定することができます。クライアント・リルトには、2 つを指定することができます。1 つは 1 次サーバー用で、もう 1 つは 2 次サーバー用です。JNDI が構成されていない場合、IBM Data Server Driver for JDBC and SQLJ は DNS アドレスを使用して、クライアント・リルト用のサーバーを識別します。

## DriverManager 接続用の接続コンセントレーターと Sysplex ワークロード・バランシング・サポートが追加された

IBM Data Server Driver for JDBC and SQLJ の以前のリリースでは、接続コンセントレーターと Sysplex ワークロード・バランシング・フィーチャーは、DataSource インターフェースで確立した接続に対してのみ使用できました。このフィーチャーは、DriverManager インターフェースで確立した接続にも使用できるようになりました。

## 明示的長さがない setXXXStream メソッドがサポートされる

setAsciiStream、setBinaryStream、または setCharacterStream メソッドの呼び出し時に、length パラメーターに -1 を指定できるようになりました。これを実行すると、IBM Data Server Driver for JDBC and SQLJ は入力ストリームがなくなるまでデータを入力します。

## Java アプリケーション・サポートが XML スキーマ更新用に追加された

DB2Connection.updateDB2XmlSchema メソッドは、ある XML スキーマを別の XML スキーマの内容で更新します。DB2Connection.updateDB2XmlSchema は、新規 SYSPROC.XSR\_UPDATE ストアド・プロシージャと同じ機能を実行します。

## Reader および InputStream オブジェクトを指定した PreparedStatement.setObject 呼び出しがサポートされる

PreparedStatement.setObject では、CLOB または XML 列への入力のための入力パラメーターのデータ・タイプは、Reader にすることができるようになりました。BLOB または XML 列への入力のための入力パラメーターのデータ・タイプは、InputStream にできるようになりました。データベース・サーバーがストリーミングをサポートしている場合、ドライバーはストリーミングを使用して、データをデータベース・サーバーに送信します。

## プロパティーが追加された

IBM Data Server Driver for JDBC and SQLJ には、以下の新規 Connection および DataSource プロパティーが組み込まれています。

- 以下の新しい 2 つの Connection および DataSource プロパティーにより、JDBC または SQLJ プログラムにおいて、接続レベルで最適化プロファイルの使用を制御することができます。

### optimizationProfile

SQLJ または JDBC プログラムに使用する最適化プロファイルを指定します。

### optimizationProfileToFlush

最適化プロファイル・キャッシュから除去する最適化プロファイルを指定します。

- currentDegree プロパティー

動的 SQL ステートメントの実行のためのパーティション内並列処理の度合いを指定する、CURRENT DEGREE 特殊レジスターを設定します。

- queryBlockSize プロパティー

データベース・サーバーがデータを戻すために使用する照会ブロックのサイズを指定します。

## javax\_jcc.jar ファイルが除去された

db2jcc\_javax.jar ファイルは、IBM Data Server Driver for JDBC and SQLJ には組み込まれなくなりました。したがって、IBM Data Server Driver for JDBC and SQLJ の CLASSPATH 環境変数には、db2jcc\_javax.jar ファイルを組み込む必要はなくなりました。

## オプティミスティック・ロックがサポートされる

バージョン 9.5 はオプティミスティック・ロックをサポートします。これは SQL データベース・アプリケーションが、行を選択した後、およびその行を更新または削除する前に、行に対するロックを解放するために使用できる技法です。IBM Data Server Driver for JDBC and SQLJ は、オプティミスティック・ロックをサポートする以下のメソッドを提供するようになりました。

### DB2Connection.prepareDB2OptimisticLockingQuery

オプティミスティック・ロック情報を要求できる PreparedStatement オブジェクトを作成します。



#### **DB2Statement.executeDB2OptimisticLockingQuery**

SELECT ステートメントを実行し、オプションでオプティミスティック・ロック列を戻すことを要求します。

#### **DB2ResultSetMetaData.getDB2OptimisticLockingColumns**

オプティミスティック・ロック列が `ResultSet` で使用可能かどうかに関する情報を戻します。

#### **DB2ResultSet.getDB2RowChangeToken**

オプティミスティック・ロックが要求された場合に現在行の行変更トークンを戻します。

#### **DB2ResultSet.getDB2RID**

オプティミスティック・ロックが要求された場合に現在行の RID 列値を戻します。

#### **DB2ResultSet.getDB2RIDType**

RID 列の基礎データ・タイプを戻します。

### **タイムアウト・メソッドがサポートされる**

DB2 Database for Linux, UNIX, and Windows バージョン 9.1 以降のデータベースに対する IBM Data Server Driver for JDBC and SQLJ Type 4 接続 の以下のメソッドのサポートが追加されました。

#### **javax.transaction.xa.XAResource.setTransactionTimeout**

`XAResource` インスタンスの現在のトランザクション・タイムアウト値を設定します。

#### **javax.transaction.xa.XAResource.getTransactionTimeout**

`XAResource` インスタンスの現在のトランザクション・タイムアウト値を入手します。

### **JDBC 3.0 メソッドが追加された**

以下の JDBC 3.0 メソッドがサポートされています。

#### **ResultSet.updateBlob**

更新可能結果セット内で SQL BLOB データ・タイプの値を更新します。

#### **ResultSet.updateClob**

更新可能結果セット内で SQL CLOB データ・タイプの値を更新します。

### **内部ステートメント・キャッシュがサポートされる**

IBM Data Server Driver for JDBC and SQLJ Type 4 接続 は、`PooledConnection` オブジェクトの内部ステートメント・キャッシュをサポートします。同じ物理 `PooledConnection` と関連付けられた複数の論理接続は、キャッシュからのステートメントを再使用できます。これによりパフォーマンスを向上させることができます。

`ConnectionPoolDataSource` で `maxStatements` プロパティを使用して、内部ステートメント・キャッシュを使用可能にして構成します。 `maxStatements` は、`PooledConnection` と関連付けられた内部ステートメント・キャッシュ内のドライバでオープンしたままにできるステートメントの最大数を指定します。

## エラー処理ユーザビリティが向上した

以下の機能拡張は、さらに有用な診断情報を提供します。

- さらに意味ある情報を提供するエラー・メッセージ・テキスト。  
`java.sql.SQLException.getMessage` および `java.sql.SQLWarning.getMessage` へのすべての呼び出しは、`SQLCODE` および `SQLSTATE` を戻すようになりました。IBM Data Server Driver for JDBC and SQLJ が引き起こしたエラーの場合、メッセージ・テキストにはドライバーのバージョンも含まれます。
- `SQLSTATE` が非 `NULL` である場合の警告の生成。以前のバージョンの IBM Data Server Driver for JDBC and SQLJ では、ドライバーまたはデータベース・サーバーが、0 の `SQLCODE` と非 `NULL` の `SQLSTATE` を戻した場合、ドライバーは警告を累積しませんでした。現在では、これらの条件下ではドライバーは警告を累積させて、ユーザーが `SQLSTATE` 情報を使用できるようにします。

## Informix Dynamic Server データベース・サーバーがサポートされる

IBM Data Server Driver for JDBC and SQLJ を使用して、Informix Dynamic Server (IDS) データベース・サーバーにアクセスできるようになりました。

## JDBC 4.0 サポートが追加された

IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 には多くの JDBC 4.0 機能が組み込まれています。これらの機能を使用するには、SDK for Java バージョン 6 が必要です。

### ドライブ名が変更された

JDBC および SQLJ の新規ドライバー名は、IBM Data Server Driver for JDBC and SQLJ です。 `java.sql.DatabaseMetaData.getDriverName` メソッドは、IBM DB2 JDBC Universal Driver Architecture の代わりにこの名前を戻します。

### JDBC 4.0 データ・タイプ・サポートが追加された

JDBC および SQLJ サポートが、ROWID または XML 列のデータの更新または取得のための以下の JDBC 4.0 インターフェースに提供されています。

- RowId。IBM Data Server Driver for JDBC and SQLJ 専用タイプ  
`com.ibm.db2.jcc.DB2Types.ROWID` およびクラス `com.ibm.db2.jcc.DB2RowId` は使用すべきではありません。
- SQLXML。IBM Data Server Driver for JDBC and SQLJ 専用タイプ  
`com.ibm.db2.jcc.DB2Types.DB2Xml` およびクラス `com.ibm.db2.jcc.DB2Xml` は使用すべきではありません。

### JDBC 4.0 メソッドがサポートされる

以下の JDBC 4.0 メソッドがサポートされています。

#### `java.sql.Array.free`

配列オブジェクトをクローズし、それが保持しているすべてのリソースを解放します。

**java.sql.Blob.free**

Blob オブジェクトをクローズし、それが保持しているすべてのリソースを解放します。

**java.sql.Blob.getBinaryStream**

Blob オブジェクトからバイナリー・ストリームとして値を取得します。

**java.sql.CallableStatement.getRowId**

SQL ROWID パラメーターの値を RowId オブジェクトとして取得します。

**java.sql.CallableStatement.getSQLXML**

SQL XML パラメーターの値を SQLXML オブジェクトとして取得します。

**java.sql.Clob.free**

Clob オブジェクトをクローズし、それが保持しているすべてのリソースを解放します。

**java.sql.Clob.getCharacterStream**

Clob オブジェクトからバイナリー・ストリームとして値を取得します。このメソッドの新しい形式では、最大で 2 GB の指定長をサポートします。

**java.sql.Connection.createBlob**

Blob オブジェクトを作成します。

**java.sql.Connection.createClob**

Clob オブジェクトを作成します。

**java.sql.Connection.createSQLXML**

SQLXML オブジェクトを作成します。

**java.sql.Connection.getClientInfo**

IBM Data Server Driver for JDBC and SQLJ がサポートするクライアント情報プロパティーに関する情報を戻します。これらのプロパティーは以下のとおりです。

- ApplicationName
- ClientAccountingInformation
- ClientHostname
- ClientUser

Connection.getClientInfo は、使用すべきでなくなった以下の IBM Data Server Driver for JDBC and SQLJ 専用メソッドと同じ機能を実行します。

- DB2Connection.getDB2ClientUser
- DB2Connection.getDB2ClientWorkstation
- DB2Connection.getDB2ClientApplicationInformation
- DB2Connection.getDB2ClientAccountingInformation

**java.sql.Connection.isValid**

接続がオープンしているかどうかを判別します。このメソッドは、使用すべきでなくなった IBM Data Server Driver for JDBC and SQLJ 専用メソッドの DB2Connection.isDB2Alive と同じ機能を実行します。

**java.sql.Connection.setClientInfo**

IBM Data Server Driver for JDBC and SQLJ がサポートするクライアント

情報プロパティの値を設定します。このメソッドは、使用すべきでなくなった以下の IBM Data Server Driver for JDBC and SQLJ 専用メソッドと同じ機能を実行します。

- `DB2Connection.setDB2ClientUser`
- `DB2Connection.setDB2ClientWorkstation`
- `DB2Connection.setDB2ClientApplicationInformation`
- `DB2Connection.setDB2ClientAccountingInformation`

#### **java.sql.DatabaseMetaData.getClientInfoProperties**

IBM Data Server Driver for JDBC and SQLJ がサポートするクライアント情報プロパティのリストを取得します。

#### **java.sql.DatabaseMetaData.getColumns**

結果セット内の追加の列 `IS_AUTOINCREMENT` を戻します。

#### **java.sql.DatabaseMetaData.getFunctions**

データベース・サーバーで定義されている組み込み関数およびユーザー定義関数の記述を戻します。

#### **java.sql.DatabaseMetaData.getFunctionColumns**

データベース・サーバーで定義されている、指定した組み込み関数およびユーザー定義関数のパラメーターに関する情報を取得します。

#### **java.sql.DatabaseMetaData.getProcedureColumns**

結果セット内の追加の列 `IS_AUTOINCREMENT` を戻します。

#### **java.sql.DatabaseMetaData.getProcedures**

結果セット内の追加の列 `SPECIFIC_NAME` を戻します。

#### **java.sql.DatabaseMetaData.getRowIdLifetime**

`ROWID` 値が有効である時間の長さを戻します。

#### **java.sql.DatabaseMetaData.getSchemas**

カタログおよびスキーマ・パターンの指定が可能な新規形式を持ちます。

#### **java.sql.PreparedStatement.setBlob**

ある値に入力パラメーターを設定し、その値をデータ・ソースに `BLOB` タイプとして送信すべきことをドライバーに通知します。

#### **java.sql.PreparedStatement.setAsciiStream**

ある値に入力パラメーターを設定し、その値をデータベース・サーバーに `VARCHAR` タイプとして送信すべきことをドライバーに通知します。このメソッドの新規形式は、最大で `2GB` の指定長、または指定長なしをサポートします。

#### **java.sql.PreparedStatement.setBinaryStream**

ある値に入力パラメーターを設定し、その値をデータベース・サーバーに `VARCHAR` タイプとして送信すべきことをドライバーに通知します。このメソッドの新規形式は、最大で `2GB` の指定長、または指定長なしをサポートします。

#### **java.sql.PreparedStatement.setCharacterStream**

ある値に入力パラメーターを設定し、その値をデータベース・サーバーに `BLOB` タイプとして送信すべきことをドライバーに通知します。このメソッドの新規形式は、最大で `2GB` の指定長、または指定長なしをサポートします。

**java.sql.PreparedStatement.setClob**

ある値に入力パラメーターを設定し、その値をデータベース・サーバーに CLOB タイプとして送信すべきことをドライバーに通知します。

**java.sql.PreparedStatement.setObject**

指定したオブジェクトを使用して、入力パラメーターを値に設定します。この既存のメソッドは、RowId および SQLXML オブジェクトをサポートするようになりました。

**java.sql.PreparedStatement.setRowId**

ある値に入力パラメーターを設定し、その値をデータベース・サーバーに ROWID タイプとして送信すべきことをドライバーに通知します。

**java.sql.PreparedStatement.setSQLXML**

ある値に入力パラメーターを設定し、その値をデータベース・サーバーに XML タイプとして送信すべきことをドライバーに通知します。

**java.sql.ResultSet.getRowId**

結果セットの ROWID 列から値を取得して RowId オブジェクトに入れます。

**java.sql.ResultSet.getSQLXML**

結果セットの XML 列から値を取得して SQLXML オブジェクトに入れます。

**java.sql.ResultSet.updateAsciiStream**

更新可能結果セットの文字列を更新します。このメソッドの新規形式は、最大で 2GB の指定長、または指定長なしをサポートします。

**java.sql.ResultSet.updateBinaryStream**

更新可能結果セットのバイナリー列を更新します。このメソッドの新規形式は、最大で 2GB の指定長、または指定長なしをサポートします。

**java.sql.ResultSet.updateCharacterStream**

更新可能結果セットの文字列を更新します。このメソッドの新規形式は、最大で 2GB の指定長、または指定長なしをサポートします。

**java.sql.ResultSet.updateBlob**

更新可能結果セット内で SQL BLOB データ・タイプの値を更新します。このメソッドの新規形式は、最大で 2GB の指定長、または指定長なしをサポートします。

**java.sql.ResultSet.updateClob**

更新可能結果セット内で SQL CLOB データ・タイプの値を更新します。このメソッドの新規形式は、最大で 2GB の指定長、または指定長なしをサポートします。

**java.sql.ResultSet.updateRowId**

更新可能結果セット内で SQL ROWID データ・タイプの値を更新します。

**java.sql.ResultSet.updateSQLXML**

更新可能結果セット内で SQL XML データ・タイプの値を更新します。

**java.sql.Statement.isClosed**

Statement オブジェクトがクローズしているかどうかを判別します。

- java.sql.Statement.setPoolable**  
Statement オブジェクトがプールされているかどうかを判別します。
- java.sql.Statement.isPoolable**  
Statement オブジェクトをプールできるかどうかを判別します。
- java.sql.SQLXML.free**  
SQLXML オブジェクトをクローズし、それが保持しているすべてのリソースを解放します。
- java.sql.SQLXML.getBinaryStream**  
SQLXML オブジェクトからバイナリー・ストリームとして値を取得します。
- java.sql.SQLXML.getCharacterStream**  
SQLXML オブジェクトから文字ストリームとして値を取得します。
- java.sql.SQLXML.getString**  
SQLXML オブジェクトからストリングとして値を取得します。
- java.sql.SQLXML.getSource**  
SQLXML オブジェクト内の XML 値を読み取るための `javax.xml.transform.Source` オブジェクトを戻します。
- java.sql.SQLXML.getBinaryStream**  
SQLXML オブジェクトをバイナリー・ストリーム値で初期化します。
- java.sql.SQLXML.getCharacterStream**  
SQLXML オブジェクトを文字ストリーム値で初期化します。
- java.sql.SQLXML.setResult**  
SQLXML オブジェクトを初期化する `javax.xml.transform.Result` オブジェクトを戻します。
- java.sql.SQLXML.setString**  
SQLXML オブジェクトを文字ストリング値で初期化します。
- javax.sql.PooledConnection.addStatementEventListener**  
StatementEventListener オブジェクトを PooledConnection オブジェクトと共に登録します。
- javax.sql.PooledConnection.removeStatementEventListener**  
StatementEventListener オブジェクトを PooledConnection オブジェクトから除去します。

## JDBC 4.0 例外クラスがサポートされる

以下の JDBC 4.0 例外クラスがサポートされています。

- **SQLNonTransientException** およびそのサブクラス:
  - **SQLDataException**
  - **SQLFeatureNotSupportedException**
  - **SQLIntegrityConstraintViolationException**
  - **SQLInvalidAuthorizationException**
  - **SQLNonTransientConnectionException**
  - **SQLSyntaxErrorException**
- **SQLTransientException** and its subclasses:

- `SQLException`
- `SQLTransactionRollbackException`
- `SQLTransientConnectionException`
- `SQLRecoverableException`
- `SQLClientInfoException`

## Wrapper インターフェース・サポートが追加された

Wrapper インターフェースにより、ラップされるリソースのインスタンスにアクセスすることができます。以下の IBM Data Server Driver for JDBC and SQLJ 専用クラスは、Wrapper インターフェースを実装します。

- `DB2Connection`
- `DB2BaseDataSource`
- `DB2SimpleDataSource`
- `DB2Statement`
- `DB2ResultSet`
- `DB2DatabaseMetaData`

## DB2Driver クラスが自動的にロードされる

以前は、`DriverManager` インターフェースを使用してデータ・ソースへの接続を確立した場合、`Class.forName` メソッドを呼び出して `DB2Driver` クラスを明示的にロードする必要がありました。JDBC 4.0 サポートを使用すれば、このステップは不要になります。

## リモート・トレース・コントローラーが追加された

リモート・トレース・コントローラーにより、複数のドライバー・インスタンスに対する以下のような操作を実行することができます。

- トレースの開始、停止、および再開
- 出力トレース・ファイルまたはディレクトリ位置の変更
- トレース・レベルの変更

リモート・トレース・コントローラーは、SDK for Java バージョン 6 以降のコンポーネントである、Java Management Extensions (JMX) アーキテクチャーを使用します。

---

## IBM Data Server Provider for .NET は IBM Informix Dynamic Server、IBM UniData、および IBM UniVerse をサポートする

IBM Data Server Provider for .NET は、DB2 データ・サーバーと共に、IBM Informix Dynamic Server、IBM UniData、および IBM UniVerse データ・サーバーをサポートするようになりました。そのため、同じ .NET アプリケーション中の DB2、Informix、UniData、または UniVerse 製品のいずれの組み合わせにも同じデータ・サーバー・プロバイダーを使用できます。

Data Server Provider for .NET を使用すると、.NET アプリケーションが以下のデータベース管理システムにアクセスできます。

- DB2 Database for Linux, UNIX, and Windows バージョン 9 (以降)

- DB2 Universal Database Version 8 for Windows, UNIX, and Linux ベースのコンピュータ
- DB2 Universal Database Version 7.3 (以降) for VSE & VM (DB2 Connect 経由)
- DB2 Universal Database Version 6 (以降) for OS/390 and z/OS (DB2 Connect 経由)
- DB2 Universal Database for iSeries (OS/400 V5R1 以降を使用、DB2 Connect 経由)
- IBM Informix Dynamic Server バージョン 11.10 以降
- IBM UniData 7.1.11 以降
- IBM UniVerse 10.2 以降

データ・タイプの互換性は、使用するデータ・サーバーに応じて変わります。例えば、Informix Dynamic Server、UniData、および UniVerse データ・サーバーは XML データ・タイプをサポートしていません。

#### 関連概念

データベース・アプリケーション 開発の基礎の『IBM Data Server Provider for .NET』

23 ページの『新しい DB2 クライアント製品ではデプロイメントが単純化されている (Windows)』

#### 関連タスク

ADO.NET および OLE DB アプリケーションの開発の『.NET アプリケーションのデプロイ (Windows)』

---

## 新しい DB2 サンプル・プログラムが追加された

独自のアプリケーション・プログラムを作成するためのテンプレートとして、また DB2 製品の機能を理解する目的で、DB2 サンプル・プログラムを使用できます。

サンプルは、DB2 Database for Linux, UNIX, and Windows のすべてのサーバー版、および IBM Data Server Client にバンドルされています。サンプルは以下の場所にあります。

- Windows オペレーティング・システム: `%DB2PATH%\sqlib\samples`  
(`%DB2PATH%` は、**DB2PATH** 環境変数で設定されている、DB2 製品がインストールされているディレクトリー)
- UNIX オペレーティング・システム: `$HOME/sqlib/samples` (`$HOME` は **\$HOME** 環境変数で設定されているインスタンス所有者のホーム・ディレクトリー)

以下のように、バージョン 9.5 では、主なフィーチャーと機能拡張を例示する多数の新しいサンプル・プログラムを使用できます。



表 6. 新しいバージョン 9.5 サンプル・プログラム

カテゴリー	フィーチャーまたは機能拡張	サンプルについての説明
パフォーマンス	データ再配分の高速化 (フィックスパック 1 で使用可能)	フィックスパック 1 で使用可能となる、 <b>REDISTRIBUTE</b> ユーティリティのさまざまなオプションを使用します
	自動ストレージ表スペースのストレージ削減	表スペースの最後の未使用ストレージを解放して再使用します
	行圧縮の機能拡張	ディクショナリー自動作成の機能拡張を使用します
	ロールアウト削除のための索引クリーンアップの据え置き	<b>MDC</b> ロールアウト・タイプを、索引の即時クリーンアップから索引の据え置きクリーンアップに変更します
	オプティミスティック・ロックのサポート	データベースでオプティミスティック・ロックを使用します
管理の容易性	データベース構成用の単一システム・ビュー	パーティション・データベース環境で、複数のパーティションのデータベース構成パラメータを更新します
	パーティション・データベース環境での <b>BACKUP DATABASE</b> の機能拡張	<b>BACKUP DATABASE</b> コマンドを実行して、複数パーティション・データベースの全パーティションのバックアップを 1 度に取ります
セキュリティ	監査機能の機能拡張	監査ポリシーの作成、変更、削除、および監査データのアーカイブと表示
	データベースロールのサポート	ロールの作成、ロールを使用したオブジェクト所有権の転送、グループの代替としてのロールの使用、ロール階層を使用したロールを介する特権の付与と取り消し
	トラステッド・コンテキストのサポート	トラステッド・コンテキストの作成、ユーザー ID を切り替えるためのトラステッド接続の識別と使用、トラステッド・コンテキスト固有の特権の取得
アプリケーション開発	グローバル変数のサポート	グローバル変数の作成と削除、トリガーやストアード・プロシージャでのグローバル変数の使用
	配列のサポート	配列データ・タイプの宣言、ストアード・プロシージャへの配列の引き渡し、配列を操作するさまざまな関数の使用、配列からの表の作成 (および表からの配列の作成)
	PHP サポート	PHP を、データベース・レベルの権限、DDL ステートメント、DML ステートメント、データ・タイプ、および XML などのさまざまな <b>DB2</b> フィーチャーと併用します。サンプルには <b>IBM_DB2</b> ドライバーと <b>PDO</b> ドライバーのサポートが組み込まれています。
	.NET サポート	<b>DB2 XML</b> フィーチャーを .NET と併用します

表 6. 新しいバージョン 9.5 サンプル・プログラム (続き)

カテゴリー	フィーチャーまたは機能拡張	サンプルについての説明
XML	トリガー処理の改善	トリガー処理機能を使用して、着信 XML 文書の自動検証を強制的に実行します
	XML 列のチェック制約	IS VALIDATED および IS NOT VALIDATED 述部を使用して、XML 列に対するチェック制約付きの表を作成します。また、ACCORDING TO XMLSCHEMA 節を使用して 1 つ以上のスキーマを指定します。
	使いやすい発行関数	リレーショナル・データと XML とのマッピングに XMLROW および XMLGROUP 関数を使用します
	XSLT サポート	XSLTRANSFORM 関数を使用して、データベース内の XML 文書を HTML、プレーン・テキスト、またはスタイル・シートを使用した他の XML 形式に変換します
	XQuery を使用した文書更新のサポート	XQuery トランスフォーム式を使用して、1 つ以上の XML 文書またはフラグメントを挿入、削除、更新、置換、名前変更します
	互換 XML スキーマの進化	元のスキーマと新規スキーマの互換性を維持しながら、登録済み XML スキーマを更新します
	アノテーション付き XML スキーマの分解機能の拡張	XML 文書の分解時の挿入順序を指定し、再帰的スキーマを登録して、それを分解で使用できるようにします
	SQLQuery へのパラメーター引き渡し	db2-fn:sqlquery 関数の中で指定した SQL 全選択に、XQuery からパラメーターを渡します
	XML 用のロード・ユーティリティのサポート	さまざまな LOAD コマンド・オプションを使用して、XML 文書を DB2 表にロードします

## CLIENT APPLNAME 特殊レジスターは CLP により自動的に設定される

バージョン 9.5 では、db2 **-tvf filename** コマンドを使用して CLP スクリプト・ファイルを実行すると、特殊レジスター CLIENT APPLNAME が CLP filename に設定されます。これは、どのバッチ・ジョブが現在実行中であるかをモニターし、CLP ワークロードを区別するのに有用です。

実行中のアプリケーションの値は、CLIENT\_APPLNAME 特殊レジスター内の値を取り出すことによって、または GET SNAPSHOT FOR ALL APPLICATIONS コマンドを使用することにより入手できます。

### 関連資料

管理 API リファレンスの『sqleseti API - クライアント情報の設定』

管理 API リファレンスの『sqlqryi API - クライアント情報の照会』

SQL リファレンス 第 1 巻の『CURRENT CLIENT\_APPLNAME 特殊レジスター』

## DB2 Developer Workbench が名前変更されて拡張された

DB2 Developer Workbench はバージョン 9.5 で名前変更されて、データ・サーバー開発者ツールと呼ばれるようになりました。データ・サーバー開発者ツールには、主要な新しい機能拡張が含まれています。

### データベース接続

- IBM Data Server Driver for JDBC and SQLJ は、製品に含まれています。このドライバーを使用して、任意の IBM データ・サーバーに接続できます。
- Database Explorer ツールバー・ボタンまたは「ファイル」→「エクスポート」メニューを使用して、データベース接続情報をデータベース記述子 (XML) ファイルにエクスポートできます。その後、これらのファイル内の情報を別のワークスペースにインポートできます。このフィーチャーにより、チームのメンバーはより容易にデータベース接続情報を共有できます。
- DB2 Database for Linux, UNIX, and Windows または DB2 for z/OS に、Kerberos 認証を使用して、または LDAP を使用して接続できます。Kerberos 接続では、サーバー上の Kerberos 環境を構成する必要があります。LDAP では、Windows Active Directory サーバーのスキーマを拡張して、DB2 オブジェクトをサポートするようにする必要があります。LDAP は、ホスト名およびポート番号の検索のためだけに使用されます。
- 「設定」ウィンドウの新規ページにデータベース接続の設定を行うことができます。接続設定には、接続タイムアウト、再接続オプション、およびユーザー ID とパスワードの永続性が含まれます。
- DB2 データベースに接続するとき、「新規接続 (New Connection)」ウィザードにデータベースのバージョン番号を指定する必要はなくなりました。
- JDBC 接続のためのトレース・ファイルを生成できます。このフィーチャーがサポートされるのは、IBM Data Server Driver for JDBC and SQLJ を使用する接続だけです。

### データ・アプリケーション開発

- SQL エディターでは、リレーショナルおよび XML データの両方に対して機能する照会を開発できるようになりました。エディターを使用して SQL および XQuery 式を作成できます。コンテンツ・アシスト、構文の強調表示、および照会の構文解析と妥当性検査などのエディター・フィーチャーは、両方の照会タイプに関して使用可能です。エディターの機能拡張に加えて、照会のテンプレートを作成および使用できるようにする、エディターの新しい設定ページがあります。これらの SQL エディターの機能拡張は、製品の一部ではなくなった XQuery ビルダーに置き換わるものとして設計されました。
- ルーチン・エディター (ストアード・プロシージャおよび UDF) がさらに便利になりました。Java ストアード・プロシージャでは、ルーチン・エディターから Java ソースへのリンクをクリックして、より簡単に Java ソース・コードを検索および編集できます。さらに、ネストされた

ストアード・プロシージャをデプロイするとき、ストアード・プロシージャをネストされた従属関係と共に表示して、デプロイメントのために選択できるようになりました。

- ウィザードまたは表データ・エディターを使用して、XML データを、データ開発プロジェクト内またはファイル・システム上に保管された XML ファイルから XML 列に挿入できます。DB2 Database for Linux, UNIX, and Windows サーバーでは、CLP スクリプトを作成してデータ開発プロジェクト内に保管された XML スキーマを登録できます。CLP スクリプトをワークベンチの外部で実行して、登録を実行できます。

### データベース管理

通常はデータベース管理者によって実行されるもので、Database Explorer から使用可能な新しいタスクがいくつかあります。

- データ・オブジェクトの管理。データ・オブジェクト・エディターを使用して、DB2 および Informix Dynamic Server データベース・オブジェクトの多くを作成および変更できます。
- 特権管理。データ・オブジェクト・エディターを使用して、データ・オブジェクトまたは許可 ID に関連した特権を付与、取り消し、および変更できます。
- データ値および関係の視覚化。データ・オブジェクトの関係を視覚化するには、Database Explorer で総括ダイアグラムを作成してから、それらのダイアグラムをイメージ・ファイルとして保存または印刷できます。さらに、データ値の分散を視覚化するために、Database Explorer からグラフィカルなデータ値分散ビューを作成できます。これらのフィチャーは、照会管理および分析 (Visual Explain) タスクに役立ちます。
- 統計サポート。アプリケーション・パフォーマンスの改善に役立つように、データ・オブジェクトの統計を表示および更新できます。また、製品内の DDL 生成フィチャーを使用して 1 つのデータベースから別のデータベースに統計をクローンまたはマイグレーションできます。

### Informix Dynamic Server

このリリースでは、Informix Dynamic Server (IDS) 用に新しいサポートがあります。Database Explorer 内の IDS データベースに接続した後、ほとんどのデータベース管理タスクを実行でき、接続情報を使用して IDS をターゲットにするデータ開発プロジェクトを作成できます。データ開発プロジェクト内で SQL ステートメントを開発および保管できます。ただし、DB2 ルーチン用に使用可能なウィザードおよびエディターは、IDS についてはまだ使用できません。IDS のルーチンを作成およびデプロイするには、CREATE 構文を入力してそれを SQL エディターで実行できます。さらに、ルーチンを Database Explorer から実行することもできます。

Informix JDBC ドライバーまたは IBM Data Server Driver for JDBC and SQLJ を使用して、Informix データベースに接続できます。

### Web サービス

ワークベンチを使用して、データベース操作 (SQL SELECT および DML ステートメント、XQuery 式、またはストアード・プロシージャの呼び出し) をクライアント・アプリケーションに公開する Web サービスを作成できるようになりました。データ開発プロジェクト内で Web サービスを作成して、Web サービスにドラッグ・アンド・ドロップすることにより、また

はウィザードを使用することにより、ストアード・プロシージャーおよび SQL ステートメントを簡単に追加できます。さらに、ワークベンチを使用して、Web サービスを Web サーバーにデプロイしたり Web サーバーからアンデプロイしたりすることができます。

さらに、ワークベンチを使用して既存の Web サービス Object Runtime Framework (WORF) アプリケーションをマイグレーションできます。

### Java 開発用の Pure Data Query

Pure Data Query (PDQ) は、単一のインターフェースである標準 SQL により、リレーショナル・データベースおよびメモリー内の Java コレクションにアクセスする共通照会 API を提供します。PDQ 機能により、Java プロジェクト内で以下のタスクを実行できます。その結果、リレーショナル・データと Java ソース・コードとの間の隔たりがなくなります。

- SQL コード・アシストを含む、Java エディターによる Java プログラム内の SQL の編集およびテスト
- Java クラス、SQL ステートメント、およびデータベース表を、互い同士任意の組み合わせで作成する
- メソッド起動による SQL の関連付け
- SQL への入力としての Java オブジェクトの引き渡し
- SQL からの出力としての Java オブジェクトの作成

### インストール

IBM Installation Manager を使用して、次のプラットフォーム上の DB2 サーバー製品から、データ・サーバー開発者ツールをインストール、更新、および管理するようになりました。すなわち、Linux on x86、32 ビットおよび 64 ビット Windows プラットフォームです。Installation Manager を製品のサイレント・インストールにも使用できます。さらに、ユーザーが開発者ツールを既存の Eclipse 環境にインストールできる、新しい「プラグインのみの」インストールも使用可能です。

### 関連概念

25 ページの『データ・サーバーの管理および開発用インターフェースの追加』

---

## 新しいスカラー関数でアプリケーションの移植が単純化された

バージョン 9.5 には、他のデータベース・ベンダーで使用されるスカラー関数と同じ名前の新しいスカラー関数が含まれています。既存のアプリケーションをバージョン 9.5 に移植するとき、コードを変更することなく、他のベンダーで使用される関数の名前を継続して使用できます。

以下のスカラー関数が提供されます。

- NVL (既存の COALESCE および VALUE 関数の同義語)
- LEAST または MIN (互いに同義語)
- GREATEST または MAX (互いに同義語)
- DECODE (既存の CASE 式に類似)

### 関連資料

SQL リファレンス 第 1 巻の『DECODE スカラー関数』

SQL リファレンス 第 1 巻の『GREATEST スカラー関数』  
SQL リファレンス 第 1 巻の『LEAST スカラー関数』  
SQL リファレンス 第 1 巻の『MAX スカラー関数』  
SQL リファレンス 第 1 巻の『MIN スカラー関数』  
SQL リファレンス 第 1 巻の『NVL スカラー関数』

---

## 追加された新しいビット単位のスカラー関数

新しい関数および演算子を使用して DB2 データのビット単位の操作を実行することにより、アプリケーション・コードを単純化できます。

以下の新しいビット操作スカラー関数が使用可能です。

- BITAND
- BITOR
- BITXOR
- BITNOT
- BITANDNOT

これらのビット単位の関数は、入力引数の整数値を「2 の補数」で表したものに対して実行されて、結果を、入力引数のデータ・タイプに基づくデータ・タイプの、対応する 10 進数の整数値で返します。サポートされる最大のタイプは、113 ビットをサポートします。

### 関連資料

SQL リファレンス 第 1 巻の『BITAND、BITANDNOT、BITOR、BITXOR、および BITNOT スカラー関数』

---

## 第 10 章 高可用性、バックアップ、ロギング、およびリカバリーの機能拡張

バージョン 9.5 には、ユーザーがデータを常に使用できるようにするための機能拡張が含まれます。

このセクションでは、バージョン 9.5 における高可用性、バックアップ、ロギング、およびリカバリーの機能拡張について詳しく説明します。

---

### 新規システム・ストアード・プロシージャにより自動保守ポリシー構成が単純化される

4 つの新規システム・ストアード・プロシージャを使用して、自動保守ポリシー情報を収集し、自動保守ポリシーを構成することができます。

新しい `SYSPROC.AUTOMAINT_SET_POLICY` および `SYSPROC.AUTOMAINT_SET_POLICYFILE` システム・ストアード・プロシージャを使用して、保守ウィンドウ、自動バックアップ、表および索引の自動再編成、および表 `RUNSTATS` の自動操作のためのポリシーを作成することができます。  
`AUTOMAINT_SET_POLICY` は XML LOB を入力として取り、  
`AUTOMAINT_SET_POLICYFILE` は XML ファイルを入力として取ります。要件に合わせて変更できるサンプル XML 入力ファイルが、`SQLLIB/samples/automaintcfg` ディレクトリーにあります。

さらに、`SYSPROC.AUTOMAINT_GET_POLICY` および `SYSPROC.AUTOMAINT_GET_POLICYFILE` という 2 つの新規システム・ストアード・プロシージャを使用して、保守ウィンドウ、自動バックアップ、表および索引の自動再編成、および表 `RUNSTATS` の自動操作に関する自動保守ポリシー情報を収集することができます。  
`AUTOMAINT_GET_POLICY` は、XML 形式の BLOB でポリシー情報を戻します。  
`AUTOMAINT_GET_POLICYFILE` は、XML ファイルでポリシー情報を戻します。

`AUTOMAINT_GET_POLICY` からの出力は `AUTOMAINT_SET_POLICY` への入力として渡し、  
`AUTOMAINT_GET_POLICYFILE` からの出力は `AUTOMAINT_SET_POLICYFILE` への入力として渡すことができます。

#### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの

『`SYSPROC.AUTOMAINT_SET_POLICY` または `SYSPROC.AUTOMAINT_SET_POLICYFILE` を使用した自動保守ポリシーの構成』

データ・リカバリーと高可用性 ガイドおよびリファレンスの

『`SYSPROC.AUTOMAINT_GET_POLICY` または `SYSPROC.AUTOMAINT_GET_POLICYFILE` を使用した自動保守ポリシー情報の収集』

#### 関連資料

管理ルーチンおよびビューの『AUTOMAINT\_GET\_POLICY プロシージャ - 自動保守ポリシーの検索』

管理ルーチンおよびビューの『AUTOMAINT\_GET\_POLICYFILE プロシージャ - 自動保守ポリシーの検索』

管理ルーチンおよびビューの『AUTOMAINT\_SET\_POLICYFILE プロシージャ - 自動保守ポリシーの構成』

管理ルーチンおよびビューの『AUTOMAINT\_SET\_POLICY プロシージャ - 自動保守ポリシーの構成』

---

## 新しい DB2 拡張コピー・サービス (ACS) API によりストレージ・ハードウェアとの統合が使用可能になっている

新しい DB2 拡張コピー・サービス (ACS) アプリケーション・プログラミング・インターフェース (API) を使用すると、ストレージ・ハードウェアを使用してスナップショット・バックアップ操作を実行できます。

従来のバックアップまたはリストア操作では、データベース・マネージャはオペレーティング・システム呼び出しを使用してディスクまたはストレージ・デバイスとの間でデータをコピーします。データ・コピーの実行にストレージ・デバイスを使用できるので、バックアップおよびリストア操作がずっと高速になります。DB2 ACS を使用したバックアップ操作のことを、スナップショット・バックアップと呼びます。

DB2 ACS アプリケーション・プログラミング・インターフェース (API) は、データベース・マネージャがスナップショット・バックアップ操作を実行するためにストレージ・ハードウェアとの通信に使用する関数の集合を定義します。

以下のストレージ・ハードウェア用の DB2 ACS API ドライバーが、IBM Data Server に統合されています。

- IBM TotalStorage SAN ボリューム・コントローラー
- IBM Enterprise Storage Server Model 800
- IBM System Storage DS6000
- IBM System Storage DS8000
- IBM N Series
- NetApp V-Series

その他のストレージ・ハードウェアを使用してスナップショット・バックアップ操作を実行するには、そのストレージ・ハードウェア用の DB2 ACS API ドライバーのみ必要になります。

### 関連概念

データ・リカバリーと高可用性 ガイドおよびリファレンスの『DB2 拡張コピー・サービス (ACS) の使用可能化』

### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの『DB2 拡張コピー・サービス (ACS) の使用可能化』



---

## リカバリー・オブジェクトの除去が自動化されてリカバリー・オブジェクト管理が単純化されている

バックアップ・イメージ、ロード・コピー・イメージ、およびリカバリーに不要な古いログ・ファイルを自動的に削除するよう、DB2 データベース・マネージャーを構成できます。

バックアップ・イメージ、ロード・コピー・イメージ、および古いログ・ファイルは大量のストレージ・スペースを消費します。ストレージ・スペースを節約するには、これらのリカバリー・オブジェクトを定期的に削除する必要があります。DB2 データベース・マネージャーは、**num\_db\_backups** 構成パラメーターで指定された数値を超えていて、**rec\_his\_retentn** 構成パラメーターで指定された日付より古い項目のデータベース履歴ファイルを自動的に整理します。新しい構成パラメーター **auto\_del\_rec\_obj** を ON に設定すると、データベース・マネージャーは、自動的に整理される履歴ファイル項目に関連したバックアップ・イメージ、ロード・コピー・イメージ、およびログ・ファイルも削除します。**auto\_del\_rec\_obj** を使用可能にすると、**num\_db\_backups** と **rec\_his\_retentn** の両方の値を超えた場合のみシステムはこの保守を実行します。

PRUNE HISTORY コマンドを使用して履歴ファイルを手動で整理することもできます。AND DELETE 節と PRUNE HISTORY コマンドを併用するか、db2Prune API 関数の **iOption** パラメーターを DB2PRUNE\_OPTION\_DELETE に設定すると、データベース・マネージャーは手動で整理される履歴ファイル項目に関連したログ・ファイルを削除します。**auto\_del\_rec\_obj** を ON に設定すると、データベース・マネージャーは、手動で整理される履歴ファイル項目に関連したバックアップ・イメージ、ロード・コピー・イメージ、およびログ・ファイルも削除します。

### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの『データベース・リカバリー・オブジェクト管理の自動化』

データ・リカバリーと高可用性 ガイドおよびリファレンスの『リカバリー・オブジェクトの削除に対する保護』

データ・リカバリーと高可用性 ガイドおよびリファレンスの『リカバリー・オブジェクトの管理』

### 関連資料

管理 API リファレンスの『db2Prune API - 履歴ファイル項目の削除、あるいはアクティブ・ログ・パスからのログ・ファイルの削除』

コマンド・リファレンスの『PRUNE HISTORY/LOGFILE コマンド』

管理ルーチンおよびビューの『PRUNE HISTORY/LOGFILE コマンド (ADMIN\_CMD プロシージャを使用)』

---

## 新しい DB2 高可用性インスタンス構成ユーティリティによりクラスタの構成と管理が単純化されている

新しい DB2 高可用性インスタンス構成ユーティリティ (db2haicu) を使用して、クラスタ化された環境中のデータベース・ソリューションの構成と管理を行えます。db2haicu を使用すると直接クラスタ・マネージャーと対話せずにデータベース関連クラスタの構成と管理を実行できるので、db2haicu によりクラスタの構成と管理が単純化されます。

db2haicu には対話式コマンド行インターフェースがあります。db2haicu は一連の質問を行い、直接システムを調査して、クラスタ、データベース・インスタンス、およびマシンに関する構成情報を収集します。構成情報が収集されると、次に db2haicu はクラスタ・ドメインというクラスタ化された環境の抽象モデルを作成します。

db2haicu がこの情報を収集してクラスタ・ドメインを作成したら、db2haicu を使用して以下のタスクなどのクラスタ管理タスクを実行できます。

- 新しいデータベースのクラスタ・ドメインへの追加
- DB2 高可用性災害時リカバリー (HADR) 1 次データベースとスタンバイ・データベースのペアの識別
- 新しいマシンのクラスタ・ドメインへの追加
- マシン上で保守を実行するための、クラスタ内のマシン外へのデータベース・インスタンスの移動
- フェイルオーバー・ポリシーの指定

db2haicu はクラスタ・マネージャーとの間のインターフェースの働きをします。例えば、データベースをクラスタ・ドメインに追加するには、クラスタ内のマシン上に作成した新しいデータベースをクラスタ・マネージャーに知らせます。

db2haicu がクラスタ・マネージャーと正常に連動するためには、クラスタ・マネージャーは DB2 クラスタ・マネージャー API ドライバーを提供して、DB2 クラスタ・マネージャー API をサポートしなければなりません。IBM Tivoli System Automation for Multiplatforms (SA MP) は DB2 クラスタ・マネージャー API をサポートし、Tivoli SA MP Base Component は DB2 High Availability Feature の一部として Linux および AIX 上の IBM Data Server インストールと統合されます。クラスタ・マネージャーとして Tivoli SA MP を使用している場合には、db2haicu を使用してクラスタ化された環境を構成できます。

### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの『高可用性のためのクラスタ環境の構成』

データ・リカバリーと高可用性 ガイドおよびリファレンスの『DB2 高可用性インスタンス構成ユーティリティ (db2haicu) を使用したデータベース・クラスタ環境の構成』

---

## 重複ログ制御ファイルによりデータベース・リカバリーの回復力が向上している

バージョン 9.1 では、データベース・マネージャーは SQLOGCTL.LFH という 1 つのログ制御ファイルを維持していました。バージョン 9.5 では、データベース・マネージャーは 2 つのログ制御ファイルのコピー (SQLOGCTL.LFH.1 と SQLOGCTL.LFH.2) を維持します。ログ制御ファイルのコピーが 2 つあるので、障害発生時にデータが失われるリスクが減っています。

障害後にデータベースが再始動するとき、データベース・マネージャーはログ・ファイルに保管されたトランザクション情報を適用して、データベースを整合性のある状態に戻します。ログ・ファイル内のどの項目を適用する必要があるか判別するために、データベース・マネージャーはログ制御ファイルを使用します。

ログ・ファイルの制御が壊れている場合、データベース・マネージャーはデータベースを整合性のある状態に戻すことができない可能性があります。しかし、ログ制御ファイルのコピーが 2 つあれば、ログ制御ファイルの 1 つのコピーが壊れた場合に、データベース・マネージャーは再始動中にもう一方のコピーを使用できるので、データベースの回復力を増強できます。

### 関連概念

データ・リカバリーと高可用性 ガイドおよびリファレンスの『データベース・ロギング』

データ・リカバリーと高可用性 ガイドおよびリファレンスの『ログ制御ファイル』

---

## HADR のピア・ウィンドウにより、連鎖または複数の障害発生時のデータ損失のリスクを削減する

新規の `hadr_peer_window` データベース構成パラメーターを使用して、DB2 高可用性災害時リカバリー (HADR) のプライマリがスタンバイ・データベースとの接続を失った場合に、プライマリとスタンバイ・データベースのペアが、ピア状態にあるかのように動作させることができます。

HADR の 1 次データベースとスタンバイ・データベースのペアがピア状態にある場合、スタンバイ・データベースのデータベース・ログが (同期モードに応じて) メモリーまたはローカル・ログ・パスに書き込まれたことの確認を、1 次データベースがスタンバイ・データベースから受け取るまで、トランザクションはコミットされたとは見なされません。これによりデータ整合性が保証されます。1 次データベースに障害が発生するとしても、1 次データベース上のデータベース・ログ内にあったすべてのトランザクション情報は、スタンバイ・データベース上のデータベース・ログにもあることになるからです。

1 次データベースとスタンバイ・データベースがピア状態の場合、1 次データベースがスタンバイ・データベースへの接続を失うと、1 次データベースがトランザクションについての確認をスタンバイ・データベースから受け取れないため、トランザクションをコミットできなくなります。以前のバージョンの IBM Data Server では、1 次データベースがスタンバイ・データベースとの接続を失うと、1 次データベースは単独でリモート・キャッチアップ・ペンディング状態に移行し、スタンバ

イ・データベースとは関係なくデータベース・アプリケーション要求の処理を続行できました。スタンバイ・データベースと関係ない状態の1次データベースで、トランザクションの処理中に障害が発生した場合、1次データベース上のトランザクション情報は失われる可能性があります。

バージョン 9.5 では、**hadr\_peer\_window** データベース構成パラメーターを非ゼロ値に設定すると、1次データベースは、スタンバイ・データベースとの接続を失った場合には、ピア状態から新規の切断ピア状態に移行します。1次データベースは、切断ピア状態の場合にもピア状態であるかのように動作し、トランザクションをコミットする前にスタンバイ・データベースからの確認を待機します。1次データベースが切断ピア状態である期間は、ピア・ウィンドウと呼ばれます。1次データベースの可用性はピア・ウィンドウの間は低減しますが、複数または連鎖の障害発生の場合に1次データベースでピア・ウィンドウ内に障害が発生したとしても、コミット済みのトランザクションは失われません。

#### 関連概念

データ・リカバリーと高可用性 ガイドおよびリファレンスの『DB2 高可用性災害時リカバリー (HADR) スタンバイ・データベースの状態』

#### 関連資料

管理 API リファレンスの『db2HADRTakeover API - データベースへの高可用性災害時リカバリー (HADR) 1次データベースとしてのテークオーバーの指示』

コマンド・リファレンスの『TAKEOVER HADR コマンド』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『hadr\_peer\_window - HADR ピア・ウィンドウ構成パラメーター』

---

## 単一システム・ビュー・バックアップで即時に複数のデータベース・パーティションをバックアップおよびリストアできる

新しい単一システム・ビュー (SSV) バックアップを使用して、複数のデータベース・パーティションを即時にバックアップおよびリストアできるようになりました。

バージョン 9.5 より前は、パーティション・データベースをデータベース・パーティションごとに1つずつバックアップする必要がありました。複数のデータベース・パーティションをデータベース・パーティションごとに1つずつバックアップすると誤りが発生しやすくなり、時間もかかります。パーティション・データベースをデータベース・パーティションごとに1つずつバックアップする場合、リストアおよびリカバリーに必要なログ・ファイルをバックアップ・イメージに含めることができません。

個別にバックアップされた複数のデータベース・パーティションをリストアする作業は、各データベース・パーティションのバックアップ・タイム・スタンプがわずかに異なるため、複雑です。各データベース・パーティションのバックアップ・タイム・スタンプが異なるため、同じバックアップに属するすべてのデータベース・パーティションの識別は難しく、それらすべてのデータベース・パーティションを含むバックアップの最小リカバリー時間を判別することは困難です。

db2\_all コマンドを使用することで、ある程度はパーティション・データベースのバックアップが簡素化されていましたが、作業が難しくなるような、バックアップとリストア操作での制限がまだありました。

バージョン 9.5 では、パーティション・データベースのカatalog・ノードからバックアップ操作を実行するとき、バックアップに含めるパーティションを指定したり、すべてのデータベース・パーティションを含めるように指定したりできます。指定されたパーティションは同時にバックアップされ、すべての指定されたデータベース・パーティションに関連付けられたバックアップ・タイム・スタンプは同じになります。さらに、SSV バックアップでデータベース・ログを含めることができます。スナップショットのバックアップ操作では、デフォルトの動作としてバックアップ・イメージにログが含まれます。最後の点として、SSV バックアップ・イメージからリストアするとき、*end of logs* へロールフォワードするように指定できます。これはデータベース・マネージャーにより計算される最小リカバリー時間で

#### 関連概念

データ・リカバリーと高可用性 ガイドおよびリファレンスの『バックアップの概要』

#### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの『バックアップの使用』

データ・リカバリーと高可用性 ガイドおよびリファレンスの『パーティション・データベースのバックアップ』

#### 関連資料

コマンド・リファレンスの『BACKUP DATABASE コマンド』

管理 API リファレンスの『db2Backup API - データベースまたは表スペースのバックアップ』

管理ルーチンおよびビューの『BACKUP DATABASE コマンド (ADMIN\_CMD プロシージャを使用)』

---

## 最小リカバリー時間へのロールフォワードが使用可能になった

**TO END OF BACKUP** 節を **ROLLFORWARD** コマンドと共に、または **DB2ROLLFORWARD\_END\_OF\_BACKUP** フラグを **db2Rollforward** API と共に使用して、パーティション・データベース内の全パーティションを最小リカバリー時間までロールフォワードできます。

最小リカバリー時間とは、ロールフォワード中に、データベースが整合状態になる(つまり、データベース・カatalogにリストされたオブジェクトが、ディスク上に物理的に存在するオブジェクトと一致する) 最も早い時点のことです。データベースのロールフォワード先として正しい時点を手作業で判別することは、特にパーティション・データベースの場合は困難です。バージョン 9.5 では、**TO END OF BACKUP** パラメーターを **ROLLFORWARD DATABASE** コマンドと共に使用して、または **DB2ROLLFORWARD\_END\_OF\_BACKUP** オプションを **db2Rollforward** API と共に使用して、データベース・マネージャーによって判別された最小リカバリー時間にデータベースをロールフォワードできます。

#### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの『ロールフォワードの使用』

#### 関連資料

管理 API リファレンスの『db2Rollforward API - データベースのロールフォワード』

コマンド・リファレンスの『ROLLFORWARD DATABASE コマンド』

---

## スナップショット・バックアップによって高速化されたデータのバックアップおよびリストア

スナップショット・バックアップまたはリストア操作を実行すると、ストレージ・デバイスはバックアップまたはリストアのデータ・コピーの部分を実行します。データ・コピーの実行にストレージ・デバイスを使用できるので、バックアップおよびリストア操作がずっと高速になります。

従来のバックアップまたはリストア操作では、データベース・マネージャーはオペレーティング・システム呼び出しを使用してディスクまたはストレージ・デバイスとの間でデータをコピーします。データ・コピーの実行にストレージ・デバイスを使用できるので、バックアップおよびリストア操作がずっと高速になります。DB2 ACS を使用したバックアップ操作のことを、スナップショット・バックアップと呼びます。

スナップショット・バックアップを実行するには、DB2 拡張コピー・サービス (ACS) が使用可能になっている必要があります。ストレージ・ハードウェア用の DB2 ACS API ドライバーがなければなりません。

以下のストレージ・ハードウェア用の DB2 ACS API ドライバーが、IBM Data Server に統合されています。

- IBM TotalStorage SAN ボリューム・コントローラー
- IBM Enterprise Storage Server Model 800
- IBM System Storage DS6000
- IBM System Storage DS8000
- IBM N Series
- NetApp V-Series

#### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの『スナップショット・バックアップ・イメージからのリストア』

データ・リカバリーと高可用性 ガイドおよびリファレンスの『スナップショット・バックアップの実行』

#### 関連資料

コマンド・リファレンスの『BACKUP DATABASE コマンド』

データ移動ユーティリティ ガイドおよびリファレンスの『RESTORE DATABASE コマンド』

コマンド・リファレンスの『db2acsutil - DB2 スナップショット・バックアップ・オブジェクトの管理コマンド』

---

## クラスター管理ソフトウェアの統合が使用可能

新しい DB2 クラスター・マネージャーのアプリケーション・プログラミング・インターフェース (API) によって、クラスター環境を構成するための DB2 高可用性インスタンス構成ユーティリティ (db2haicu) などの IBM Data Server クラスター構成ツールを使用できます。

DB2 クラスター・マネージャー API は、データベース・マネージャーがクラスター環境の構成のために、クラスター・マネージャーとインターフェースを取るために使用する関数のセットを定義します。データベース・マネージャーがクラスター・マネージャーと正常に連動するためには、クラスター・マネージャーは DB2 クラスター・マネージャー API ドライバーを提供して、DB2 クラスター・マネージャー API をサポートしなければなりません。

IBM Tivoli System Automation for Multiplatforms (SA MP) は、DB2 クラスター・マネージャー API をサポートします。さらに、Tivoli SA MP Base Component は Linux および AIX 上で DB2 High Availability Feature の一部として IBM Data Server インストールと統合されます。クラスター・マネージャーとして Tivoli SA MP を使用している場合には、db2haicu を使用してクラスター化された環境を構成できます。

### 関連概念

データ・リカバリーと高可用性 ガイドおよびリファレンスの『DB2 クラスター・マネージャー API』

### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの『高可用性インスタンス構成ユーティリティ (db2haicu) を使用したデータベース・クラスター環境の構成』

### 関連資料

データ・リカバリーと高可用性 ガイドおよびリファレンスの『サポートされているクラスター管理ソフトウェア』





---

## 第 11 章 インストール、マイグレーション、およびフィックスパック機能拡張

バージョン 9.5 には、製品のデプロイをより速くし、製品の保守をより簡単にする機能拡張があります。

バージョン 9.1 またはバージョン 8 のコピーがインストール済みでバージョン 9.5 にアップグレードする場合、バージョン 9.5 にマイグレーションする必要があります。DB2 バージョン 9.5 は新規リリースです。フィックスパックを適用してバージョン 9.1 からバージョン 9.5 にアップグレードすることはできません。

このセクションでは、バージョン 9.5 のインストール、マイグレーション、およびフィックスパック機能拡張について詳しく説明します。

マイグレーションの制約、潜在的な問題、および覚えておく必要がある他の詳細を調べるには、「マイグレーション・ガイド」にある『DB2 サーバーのマイグレーションに関する重要事項』および「マイグレーション・ガイド」にある『クライアントのマイグレーションに関する重要事項』を参照してください。

DB2 サーバーおよび DB2 クライアントをバージョン 9.5 にマイグレーションする場合、データベース・アプリケーションおよびルーチンをマイグレーションする必要が生じる場合があります。マイグレーションによる影響があるかどうかは、「マイグレーション・ガイド」にある『データベース・アプリケーションのマイグレーションに関する重要事項』および「マイグレーション・ガイド」にある『ルーチンのマイグレーションに関する重要事項』を検討してください。

---

### IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component が DB2 インストール済み環境に統合される (Linux および AIX)

IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component バージョン 2.2 は、IBM Data Server on Linux and AIX にバンドルされました。

SA MP Base Component は、やはり IBM Data Server 製品とバンドルされている DB2 インストーラーまたは Tivoli スクリプトを使用して、インストール、更新、およびアンインストールできるようになりました。

IBM Tivoli System Automation for Multiplatforms は、システム・モニター、フェイルオーバー、および自動修復および保守などのクラスター管理機能を提供します。詳しくは、Tivoli software information center を参照してください。

#### 関連概念

データ・リカバリーと高可用性 ガイドおよびリファレンスの『IBM Tivoli System Automation for Multiplatforms (Linux および AIX)』

データ・リカバリーと高可用性 ガイドおよびリファレンスの『DB2 インストーラーを使用した SA MP Base Component のインストールおよびアップグレード』

---

## 追加ツールのインストールを DB2 インストールから立ち上げられる

DB2 サーバー製品のインストール中に、データ・サーバー開発者ツールかデータ・サーバー管理者ツールのインストールを選択できます。

DB2 サーバーのインストール中にデータ・サーバー開発者ツールかデータ・サーバー管理者ツールのインストールを選択すると、DB2 セットアップ・ウィザードが、DB2 サーバー製品のインストール後にスタンドアロン・インストール・プログラムを立ち上げるようになります。また、DB2 サーバー製品をインストールせずに、DB2 セットアップ・ランチパッドからデータ・サーバー開発者ツールかデータ・サーバー管理者ツールをインストールできるようになりました。

データ・サーバー開発者ツールとデータ・サーバー管理者ツールには独自のスタンドアロン・インストール・プログラムがあり、DB2 インストールとは独立して立ち上げることができます。DB2 インストール・プログラムは、DB2 応答ファイルを使用したこれらのツールのインストールをサポートしていません。

サイレント・モードでデータ・サーバー開発者ツールかデータ・サーバー管理者ツールをインストールするには、スタンドアロン・インストール・プログラムで提供される非対話式のインストール方法を使用します。

---

## 共通フィックスパックによりサーバー製品の更新が単純化されている

同じインストール場所のすべてまたはいずれかの DB2 サーバー製品を更新する場合に、製品固有の DB2 サーバー・フィックスパックを個別に適用する必要がなくなりました。1 つの DB2 サーバー・フィックスパック・イメージを使用して、これらの製品を更新できます。

サーバー・フィックスパック・イメージを使用して、すべてのプラットフォーム上の IBM Data Server Client や、Linux およびUNIX プラットフォーム上の IBM Data Server Runtime Client を更新することもできます。

### 関連タスク

問題判別ガイドの『フィックスパックの適用』

---

## 自動化されたフィックスパックのインストール後タスク (Linux および UNIX)

以前はフィックスパックのインストール中に手動で行っていた 2 つのステップ、つまり db2iupdt および dasupdt コマンドの実行が自動化され、フィックスパックのインストールが簡素化されました。さらに、最初の接続時にバインディングが自動的に行われます。その結果、データベース・マネージャーを一度開始すると、DB2 製品はインストール直後に使用できるようになります。

以前のリリースでは、インスタンスを更新し、手動でバインドする必要がありました。

### 関連タスク

問題判別ガイドの『フィックスパックの適用』

---

## 非ルート・ユーザーが DB2 製品のインストールと構成を実行できる (Linux および UNIX)

バージョン 9.5 より前では、ルート特権がある場合に限り、製品のインストール、フィックスパックの適用とロールバック、インスタンスの構成、フィーチャーの追加、製品のアンインストールを行えました。現バージョンでは、非ルート・ユーザーの場合、Linux および UNIX プラットフォーム上でこれらのタスクを実行できます。

DB2 インストーラーは、非ルート・インストール中に非ルート・インスタンスを自動的に作成して構成します。非ルート・ユーザーとして、インストール中に非ルート・インスタンスの構成をカスタマイズできます。またルート特権がなくても、インストール済みの DB2 製品を使用したり保守したりできます。

DB2 製品の非ルート・インストールには 1 つの DB2 インスタンスがあり、ほとんどのフィーチャーがデフォルトで使用可能になっています。

非ルート・インストールは、以下のような多数のグループにとって魅力的になる可能性があります。

- 数千のワークステーションとユーザーを抱え、システム管理者が時間を浪費することなく DB2 製品をインストールしたい企業。
- DB2 製品を使用してアプリケーションを開発したい、通常はシステム管理者ではないアプリケーション開発者。
- ルート権限を必要としない、DB2 製品を組み込んだソフトウェアを開発している独立系ソフトウェア・ベンダー (ISV)

非ルート・インストールには、ルート・インストールの大半の機能がありますが、多少の相違点と制限があります。ルート・ユーザーが `db2rfe` コマンドを実行すると、制限の一部を除くことができます。

### 関連概念

IBM データ・サーバー・クライアント機能 概説およびインストールの『ルート・インストールと非ルート・インストールの間の相違点』

IBM データ・サーバー・クライアント機能 概説およびインストールの『非ルート・インストールの制限』

### 関連タスク

IBM データ・サーバー・クライアント機能 概説およびインストールの『`db2rfe` による非ルート・インストール内のルート・ベースのフィーチャーの使用可能化』

---

## 応答ファイルの新規キーワードの追加

応答ファイルは、セットアップおよび構成の情報が入った ASCII テキスト・ファイルです。DB2 応答ファイルを使用したインストールでは、DB2 セットアップ・ウィザードを使ったインストールとは異なり、ユーザーとの対話なしで DB2 製品またはフィーチャーをインストールできます。

応答ファイルの新規キーワードを使用すると、DB2 製品のインストールとデプロイが容易になります。

以下のキーワードを AIX および Linux プラットフォームで使用すると、IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component をセットアップすることができます。

- INSTALL\_TSAMP

以下のキーワードを使用すると、Windows プラットフォームで拡張セキュリティーを有効にすることができます。

- DB2\_ADMINGROUP\_DOMAIN
- DB2\_USERSGROUP\_DOMAIN

デフォルトの項目が設定された、すぐに使えるサンプル応答ファイルがこの製品に含まれています。サンプル応答ファイルは、DB2 CD の db2/platform/samples (ここで platform はハードウェアのプラットフォームを指す) にあります。

#### 関連資料

DB2 サーバー機能 概説およびインストールの『応答ファイル・キーワード』

---

## 非 DB2 インスタンス・マージ・モジュールが追加された

バージョン 9.5 より前では、Windows インストーラーを使う任意の製品に、DB2 Runtime Client マージ・モジュールを使用して、DB2 Runtime Client 機能を追加できました。バージョン 9.5 では、DB2 Runtime Client マージ・モジュールは DB2 インスタンス・マージ・モジュールと呼ばれ、非 DB2 インスタンス・マージ・モジュールが追加されています。

新しい非 DB2 インスタンス・マージ・モジュールを使用すると、Windows インストーラーを使用する製品に IBM Data Server Driver for ODBC, CLI, and .NET 機能を簡単に追加できます。IBM Data Server Driver for ODBC, CLI, and .NET は新しいインストール可能コンポーネントで、Windows ベースのアプリケーションから DB2 サーバーへのアクセスの提供が非常に容易になります。IBM Data Server Driver for ODBC, CLI, and .NET は、独立系ソフトウェア・ベンダー (ISV) によって再配分され、大企業にありがちな一括デプロイメントのシナリオでアプリケーション配分を使用されるように設計されています。

IBM Data Server Driver for ODBC, CLI, and .NET を特徴付ける主なフィーチャーは以下のとおりです。

- 1 つの実行可能ファイルとして出荷されるので、再配分やデプロイを行いやすくなっています。
- Windows Installer マージ・モジュール (.msm ファイル) を使用できます。このモジュールは、IBM Data Server Driver for ODBC, CLI, and .NET コードのもっと大きなアプリケーション中での統合を単純化します。

#### 関連概念

IBM データ・サーバー・クライアント機能 概説およびインストールの『IBM データ・サーバー・クライアントのタイプ』

107 ページの『第 11 章 インストール、マイグレーション、およびフィックスパック機能拡張』

#### 関連資料

---

## スタンドアロン・インスタンス・タイプによりサポートされているプラットフォーム間の整合性が向上している (Linux および UNIX)

Linux および UNIX オペレーティング・システムにスタンドアロンのインスタンス・タイプが追加され、サポートされている DB2 プラットフォーム間のインスタンス・タイプの整合性が向上しました。

スタンドアロン・インスタンスとは、ローカル接続のみ実行できるデータベース・サーバーとローカル・クライアントのインスタンスのことです。スタンドアロン・タイプを作成するには、db2icrt コマンドの `-s` パラメーターに `standalone` を指定します。

### 関連資料

コマンド・リファレンスの『db2icrt - インスタンスの作成コマンド』

---

## Windows Vista のデプロイメントと使用がさらに簡単になった

Windows Vista オペレーティング・システムのサポートがバージョン 9.1 フィックスバック 2 で導入されました。これにより DB2 および DB2 Connect 製品の Windows Vista オペレーティング・システム上でのデプロイおよび使用はさらに簡単になりました。

バージョン 9.5 には、以下の機能拡張が含まれています。

- IBM データ・サーバー開発者ツールおよび Query Patroller 用のサポートが追加されています。
- 完全な管理者権限で DB2 コマンド・ウィンドウを起動するための DB2 ショートカットが追加されました。ローカル管理者のグループのメンバーであれば、オペレーティング・システムのローカル管理者権限を必要とする DB2 コマンドおよびツールを起動するには、このショートカットを使用します。

### 関連概念

153 ページの『拡張セキュリティーでは、ユーザーが DB2ADMNS または DB2USERS グループに属する必要がある (Windows Vista)』

### 関連資料

データベース・アプリケーション 開発の基礎の『DB2 サーバーおよび IBM データ・サーバー・クライアントのインストール要件 (Windows)』

DB2 Connect Personal Edition 概説およびインストールの『DB2 Connect Personal Edition のインストール要件 (Windows)』

DB2 Connect サーバー機能 概説およびインストールの『DB2 Connect サーバー製品のインストール要件 (Windows)』



---

## 第 12 章 フェデレーションの機能強化

アプリケーション開発、構成、およびセキュリティーのすべてにおいて、IBM WebSphere Federation Server バージョン 9.5 では向上が図られています。

IBM WebSphere Federation Server バージョン 9.5 では、以下のフェデレーションの機能拡張が使用可能です。

---

### アプリケーション開発がフェデレーション用に拡張された

バージョン 9.5 で使用できるアプリケーション開発の機能拡張には、XML データ・タイプのサポート、アプリケーション・セーブポイント、および WITH HOLD カーソルがあります。

#### XML データ・タイプ・サポート

リモート XML データ・タイプのサポートにより、リモートの DB2 Database for Linux, UNIX, and Windows XML データを、フェデレーテッド・サーバーからアクセスおよび操作することができます。XML サポートには、以下の主要フィーチャーが組み込まれています。

- フェデレーション・サーバー上の XML タイプとリモート XML タイプとの間のタイプ・マッピング
- 柔軟な XML データ操作を提供する SQL/XML および XQuery 言語の使用

#### アプリケーション・セーブポイント

フェデレーテッド・アプリケーション・セーブポイントにより、トランザクションに対する制御は向上し、ロック競合は削減され、アプリケーション・ロジックとの統合が向上します。トランザクションは、単一レベルまたはネスト・レベルのセーブポイント単位に論理的に分割することができます。それぞれのセーブポイントは、個別に解放することも、アプリケーション・ロジックごとにロールバックすることもできます。

WebSphere Federation Server は、DB2 Database for Linux, UNIX, and Windows データ・ソースに対する挿入、更新、および削除操作のためのアプリケーション・セーブポイントをサポートするようになりました。

#### WITH HOLD カーソル

WebSphere Federation Server では、WITH HOLD 属性を使用して宣言したカーソルを、DRDA<sup>®</sup> ラッパーおよび DB2 Database for Linux, UNIX, and Windows データ・ソースの複数の作業単位にまたがってオープンしたままにしておけるようになりました。この機能拡張により、カーソルを使用した処理の結果セットに対する制御を向上させることができます。

以前はコミットまたはロールバック操作により作業単位が完了すると、カーソルはクローズされました。現在では、フェデレーテッド・サーバーのカーソルはデー

タ・ソースのカーソルと同期し、フェデレーテッド・カーソルは適正な時点でクローズします。

---

## セキュリティがフェデレーション用に拡張された

フェデレーション用の新規セキュリティ機能拡張には、トラステッド・コンテキスト、データベースのロール、および外部ユーザー・マッピング・リポジトリへの C ベースの新規インターフェースがあります。

### トラステッド・コンテキスト

トラステッド・コンテキストは、フェデレーテッド・サーバーと (アプリケーション・サーバーやデータ・ソース・サーバーなどの) 外部エンティティとの間のトラステッド関係を定義する、データベース・セキュリティ・オブジェクトです。トラステッド・コンテキストの定義と一致する明示接続要求が行われた場合、フェデレーテッド・サーバーはアプリケーション・サーバーからフェデレーテッド・サーバーへのインバウンド・トラステッド接続を確立し、フェデレーテッド・サーバーからデータ・ソース・サーバーへのアウトバウンド・トラステッド接続を確立しようとします。

フェデレーテッド・システムでは、トラステッド・コンテキストには以下の利点があります。

- エンドツーエンドのユーザー ID。ユーザーの ID は、フェデレーテッド・システム全体で伝搬されます。
- アカウンタビリティ。監査ログは、サーバーがそれぞれの目的に従って実行するトランザクションと、個々のユーザーが実行するトランザクションを識別します。したがって、個々のユーザーは特定のトランザクションについての責任を持つことができます。
- 特権制御。デフォルトのロールをトラステッド・コンテキストのすべてのユーザーに付与することができます。加えて、付加的な特定のロールを特定のユーザーに付与して、ユーザーがその特定のコンテキスト内で作業する間だけ付加的な特権を持てるようにすることができます。
- セキュリティ。システム許可 ID は、すべてのユーザーが必要とするすべての特権のスーパーセットではなく、システムが必要とする特権だけを入手します。システム許可 ID に問題が生じた場合、許可 ID がすべての特権を持っている場合よりもデータのぜい弱性は低くなります。
- 管理保守。ユーザー・マッピングおよびグループの数は大幅に削減されました。
- パフォーマンス。トラステッド接続は、確立されると接続の存続期間中は有効のままとなります。サーバーは、別のユーザーが接続するたびに、物理接続をクローズして新規接続を確立することはしません。代わりに、サーバーは接続の現行ユーザー ID を別のユーザー ID に切り替えます。トラステッド・コンテキストの定義方法に応じて、認証が不要になる場合があります。

フェデレーテッド・トラステッド・コンテキストは、DRDA ラッパー、および DB2 Universal Database for Linux, UNIX and Windows および DB2 Universal Database for z/OS の各データ・ソースでの使用がサポートされています。



## 外部ユーザー・マッピング・リポジトリへの C ベースのインターフェース

デフォルトでは、管理者としてユーザー・マッピングを作成するときに、それらは暗号化されて、各フェデレーテッド・サーバー上のグローバル・カタログに格納されます。一般に、ユーザーは使用する各フェデレーテッド・サーバー上に 1 つ以上のマッピングが必要です。データ・ソースのリモート・パスワードが変更されるたびに、1 つ以上のグローバル・カタログ内でユーザー・マッピングを更新する必要があります。

ユーザー・マッピング情報のセキュリティーを改善し、その保守を削減するには、ユーザー・マッピング情報を外部リポジトリに格納します。外部リポジトリには例えば LDAP サーバーがあり、これは SSL や強い暗号化などの追加のセキュリティー・フィーチャーを使用して情報を保護することができます。フェデレーテッド・サーバーを外部リポジトリを使用するように構成した後に、リモート・パスワードを変更した場合は、ユーザー・マッピングを一度だけ更新する必要があります。

フェデレーテッド・サーバーはプラグインを使用して、外部リポジトリへのインターフェースを提供します。以前には、WebSphere Federation Server では、プラグインの作成には Java の使用しかサポートされていませんでした。現在では、C および C++ がサポートされています。インターフェース・ライブラリー、ヘッダー・ファイル、およびサンプル C プラグインが提供されています。

---

## 構成がフェデレーション用に拡張された

バージョン 9.5 で使用可能なフェデレーション用の構成の機能拡張には、ニックネーム統計の自動収集や、ニックネーム列と索引名の生成のための改善されたメソッドがあります。

### ニックネーム統計の自動リフレッシュ

自動統計収集が拡張され、統計を自動的にリフレッシュするニックネーム統計 (NNSTAT) ストアード・プロシージャの実行によって、ニックネーム統計を最新に保つことができます。最新の統計により、フェデレーテッド・サーバー上のオプティマイザーは、照会プランのパフォーマンスを向上させる、情報を得た上での選択を行うことができます。

詳しくは、<http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/topic/com.ibm.swg.im.iis.fed.query.doc/topics/iiyfqnnsatm.html> の『ニックネーム統計の自動リフレッシュ』を参照してください。

### 改善されたニックネーム列および索引名の生成

リレーショナル・ニックネームのニックネーム列および索引名の生成に使用するメソッドは拡張され、生成名は元の名前とより一層一致するようになりました。

詳しくは、<http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/topic/com.ibm.swg.im.iis.fed.query.doc/topics/iiyfqnnonam.html> の『ニックネーム列および索引名』を参照してください。



---

## 第 13 章 レプリケーションの機能拡張

バージョン 9.5 のレプリケーションの機能拡張には、新しい CCD ターゲット・タイプと、DECFLOAT 10 進浮動小数点データ・タイプのサポートが含まれます。

このセクションでは、バージョン 9.5 のレプリケーションの機能拡張について詳しく説明します。

---

### 新規 CCD ターゲット・タイプにより UOW 表と CD 表の結合を回避する

整合変更データ (CCD) ターゲット表を、変更データ (CD) 表と IBMSNAP\_UOW 表を結合せずに複製するオプションが備えられました。

新規の表タイプを使用するには、IBMSNAP\_SUBS\_MEMBR 表の TARGET\_STRUCTURE 列に値 9 を指定します。

タイプ 9 の CCD 表に IBMSNAP\_LOGMARKER 列が含まれているとしても、アプライ・プログラムは、この列のソース・コミット情報を入手するために CD 表と IBMSNAP\_UOW 表との結合を必要としません。代わりに、アプライ・プログラムは同じサイクル内のすべての行に対して、IBMSNAP\_LOGMARKER 列内に同じ値を生成します。

新規の CCD 表タイプは、タイプ 3 の CCD 表と同じ構造を持ちます。これにはユーザー列に加えて、以下にある 4 つの必須の IBM 列が含まれています。

```
IBMSNAP_COMMITSEQ  
IBMSNAP_INTENTSEQ  
IBMSNAP_OPERATION  
IBMSNAP_LOGMARKER
```

三層のレプリケーション構成のソース表として、新規ターゲット表タイプを登録できます。

---

### DECFLOAT データ・タイプがレプリケーション用にサポートされる

新規の DECFLOAT 10 進浮動小数点データ・タイプは、16 または 34 桁の精度を持つ、厳密な 10 進値を要求するビジネス・アプリケーションをサポートします。DECFLOAT データ・タイプで定義したデータは、DB2 Database for Linux, UNIX, and Windows および DB2 for z/OS のどちらでも複製することができます。

レプリケーション・キー列の一部として DECFLOAT で定義した列を使用することができます。

SQL レプリケーションは、データ切り捨てを理由として、DECFLOAT (34) のソース列の、DECFLOAT (16) ターゲット列へのマッピングをサポートしていません。



---

## 第 14 章 各国語の機能拡張

各国語の機能拡張により、データの配列のためのより多くのオプション、新しい Unicode スtring・リテラル、および新しいスカラー関数のための文字ベースの処理が提供されます。

以下のセクションでは、バージョン 9.5 で導入された各国語の機能拡張について説明します。

---

### 言語認識照合によりデータの並び順にさらに多くのオプションが提供される

Unicode データベースの作成時に、言語認識照合を指定できるようになりました。言語認識照合により、ご使用の言語および地域の慣習により適したデータの並び順を選択することができます。

言語認識照合は、非 Unicode データベースの SYSTEM 照合をベースとしています。それぞれの言語認識照合では Unicode データを、非 Unicode コード・ページに変換して、SYSTEM 照合を適用するかのように並べ変えます。

非 Unicode データベースを Unicode に変換する場合、言語認識照合によりデータの配列に相違がないようにすることができます。

さらに非 Unicode SYSTEM 照合は、新規の COLLATION\_KEY\_BIT スカラー関数と共に使用することができます。例えば、SQL 照会の結果をソートするためにどの照合を使用するかを指定することができます。

#### 関連概念

国際化対応ガイドの『Unicode データの言語対応型の照合』

---

### Unicode スtring・リテラルにより任意の Unicode 文字を指定できる

バージョン 9.5 は、キーボードから直接入力できないサポート対象文字に対応する、新規 Unicode スtring・リテラルを導入しています。

このリテラルにより、キーボードから簡単にまたはまったく入力できないコード・ポイントを指定することが可能になります。例えば、日本語ひらがな文字を英語キーボードで入力することはできませんが、Unicode の同等文字コードを指定することはできます。

このリテラルには、非 Unicode スtring・リテラルと同じセマンティクスがあります。ただし、これらの文字は直接入力することに加え、Unicode コード・ポイントにより指定することができます。

#### 関連資料

SQL リファレンス 第 1 巻の『定数』

---

## スカラー関数の文字ベースの処理は可変文字サイズをサポートする

バージョン 9.5 には、ストリング・データを一連の文字として処理できる新規スカラー関数が含まれています。それぞれの文字のバイト表現を知る必要はありません。

Unicode での各文字は、1 から 4 バイトで構成されます。可変文字サイズに対応するために、以下の関数はデフォルトで文字ベースの処理をするようになりました。

- INSERT
- LEFT
- RIGHT
- STRIP
- TRIM

バイト・ベースの処理を使用する場合は、SYSFUN スキーマに格納されたこれらの関数の各バージョンを使用します。例えば、INSERT 関数のバイト・ベースの処理を使用するには、SYSFUN.INSERT を要求します。

バージョン 9.5 の新機能である OVERLAY スカラー関数も、文字ベースの処理をサポートします。

### 関連資料

SQL リファレンス 第 1 巻の『INSERT スカラー関数』

SQL リファレンス 第 1 巻の『LEFT スカラー関数』

SQL リファレンス 第 1 巻の『RIGHT スカラー関数』

SQL リファレンス 第 1 巻の『STRIP スカラー関数』

SQL リファレンス 第 1 巻の『TRIM スカラー関数』

SQL リファレンス 第 1 巻の『OVERLAY スカラー関数』

---

## Big5-HKSCS–Unicode 変換表による、Unicode データベース内の HKSCS データ保管のサポートの拡大

バージョン 9.5 は、Big5-HKSCS クライアントが Unicode データベースに接続して HKSCS (Hong Kong Supplementary Character Set) でデータを保管することを可能にする、新しい Unicode 変換表を提供します。

新しい変換表は、HKSCS-2004 および Unicode 4.1 文字をサポートします。

Unicode 4.1 の時点ですべての HKSCS-2004 文字は非 PUA 割り当てとなっているので、Big5-HKSCS から Unicode への変換によって、Private Use Area (PUA) コード・ポイントが生成されることはありません。ただし、Unicode から Big5-HKSCS への変換中に、HKSCS 文字用に以前の Unicode バージョンで使用された PUA コード・ポイントは、適切な Big5-HKSCS コード・ポイントに変換されます。

### 関連概念

国際化対応ガイドの『コード・ページ 950 データベースに接続する Windows クライアント』

### 関連タスク

国際化対応ガイドの『HKSCS データが含まれるコード・ページ 950 データベースの Unicode データベースへのマイグレーション』

#### 関連資料

国際化対応ガイドの『サポートされているテリトリー・コードおよびコード・ページ』

---

## UPPER (UCASE) および LOWER (LCASE) スカラー関数はロケールをサポートする

スカラー関数 UPPER (UCASE) および LOWER (LCASE) は、ロケール依存変換を使用してテキストの大/小文字を変更できるようになりました。デフォルトでは、UPPER および LOWER はロケールに関係なくストリング内の文字を変換します。一部の文字では、ロケール・ベースの変換を使用するとき、大文字と小文字とのマッピングが異なります。

例えば、トルコ語では、文字 *i* に 4 つの異なるバージョンがあります。ドットの付いた *i* およびドットの付いた大文字の *I* は、大文字の場合も小文字の場合もあります。ドットの付いた小文字の *ı* およびドットの付いた大文字の *İ* は、それぞれのドットなしのバージョンとは異なります。トルコ語のロケール `tc_TR` を指定すると、UPPER スカラー関数はローマ字の小文字 *i* を上にドットの付いた大文字 *I* に変換します。この Unicode 文字コードは `U&'\0130'` です。トルコ語のロケールを指定すると、LOWER スカラー関数は、ローマ字の大文字 *I* をドットの付いた小文字 *i* に変換します。この Unicode 文字コードは `&'\0131'` です。ロケールを指定しない場合、UPPER スカラー関数はローマ字の小文字 *i* をローマ字の大文字 *I* (Unicode 文字 `U&'\0049'`) に変換し、LOWER スカラー関数はローマ字の大文字 *I* をローマ字の小文字 *i* (Unicode 文字 `U&'\0069'`) に変換します。

#### 関連資料

SQL リファレンス 第 1 巻の『UPPER スカラー関数』

SQL リファレンス 第 1 巻の『LOWER スカラー関数』

SQL リファレンス 第 1 巻の『LOWER (ロケール依存) スカラー関数』

SQL リファレンス 第 1 巻の『UPPER (ロケール依存) スカラー関数』

SQL リファレンス 第 1 巻の『LCASE スカラー関数』

SQL リファレンス 第 1 巻の『UCASE スカラー関数』

SQL リファレンス 第 1 巻の『LCASE (ロケール依存) スカラー関数』

SQL リファレンス 第 1 巻の『UCASE (ロケール依存) スカラー関数』





---

## 第 15 章 トラブルシューティングと問題判別の機能拡張

このセクションは、トラブルシューティングおよび問題判別の機能拡張について要約しています。これには、データベースの回復性とデータ整合性の改善、エラーおよびログ表示の簡素化、およびエラーの追跡とロギングのための新しいツールが含まれます。

このセクションでは、トラブルシューティングおよび問題判別の機能拡張について詳しく説明します。

---

### データ収集ツールで予期しないエラーを追跡する

新しいツールの db2fodc (First Occurrence Data Capture) は、DB2 インスタンス中で検出された予期しないエラー条件がある場合に、症状に基づくデータを収集します。このデータを使用して、データベースで発生した可能性のあるエラー条件のトラブルシューティングに役立てることができます。

データベース・マネージャーに db2fodc ツールを自動的に実行させるか、またはこのツールを手動で実行することができます。データベース・マネージャーがこのツールを実行する原因となる症状を指定しなければなりません。db2fodc ツールを開始する症状には、シグナル (UNIX オペレーティング・システム上)、例外 (Windows オペレーティング・システム上)、および障害、マシン停止、またはパニックを生じさせるデータ破壊などによる、予期しない DB2 システム・エラーが含まれます。db2pdcfg 構成パラメーターか、DB2FODC レジストリー変数とその関連パラメーター中で値を設定して、収集するデータのタイプを指定します。データ収集の完了後、db2support ツールを実行して、結果の診断ファイルを準備して収集し、IBM サポートに送信するパッケージを準備しなければなりません。

このツールは、一部のデータ収集操作に取って代わっており、db2support などの他のトラブルシューティング・ツールや IBM サポートで使用されるツール中にある類似した他のデータ収集操作を取り込んでいます。

#### 関連資料

コマンド・リファレンスの『db2support - 問題分析および環境収集ツール・コマンド』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『汎用レジストリー変数』

コマンド・リファレンスの『db2pdcfg - 問題判別動作の DB2 データベースの構成コマンド』

コマンド・リファレンスの『db2fodc - DB2 First Occurrence Data Collection コマンド』

---

## ロギング機能に SQL 管理ルーチンが追加された

新しい PD\_GET\_DIAG\_HIST システム定義ルーチンを使用して、オプティマイザー統計や管理通知ログなどのさまざまな機能からイベント、通知、および診断ログ・レコードを戻すことができます。

このルーチンは、顧客への影響の重要度、レコード・タイプ、およびレコードのタイム・スタンプに従う事前フィルター操作もサポートします。顧客への影響の重大度は、通知 (Informational) から重大 (Critical) までの範囲にすることができます。診断情報の有効範囲とフォーカスを制御できます。

### 関連資料

管理ルーチンおよびビューの『PD\_GET\_DIAG\_HIST 表関数 - 特定の機能からレコードを戻す』

---

## ストレージ・キーによりメモリー・アクセス問題が検出される

バージョン 9.5 はストレージ・キーをサポートしています。ストレージ・キーは IBM POWER6 プロセッサおよび AIX オペレーティング・システムの新規フィーチャーで、ハードウェア・キーをカーネル・スレッド・レベルで使用して一定の範囲のメモリーを保護します。

ストレージ・キーを使用して、バッファー・プール・メモリーを保護できます。データベースに対する更新は、バッファー・プールの使用中に行われます。ストレージ・キー保護により、バッファー・プール・メモリーの破損の問題が減少し、データベースを停止させる可能性のあるエラーが制限されます。プログラミングという手段によってバッファー・プールに不正にアクセスしようとするとエラー条件が発生し、そのエラーはデータベース・マネージャーによって検出されて処理されません。

ストレージ・キーのサポートを使用可能にするには、新しい **DB2\_MEMORY\_PROTECT** レジストリー変数を使用します。

### 関連概念

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『バッファー・プール・メモリー保護 (POWER6 上で実行している AIX)』

### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『その他の変数』

---

## オンライン・データ整合性検査が改善されている

2 つの新しいキーワード **high** および **indexdata** を INSPECT コマンドおよび db2inspect API と併用して、索引の検査を実行できます。

INSPECT コマンドと db2inspect API は、データベースのページの整合性が取れているかどうかを調べることにより、データベースの構造上の整合性を検査します。この変更により、INSPECT コマンドと db2inspect API は、db2dart および RUNSTATS で実行される検査と同様の検査を実行して、データ破壊を検索できま

す。検査中のものを除くすべてのデータベース・オブジェクトにアクセスできます。INSPECT および db2inspect のデフォルトの動作は変更されていません。

#### 関連資料

管理 API リファレンスの『db2Inspect API - データベースの構造上の整合性の検査』

コマンド・リファレンスの『INSPECT コマンド』

---

## 索引データの整合性の問題の許容度が高くなっている

バージョン 9.5 では、索引が整合しなくなった場合に、データベースとインスタンスがダウンする代わりに、エラー・メッセージ (SQL0901N) がアプリケーションに戻されます。

このエラー・メッセージが戻された場合は、INSPECT コマンドか db2inspect API を使用して索引の不整合の原因をオンライン検査でき、その一方で他のアプリケーションは評価中でないこれらのデータベース・オブジェクトに引き続きアクセスできます。このタイプのエラーの許容は、パーティション表および非パーティション表の両方に対する正規索引と、マルチディメンション・クラスタリング索引に対する索引に限り使用可能になります。このタイプのエラーの許容は、マルチディメンション・クラスタリング・ブロック索引、複合ブロック索引、空間インデックス、および XML 索引に関しては使用可能になりません。

#### 関連資料

管理 API リファレンスの『db2Inspect API - データベースの構造上の整合性の検査』

コマンド・リファレンスの『INSPECT コマンド』

---

## 予期しないエラー時のデータベース回復力が改善されている

以前は、アプリケーション中のエラー条件によっては、データベースとインスタンス (データベース・マネージャー) がダウンすることがありました。現在は、データ保全性に影響がなく、エラーのあるアプリケーションに関する 1 つの DB2 エージェントを中断または終了できる場合は、そのアプリケーションのみ中断または終了されます。

トラップ、セグメント化違反、または他の例外が発生した場合、必要な診断情報がレビュー用に維持され、エラー条件がアプリケーションに戻され、DB2 エージェントの状況が変更され、アプリケーションがロールバックされます。エラーの影響を受けない他のアプリケーションは完了するまで実行できます。データベースとインスタンスをダウンして再始動する時点を決めることができます。

#### 関連概念

パーティションおよびクラスタリングのガイドの『DB2 のトラブルシューティング』



---

## 第 2 部 変更事項

このセクションでは、変更された機能、推奨されなくなった機能、および廃止された機能について説明します。新しいアプリケーションをコーディングする際、または既存のアプリケーションを修正する際には、これらの点を考慮に入れる必要があります。

これらの変更点を理解すれば、最新のアプリケーション開発やバージョン 9.5 へのマイグレーション計画に役立ちます。

### 129 ページの『第 16 章 変更された機能』

この章では、データベースのセットアップ、データベース管理、アプリケーション開発、および CLP コマンドとシステム・コマンドに関連した変更を含む、既存の DB2 機能に対する変更点について説明します。

### 173 ページの『第 17 章 推奨されない機能』

この章では、推奨されない機能をリストします。これは、以前はサポートされていたもののすでに推奨されなくなっており、将来のリリースで除去される可能性がある特定の機能またはフィーチャーです。

### 185 ページの『第 18 章 廃止された機能』

この章では、バージョン 9.5 でサポートされていないフィーチャーおよび機能をリストします。

### 191 ページの『第 19 章 DB2 バージョン 9.5 の使用に影響する DB2 バージョン 9.1 フィックスパックの変更』

この章では、本書の他のセクションでは説明されていない、バージョン 9.1 フィックスパック 3 (およびそれ以前のフィックスパック) の一部として追加または変更されてバージョン 9.5 にも適用されるフィーチャーおよび機能について説明します。



---

## 第 16 章 変更された機能

機能が変更される場合、通常は、デフォルト値が変更されたり、前のリリースとは異なる出力結果が生成されます。例えば、バージョン 9.1 で使用された SQL ステートメントが、バージョン 9.5 では異なる結果を生成する場合があります。

変更点はカテゴリ別に分類されています。

---

### 管理の変更のサマリー

#### 新規データベースのデフォルト・コード・ページは Unicode

新規 DB2 データベースを作成すると、デフォルトのコード・ページは Unicode です。以前は、デフォルトのデータベース・コード・ページは、データベースの作成に使用したアプリケーションの環境に基づいていました。

##### 詳細

Unicode のデータベースはどの言語からの文字でも収容できます。Unicode を使用すると、ビジネスの成長に伴って、データベースも言語や地域の境界を超えて拡張することができます。

Java や .NET など、多くの最新の開発環境のデフォルトは Unicode です。したがって、Unicode データベースはこれらの開発環境によりよく適合し、クライアント/サーバーの通信コストを削減できます。

##### 解決方法

Unicode ではないデータベースを作成する場合は、データベースのコード・セットおよびテリトリーを明示的に設定してください。

##### 関連概念

国際化対応ガイドの『DB2 Database for Linux, UNIX, and Windows での Unicode のインプリメンテーション』

##### 関連タスク

国際化対応ガイドの『データベースのコード・ページ、テリトリー、および照合の選択』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『データベースの作成』

#### 今後、データベース監査は SECADM 権限を必要とするようになった

今後、データベース・レベルでの監査の管理は、SECADM 権限を持つセキュリティ管理者のもとで行われるようになりました。今後、セキュリティ管理者はトラステッド・コンテキスト・オブジェクト、ロール、および監査ポリシーを管理（作成、変更、ドロップ、コメントを付ける）できるようになりました。

## 詳細

セキュリティー管理者は、拡張された権限を持つことに加え、データベースの監査を管理できる唯一のユーザーとなります。SYSADM 権限を持つシステム管理者は、この機能を持たなくなります。しかし、インスタンス・レベルの監査は引き続き管理できます。セキュリティー管理者は、データベースの監査を構成できるのと同様、次のタスクを実行するのに十分な SQL アクセス権限を持ちます。

- 抽出可能なアーカイブされた監査ログ・ファイルをリストする
- ARCHIVE コマンドを発行する
- 監査ログ・ファイルを区切りファイルに抽出する

## 解決方法

適切な権限が割り当てられているか確認します。今後もシステム管理者には、db2audit コマンドを使用して監査ログ・ファイルがディスクに配置された後、それらを管理する十分な権限がありますが、どのイベントを監査するかを制御することはできません。セキュリティー管理者には、db2audit コマンドを実行する十分な権限はありません。これには SYSADM 権限が必要です。

### 関連概念

49 ページの『監査機能のパフォーマンスと管理の機能拡張』

データベース・セキュリティー・ガイドの『監査ログの保管と分析』

### 関連資料

コマンド・リファレンスの『db2audit - 監査機能管理者ツール・コマンド』

## データ・コンプレッション・ディクショナリーが自動的に作成される

バージョン 9.5 では、データ・コンプレッション・ディクショナリーが自動的に作成されることがあります。

## 詳細

バージョン 9.1 では、クラシック (オフライン) 表再編成を実行してデータ・コンプレッション・ディクショナリーを手動で作成しなければなりません。まず、表の COMPRESS 属性を YES に設定しなければなりません。しかし、バージョン 9.5 では、COMPRESS 属性セットを YES に設定すれば、表に十分なデータが存在するようになったとき、表で自動的にデータ・コンプレッション・ディクショナリーの作成が行えるようになりました。結果として、INSERT ステートメントを使うと、INSERT または REPLACE オプションを使った LOAD コマンド、INSERT オプションを使った IMPORT コマンド、または REDISTRIBUTE コマンドによって自動的にデータ・コンプレッション・ディクショナリーが作成されます (辞書の作成が必要になるほどのデータが表に含まれているとデータベース・システムが判断した時)。データ・コンプレッション・ディクショナリーを作成するのに、明示的にクラシック (オフライン) 表再編成を実行する必要はありません。

バージョン 9.1 では、KEEPDICTIONARY オプションを使って表再編成を要求し、表の COMPRESS 属性を YES に設定した場合、データ・コンプレッション・ディクショナリーが存在しておらず、表に有効な長さのレコードが 1 つ存在してい



ると、データ・コンプレッション・ディクショナリーが作成されます。しかし、バージョン 9.5 では、**KEEPDICTIONARY** オプションを使って表再編成を要求し、表の **COMPRESS** 属性を **YES** に設定し、データ・コンプレッション・ディクショナリーが存在していない場合、データ・コンプレッション・ディクショナリーが作成されるには、表サイズがしきい値サイズ以上でなければならず、表に十分なデータがなければなりません。現行では、しきい値は **1 MB** に設定されており、表のデータ・ボリュームは少なくとも **70 %** でなければなりません。

バージョン 9.1 では、データ・コンプレッション・ディクショナリーを構築するのに、有効なレコード・サイズのすべてのデータ行が使用されます。データ・コンプレッション・ディクショナリーの構築時に表のすべてのレコードが最小レコード長よりも小さかった場合、**SQL2220W** エラー・メッセージが返されます。表に有効な長さのレコードが少なくとも **1** つあれば、データ・コンプレッション・ディクショナリーは構築されます。しかし、バージョン 9.5 では、どの行がデータ・コンプレッション・ディクショナリーの作成に関わるかを決定するレコード長の基準がありません。データ・コンプレッション・ディクショナリーの構築中に使用される、標本抽出されるすべてのデータ・レコードの長さが、適用可能最小レコード長よりも小さい場合、**SQL2220W** エラー・メッセージは生成されません。

## 解決方法

**COMPRESS** 属性を **YES** に設定して、表を作成したり変更した後、データ・コンプレッション・ディクショナリーを作成するために何かをする必要はありません。

### 関連概念

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『(コンプレッション) ディクショナリーの自動作成 (ADC)』

## 表書き込みイベント・モニターのターゲット表が変更された

表書き込みイベント・モニターのターゲット表内で、モニター・エレメントのサブセットのデータ・タイプまたは長さが、バージョン 9.1 とバージョン 9.5 の間で変更されました。これらの変更のために、ターゲット表内でさらに多くのモニター・データのキャプチャーが可能になりました。

### 詳細

以下のモニター・エレメントのデータ・タイプまたは長さが変更されました。

表 7. データ・タイプまたは長さが変更されたモニター・エレメント

エレメント名	説明	バージョン 9.1 のデータ・タイプおよび長さ	バージョン 9.5 のデータ・タイプおよび長さ
<b>appl_id</b>	アプリケーション ID	CHAR (64)	VARCHAR (64)
<b>appl_id_holding_lk</b>	ロック保留中のアプリケーション ID	CHAR (64)	VARCHAR (64)
<b>auth_id</b>	許可 ID	VARCHAR (30)	VARCHAR (128)
<b>corr_token</b>	DRDA 関連トークン	CHAR (64)	VARCHAR (64)
<b>creator</b>	アプリケーション作成者	VARCHAR (30)	VARCHAR (128)
<b>execution_id</b>	ユーザー・ログイン ID	VARCHAR (30)	VARCHAR (128)
<b>package_name</b>	パッケージ名	CHAR (8)	VARCHAR (128)

表7. データ・タイプまたは長さが変更されたモニター・エレメント (続き)

エレメント名	説明	バージョン 9.1 のデータ・タイプおよび長さ	バージョン 9.5 のデータ・タイプおよび長さ
rolled_back_appl_id	ロールバック・アプリケーション	CHAR (64)	VARCHAR (64)
table_schema	表スキーマ名	VARCHAR (30)	VARCHAR (128)

## 解決方法

表書き込みイベント・モニターの表にリストされているモニター・エレメントを照会するアプリケーションは、以下のステップを行わない限り、エラーを返します。

1. SYSCAT.EVENTTABLES ビューを照会して表書き込みイベント・モニターのターゲット表すべてを判別し、それらのターゲット表を名前変更します。これらのターゲット表内のデータが必要でなくなった場合は、それらの表を名前変更する代わりにドロップします。
2. イベント・モニターをドロップして、再定義します。
3. ターゲット表にアクセスするアプリケーションが、固定 CHAR データ・タイプではなく VARCHAR データ・タイプを受け入れるようにすべて変更します。あるいは、ホスト変数を変更して新規の長さを受け入れるようにします。これらのデータ・タイプに必要な構造タイプについて詳細は、「組み込み SQL アプリケーションの開発」の『C および C++ 組み込み SQL アプリケーションにおいてサポートされる SQL データ・タイプ』を参照してください。

## いくつかのシステム・カタログ・ビューと組み込みルーチンの追加および変更

バージョン 9.5 の新しいフィーチャーをサポートするために、システム・カタログ・ビューとシステム組み込みルーチンが追加および変更されました。

### システム・カタログ・ビューの変更点

バージョン 9.5 では、以下のシステム・カタログ・ビューが変更されました。カタログ・ビューのほとんどの変更点は、新しい列、列データ・タイプの変更、および列の長さの増加です。

- SYSCAT.ATTRIBUTES
- SYSCAT.CHECKS
- SYSCAT.COLAUTH
- SYSCAT.COLUMNS
- SYSCAT.DATATYPES
- SYSCAT.DBAUTH
- SYSCAT.DBPARTITIONGROUPS
- SYSCAT.EVENTMONITORS
- SYSCAT.EVENTS
- SYSCAT.EVENTTABLES
- SYSCAT.FUNCMAPPINGS

- SYSCAT.INDEXAUTH
- SYSCAT.INDEXEXPLOITRULES
- SYSCAT.INDEXEXTENSIONS
- SYSCAT.INDEXEXTENSIONPARMS
- SYSCAT.INDEXEXTENSIONMETHODS
- SYSCAT.INDEXES
- SYSCAT.NICKNAMES
- SYSCAT.PACKAGEAUTH
- SYSCAT.PACKAGEDEP
- SYSCAT.PASSTHROUGHAUTH
- SYSCAT.REFERENCES
- SYSCAT.ROUTINEAUTH
- SYSCAT.ROUTINEDEP
- SYSCAT.ROUTINESFEDERATED
- SYSCAT.ROUTINEPARMS
- SYSCAT.ROUTINES
- SYSCAT.SCHEMAAUTH
- SYSCAT.SCHEMATA
- SYSCAT.SECURITYPOLICIES
- SYSCAT.SEQUENCES
- SYSCAT.SEQUENCEAUTH
- SYSCAT.SURROGATEAUTHIDS
- SYSCAT.TABAUTH
- SYSCAT.TABCONST
- SYSCAT.TABDEP
- SYSCAT.TABLES
- SYSCAT.TABLESPACES
- SYSCAT.TBSPACEAUTH
- SYSCAT.TRIGDEP
- SYSCAT.TRIGGERS
- SYSCAT.TYPEMAPPINGS
- SYSCAT.USEROPTIONS
- SYSCAT.VIEWS
- SYSCAT.XSROBJECTAUTH
- SYSCAT.XSROBJECTS

バージョン 9.5 では、以下のシステム・カタログ・ビューが追加されました。

- SYSCAT.AUDITPOLICIES
- SYSCAT.AUDITUSE
- SYSCAT.CONTEXTATTRIBUTES

- SYSCAT.CONTEXTS
- SYSCAT.HISTOGRAMTEMPLATEBINS
- SYSCAT.HISTOGRAMTEMPLATES
- SYSCAT.ROLEAUTH
- SYSCAT.ROLES
- SYSCAT.SERVICECLASSES
- SYSCAT.THRESHOLDS
- SYSCAT.VARIABLEAUTH
- SYSCAT.VARIABLEDEP
- SYSCAT.VARIABLES
- SYSCAT.WORKLOADAUTH
- SYSCAT.WORKLOADCONNATTR
- SYSCAT.WORKACTIONS
- SYSCAT.WORKACTIONSETS
- SYSCAT.WORKCLASSES
- SYSCAT.WORKCLASSSETS
- SYSCAT.WORKLOADS

### システム定義の組み込みルーチンの変更点

バージョン 9.5 では、以下のシステム定義の組み込みルーチンが追加または変更されました。

表 8. システム定義の組み込みルーチン (新規および変更)

ルーチン名	変更点の要約
<ul style="list-style-type: none"> <li>• BITAND、BITOR、BITANDNOT、BITNOT、および BITXOR</li> <li>• COLLATION_KEY_BIT</li> <li>• COMPARE_DECFLOAT</li> <li>• DECFLOAT</li> <li>• DECODE</li> <li>• GREATEST</li> <li>• LEAST</li> <li>• MAX</li> <li>• MIN</li> <li>• NORMALIZE_DECFLOAT</li> <li>• NVL</li> <li>• QUANTIZE</li> <li>• RID_BIT and RID</li> <li>• TOTALORDER</li> </ul>	<p>これらの組み込み関数と同じ名前のユーザー定義関数が存在する場合、照会でそれらを完全に修飾しないと、解決パスでは代わりに新しい組み込み関数が呼び出されます。これらの組み込み関数を使用しない場合には、ユーザー定義関数を呼び出す際、スキーマ名を使って完全に修飾してください。</p>

表 8. システム定義の組み込みルーチン (新規および変更) (続き)

ルーチン名	変更点の要約
<ul style="list-style-type: none"> <li>• INSERT</li> <li>• LEFT</li> <li>• OVERLAY</li> <li>• RIGHT</li> <li>• STRIP</li> <li>• TRIM</li> </ul>	Unicode の可変文字サイズを使用可能にするために、これらの関数の新しい文字依存バージョンが <b>SYSIBM</b> スキーマに提供され、ストリング単位を示すパラメーターが 1 つ追加されました。デフォルト <b>SQL</b> パスを使用し、ストリング単位を指定しない場合には、これらの関数の新規バージョンが呼び出されます。新規バージョンの動作は以前のリリースの動作と互換性がありますが、いくつかの相違点もあります。以前のリリースと同じ関数を呼び出すには、 <b>SYSFUN</b> スキーマを使用して関数名を明示的に修飾してください。

## システム定義の管理ルーチンとビューの変更点

バージョン 9.5 では、以下の管理ビューとルーチンが変更されました。

- ADMIN\_CMD プロシージャ
- ADMINTABINFO 管理ビュー
- AUTHORIZATIONIDS 管理ビュー
- ENV\_PROD\_INFO 管理ビュー
- PRIVILEGES 管理ビュー
- SNAPAPPL 管理ビュー
- SNAPAPPL\_INFO 管理ビュー
- SNAPBP 管理ビュー
- SNAPDB 管理ビュー
- SNAPDBM 管理ビュー
- SNAPDYN\_SQL 管理ビュー
- SNAPTAB\_REORG 管理ビューおよび SNAP\_GET\_TAB\_REORG 表関数

バージョン 9.5 では、以下の管理ビューとルーチンが追加されました。

- ADMIN\_GET\_DBP\_MEM\_USAGE 表関数
- ADMINTABCOMPRESSINFO 管理ビューおよび ADMIN\_GET\_TAB\_COMPRESS\_INFO 表関数
- AUDIT\_ARCHIVE プロシージャおよび表関数
- AUDIT\_DELIM\_EXTRACT プロシージャ
- AUDIT\_LIST\_LOGS 表関数
- AUTH\_LIST\_AUTHORITIES\_FOR\_AUTHID 表関数
- AUTH\_LIST\_ROLES\_FOR\_AUTHID 表関数
- AUTOMAINT\_GET\_POLICY プロシージャ
- AUTOMAINT\_GET\_POLICYFILE プロシージャ
- AUTOMAINT\_SET\_POLICY プロシージャ
- AUTOMAINT\_SET\_POLICYFILE プロシージャ

- ENV\_FEATURE\_INFO 管理ビュー
- ENV\_SYS\_RESOURCES 管理ビュー
- EXPLAIN\_FORMAT\_STATS スカラー関数
- PD\_GET\_DIAG\_HIST 表関数
- SNAP\_GET\_APPL\_V95 表関数
- SNAP\_GET\_APPL\_INFO\_V95 表関数
- SNAP\_GET\_BP\_V95 表関数
- SNAP\_GET\_DB\_V95 表関数
- SNAP\_GET\_DBM\_V95 表関数
- SNAP\_GET\_DYN\_SQL\_V95 表関数
- WLM\_CANCEL\_ACTIVITY プロシージャ
- WLM\_CAPTURE\_ACTIVITY\_IN\_PROGRESS プロシージャ
- WLM\_COLLECT\_STATS プロシージャ
- WLM\_GET\_ACTIVITY\_DETAILS 表関数
- WLM\_GET\_QUEUE\_STATS 表関数
- WLM\_GET\_SERVICE\_CLASS\_AGENTS 表関数
- WLM\_GET\_SERVICE\_CLASS\_WORKLOAD\_OCCURRENCES 表関数
- WLM\_GET\_SERVICE\_SUBCLASS\_STATS 表関数
- WLM\_GET\_SERVICE\_SUPERCLASS\_STATS 表関数
- WLM\_GET\_WORKLOAD\_OCCURRENCE\_ACTIVITIES 表関数
- WLM\_GET\_WORKLOAD\_STATS 表関数

バージョン 9.5 では、以下の表関数が推奨されていません。

- ADMIN\_GET\_TAB\_INFO 表関数
- SNAP\_GET\_APPL 表関数
- SNAP\_GET\_APPL\_INFO 表関数
- SNAP\_GET\_BP 表関数
- SNAP\_GET\_DB\_V91 表関数
- SNAP\_GET\_DBM 表関数
- SNAP\_GET\_DYN\_SQL\_V91 表関数

「管理ルーチンおよびビュー」にある『使用すべきでない SQL 管理ルーチンおよびその置換ルーチンまたはビュー』のリストで、アプリケーションとスクリプトに影響を与える可能性のあるその他の変更点を確認してください。

## メモリー・ビジュアライザーによるメモリー最大消費量の表示

メモリー・ビジュアライザーでは、新しい **appl\_memory** 構成パラメーターを使用して、データベースによるアプリケーション・メモリー最大消費量を表示できるようになりました。また、更新された **instance\_memory** 構成パラメーターを使用して、インスタンスによるメモリー最大消費量を表示できるようになりました。

さらに、メモリー・ビジュアライザーでは、以下の構成パラメーターの値を表示できます (これらの AUTOMATIC 設定が可能になりました)。

- **mon\_heap\_sz**
- **stmtheap**
- **stat\_heap\_sz**
- **applheapsz**

以下の推奨されない構成パラメーターの値はバージョン 9.5 データベースでは表示されませんが、以前の DB2 バージョンからのデータベースでは引き続きサポートされます。

- **appgroup\_mem\_sz**
- **groupheap\_ratio**
- **app\_ctl\_heap\_sz**
- **query\_heap\_sz**

#### 関連概念

システム・モニター ガイドおよびリファレンスの『メモリー・ビジュアライザーの概要』

28 ページの『メモリー構成が単純化された』

#### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『instance\_memory - 「インスタンス・メモリー」構成パラメーター』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『appl\_memory - アプリケーション・メモリー構成パラメーター』

## バックアップ・イメージの読み取りおよび書き込み特権の変更

バックアップ・イメージの読み取りおよび書き込みができるのは、インスタンスの所有者のみになりました。

### 詳細

バージョン 9.5 以降では、バックアップ・イメージはファイル・モード 600 で Linux および UNIX オペレーティング・システム上で生成され、これらのシステム上では、読み取りおよび書き込み特権はインスタンスの所有者のみに付与されます。拡張セキュリティーが有効にされた Windows オペレーティング・システムでは、DB2ADMNS (および Administrators) グループのメンバーのみがバックアップ・イメージにアクセスできます。

以前のバージョンでは、Linux および UNIX オペレーティング・システム上のバックアップ・イメージはファイル・モード 640 で生成されたため、インスタンス所有者の 1 次グループの他のメンバーもバックアップ・イメージを読み取ることができました。このグループのメンバーにはバックアップ・イメージを読み取る権限がない場合もあるため、このメンバーはデフォルトで除外されるようになりました。

HADR フィーチャーで、異なるインスタンスがロード・コピー・イメージを読み取ることが必要であるため、ロード・コピー・イメージのファイル・モードは、バージョン 9.5 でも変更されていません。

## 解決方法

他のユーザーによるバックアップ・イメージへのアクセスが必要である場合は、バックアップの生成後に、ファイル許可を変更できます。

## DB2 ランチパッド上の「マイグレーション」ボタンの移動 (Windows)

DB2 ランチパッド上の「マイグレーション」ボタンが移動しました。

### 詳細

バージョン 9.5 より前の Windows オペレーティング・システムでは、「マイグレーション」ボタンは DB2 ランチパッドにありました。バージョン 9.5 以降では、「マイグレーション」ボタンは DB2 セットアップ・ウィザードの「処理する DB2 コピーの選択」パネルに現れます。

## 解決方法

マイグレーション・アクションにアクセスするには、次のようにします。

1. DB2 ランチパッドを開始します。
2. 「製品のインストール」タブから「既存の処理」をクリックします。
3. 「処理する DB2 コピーの選択」パネルで、マイグレーション対象の DB2 コピーを選択します。
4. 「DB2 セットアップ・ウィザードの起動」をクリックします。

## 表索引サイズの増加

空でないそれぞれの表の各索引にさらに 1 ページが含まれるようになりました。

### 詳細

新しいリアルタイム統計機能およびより高速なマルチディメンション・クラスタリング表 (MDC) のロールアウトには、追加の索引スペースが必要です。以前のリリースで作成された空でない表に索引がある場合、索引サイズは以下のいずれかが最初に発生したときに増加することがあります。

- 索引統計が RUNSTATS ユーティリティによって収集される。
- 据え置き索引クリーンアップでの MDC ロールアウトによって、索引がアクセスまたは更新される。
- 索引が再ビルドまたは再作成される。
- CLEANUP オプションを指定した REORG INDEX コマンドが索引に対して発行される。
- 大規模な索引の保守 (更新、削除、および挿入を含む) によって索引統計が変更される。

この場合、操作は失敗することがあり、その場合はエラー・メッセージ SQL0289N (表スペース *tablespace-name* の新規ページを割り振れません) が出されます。



## 解決方法

索引表スペースのサイズを大きくします。

### 関連タスク

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『DMS コンテナの追加または拡張』

### 関連資料

SQL リファレンス 第 2 巻の『ALTER TABLESPACE ステートメント』

SQL リファレンス 第 2 巻の『CREATE TABLESPACE ステートメント』

## 表の切り捨てによって動的ステートメント・キャッシュが無効になる

表の切り捨てが発生し、リアルタイム統計収集が使用可能になっていると、動的ステートメント・キャッシュ内の切り捨てられた表に依存するすべての項目は無効になります。

### 詳細

REPLACE オプションを指定して IMPORT コマンドを使用すると、データ・オブジェクトを切り捨てることによって表からすべての既存のデータが削除され、インポートされたデータが挿入されます。以前は、表切り捨てによってステートメント・キャッシュが無効化されることはありませんでした。この動作は、リアルタイム統計収集が使用可能になったときに変更されました。

動的ステートメントが無効化されることにより、それらが次に実行されるときには強制的に再コンパイルされます。これによって、最新の統計を使用してより最適なアクセス・プランを選択する機会が得られます。しかし、これがパフォーマンスへの影響を引き起こす可能性もあります。

### 解決方法

潜在的なパフォーマンスへの影響に注意してください。

## REFRESH TABLE および SET INTEGRITY ステートメントの ALLOW NO ACCESS オプションについて並行性が改善された

ALLOW NO ACCESS オプションを指定して REFRESH TABLE ステートメントまたは SET INTEGRITY ステートメントを使用する場合、処理中のターゲット表で Z ロックの代わりに X ロックが取得されるようになりました。

### 詳細

ターゲット表において Z ロックではなく X ロックを取得することにより、非コミット読み取り分離レベルで並行して実行しているトランザクションは、REFRESH TABLE ステートメントまたは SET INTEGRITY ステートメントによって処理中のターゲット表に読み取りアクセスができるようになりました。さらに、非コミット読み取り分離レベルのみを必要とする並行して実行中のユーティリティーもターゲット表を読み取ることができます。

## 解決方法

コードの変更をする必要はありません。処理中のターゲット表にアクセスするアプリケーションの並行性が改善されます。

## データベース・セットアップ変更のサマリー

### いくつかの構成パラメーターが変更された

バージョン 9.5 には、さらに効率的なシステム管理を可能にする、いくつかの新規および変更された構成パラメーターが含まれています。加えて、DB2 機能の変更、新規パラメーターの導入、またはサポートの停止を反映して、いくつかの構成パラメーターが非推奨になるか、または廃止されました。

### 新規構成パラメーター

新規のフィーチャーおよび機能のため、バージョン 9.5 には、いくつかの新規構成パラメーターが含まれています。

表 9. バージョン 9.5 の新規構成パラメーター

パラメーター名	説明	DB2 Connect に該当する DBM 構成パラ メーターか?	詳細
<b>appl_memory</b>	アプリケーション・メモリー	いいえ	すべての DB2 データベース・エージェントにより、サービス・アプリケーション要求に対して割り振られるアプリケーション・メモリーの最大量を制御できるようにします。デフォルトでは、この値は <b>AUTOMATIC</b> に設定されており、データベース・パーティションが割り振るメモリーの合計が <b>instance_memory</b> 制限内である場合、すべてのアプリケーション・メモリー要求は許可されます。
<b>auto_del_rec_obj</b>	リカバリー・オブジェクトの自動削除	いいえ	関連したリカバリー履歴ファイルの項目が整理された場合に、データベース・ログ・ファイル、バックアップ・イメージ、およびロード・コピー・イメージが削除されるかどうかを指定します。
<b>auto_stmt_stats</b>	自動ステートメント統計	いいえ	リアルタイム統計収集機能を使用可能または使用不可にします。これは <b>auto_runstats</b> 構成パラメーターの子パラメーターで、親パラメーターが使用可能である場合にのみ使用可能です。
<b>cluster_mgr</b>	クラスター・マネージャー名	はい	データベース・マネージャーが指定されたクラスター・マネージャーへ増分クラスター構成変更を通信することを可能にします。
<b>decflt_rounding</b>	浮動小数点の丸め	いいえ	浮動小数点型 (DECFLOAT) の丸めモードを指定できます。丸めモードの影響があるのは、サーバーでの浮動小数点の演算と、LOAD コマンドを使用するときです。

表9. バージョン 9.5 の新規構成パラメーター (続き)

パラメーター名	説明	DB2 Connect に該当する DBM 構成パラ メーターか?	詳細
<b>enable_xmlchar</b>	XML への変換を有効にする	いいえ	SQL ステートメント内の非 BIT DATA CHAR (または CHAR 型) の式に対して XMLPARSE 操作を実行できるかどうかを決定します。非 Unicode データベースで pureXML フィーチャーを使用すると、SQL ストリング・データがクライアント・コード・ページからデータベース・コード・ページに変換され、次いで内部ストレージ用に Unicode に変換されるときに、XMLPARSE 関数によって文字置換が起こることがあります。このパラメーターは、置換文字の危険のない Unicode データベースでも有効です。
<b>hadr_peer_window</b>	HADR ピア・ウィンドウ構成	いいえ	構成された期間中、1 次データベースがスタンバイ・データベースとの接続を失ったとしても、HADR 1 次/スタンバイ・データベースの対がピア状態にあるかのように動作させることにより、データの整合性の保持に役立ちます。
<b>wlm_collect_int</b>	ワークロード管理の収集間隔	はい	ワークロード管理統計収集の試行およびそのような統計のリセット試行のインターバルを分単位で指定します。指定するインターバルは、カタログ・パーティションにのみ使用されます。ワークロード管理モニター機能によって収集される統計を使用して、短期および長期両方のシステムの動作をモニターすることができます。

### 変更された構成パラメーター

以下の表に、デフォルト値が変更された構成パラメーターをリストします。これらのパラメーターはすべて動的に更新することができ、データベース・インスタンスを停止して再始動する必要はありません。

表 10. デフォルト値の変更された構成パラメーター

パラメーター名	説明	DB2 Connect に該当する DBM 構成パラ メーターか?	バージョン 9.1 のデフ ォルト値	バージョン 9.5 のデフォルト値
<b>applheapsz</b>	アプリケーション・ヒープ・サイズ	いいえ	ローカルおよびリモート・クライアントのある 32 ビットおよび 64 ビットのデータベース・サーバー: 256  ローカルおよびリモート・クライアントのある 32 ビットのパーティション・データベース・サーバー: 64  ローカルおよびリモート・クライアントのある 64 ビットのパーティション・データベース・サーバー: 128	AUTOMATIC
<b>catalogcache_sz</b>	カタログ・キャッシュ・サイズ	いいえ	ページ割り振りの計算に使用されるデフォルト値は、 <b>maxappls</b> 構成パラメーターに指定された値の 4 倍です。	ページ割り振りの計算に使用されるデフォルト値は、 <b>maxappls</b> 構成パラメーターに指定された値の 5 倍です。
<b>comm_bandwidth</b>	通信帯域幅	はい	デフォルト値は、高速スイッチが使用されているかどうかに基づいて計算されます。	デフォルト値は、基礎となる通信アダプターの速度に基づいて計算されます。ギガビット・イーサネットを使用するシステムでは、値 100 が期待されます。
<b>database_memory</b>	データベース共有メモリー・サイズ	いいえ	AIX および Windows オペレーティング・システム: AUTOMATIC  Linux、HP-UX、Solaris オペレーティング・システム: COMPUTED	Linux オペレーティング・システムでは、このパラメーターのデフォルト値は、RHEL5 および SUSE 10 SP1 以降のディストリビューションの場合、AUTOMATIC 設定です。他のすべての検証済み Linux ディストリビューションでは、カーネルが AUTOMATIC をサポートしない場合、このパラメーターのデフォルト値は COMPUTED です。  その他のすべてのオペレーティング・システム: AUTOMATIC

表 10. デフォルト値の変更された構成パラメーター (続き)

パラメーター名	説明	DB2 Connect に該当する DBM 構成パラ メーターか?	バージョン 9.1 のデフ ォルト値	バージョン 9.5 のデフォルト値
<b>dbheap</b>	データベース・ヒ ープ	いいえ	UNIX: 1200  ローカルおよびリモ ート・クライアントのあ る Windows データベ ース・サーバー: 600  ローカル・クライアン トのある Windows 64 ビットのデータベ ース・サーバー: 600  ローカル・クライアン トのある Windows 32 ビットのデータベ ース・サーバー: 300	AUTOMATIC
<b>fenced_pool</b>	fenced プロセス の最大数	はい	<b>max_coordagents</b> の値	AUTOMATIC
<b>max_connections</b>	クライアント接続 の最大数	はい	<b>max_coordagents</b> の値	AUTOMATIC
<b>mon_heap_sz</b>	データベース・シ ステム・モニタ ー・ヒープ・サイ ズ	はい	UNIX: 90  ローカルおよびリモ ート・クライアントのあ る Windows データベ ース・サーバー: 66  ローカル・クライアン トのある Windows デ ータベース・サーバー: 46	AUTOMATIC
<b>num_poolagents</b>	エージェント・プ ール・サイズ	はい	<b>maxagents/2</b> の値	AUTOMATIC
<b>stat_heap_sz</b>	統計ヒープ・サイ ズ	いいえ	4384	AUTOMATIC
<b>stmtheap</b>	ステートメント・ ヒープ・サイズ	いいえ	32 ビット・プラットフ ォーム: 2048  64 ビット・プラットフ ォーム: 4096	AUTOMATIC

バージョン 9.5 では、以下の構成パラメーターは、動作が変更されたか、または範囲が更新されました。

表 11. 動作または範囲の変更された構成パラメーター

パラメーター名	説明	DB2 Connect に該当する DBM 構成パラメーターか?	バージョン 9.5 の変更点
<b>applheapsz</b>	アプリケーション・ヒープ・サイズ	いいえ	以前のリリースでは、このパラメーターはデータベース・エージェントごとのメモリー容量を示していました。現在、このパラメーターは、アプリケーションのメモリーの合計容量を示します。
<b>federated_async</b>	照会あたりの最大非同期 TQ 数	はい	範囲の最大値は、 <b>maxagents/4</b> の値ではなくなりました。現在は 32 767 です。
<b>instance_memory</b>	インスタンス・メモリー	はい	範囲の最大値は、4 294 967 295 ではなくなりました。32 ビット・プラットフォームでは 1 000 000、64 ビット・プラットフォームでは 68 719 476 736 になりました。インスタンス・メモリーは、DBMS メモリー・セットのサイズだけではなく、パーティション全体のメモリー消費量の制限を表すようになりました。
<b>intra_parallel</b>	パーティション内並列処理を使用可能にする	はい	このパラメーターは現在でも SQL アクセス・プラン SMP 並列処理を制御しますが、索引作成並列処理は制御しなくなりました。その代わりに、索引作成並列処理は、要求時に動的に使用可能にされます。ただし、索引マネージャーによる予備的な検査に基づいて、CREATE INDEX 操作の間に一時的に使用可能になるだけです。
<b>max_coordagents</b>	コーディネーター・エージェントの最大数	はい	AUTOMATIC のサポートに加えて、範囲の最大値は、 <b>maxagents</b> の値マイナス <b>num_initagents</b> の値ではなくなりました。64 000 になりました。
<b>maxfilop</b>	データベースに対して同時に開くことのできるファイルの最大数	いいえ	このパラメーターは、データベースに対して同時に開くことのできるファイル・ハンドルの最大数を表すようになりました。以前のリリースでは、このパラメーターは、各データベース・エージェントに対して、同時に開くことのできるファイル・ハンドルの最大数を示していました。
<b>num_initagents</b>	プール中のエージェントの初期数	はい	範囲の最大値は、 <b>num_poolagents</b> の値ではなくなりました。現在は 64 000 です。
<b>num_initfenced</b>	fenced プロセスの初期数	はい	範囲の最大値は、 <b>max_connections + (maxagents - max_coordagents)</b> の和の値ではなくなりました。64 000 になりました。
<b>sorthheap</b>	ソート・ヒープ・サイズ	いいえ	OLAP 関数は、アプリケーション・ヒープ・メモリーではなく、ソート・ヒープ・メモリーを使用してメモリー・リソースの限度を高くするようになりました。

### 推奨されなくなった構成パラメーターおよび廃止された構成パラメーター

機能の変更、新規パラメーターの導入、またはサポートの停止のために、いくつかの構成パラメーターが非推奨になるか、または廃止されました。

関連する機能が非推奨になったか、または新規の構成パラメーターが導入されたため、以下の構成パラメーターは非推奨になりました。

表 12. 非推奨になった構成パラメーターの要約

パラメーター名	説明	DB2 Connect に該当する DBM 構成パラ メーターか?	詳細および解決方法
<b>agentpri</b>	エージェント 優先順位	はい	ワークロード管理の新規機能が導入されたために、この構成パラメーターの必要性が減少しました。したがって、この構成パラメーターは将来のリリースで除去される可能性があります。
<b>app_ctl_heap_sz</b>	アプリケーション制御ヒープ・サイズ	いいえ	新規メモリー・モデルでは、単一のアプリケーション共有メモリー・セットがデータベースごと (各パーティションに) に作成され、これに接続するアプリケーションすべてによって共有されます。バージョン 9.5 では、これら 3 つの構成パラメーターを使用して各アプリケーション・グループに入れることのできるアプリケーション数を決定する代わりに、新規構成パラメーター <b>appl_memory</b> を使用してアプリケーション・メモリー全体の最大消費量を設定します。
<b>appgroup_mem_sz</b>	アプリケーション・グループ・メモリー・セットの最大サイズ	いいえ	
<b>groupheap_ratio</b>	アプリケーション・グループ・ヒープ用メモリーのパーセント	いいえ	
<b>maxagents</b>	最大エージェント数	いいえ	<b>maxagents</b> パラメーターを使用してデータベース・マネージャー・エージェントの最大数を制御する代わりに、インスタンスへの接続の総数に制限を設定します。
<b>maxcagents</b>	最大並行エージェント数	いいえ	同様に、 <b>maxcagents</b> パラメーターを使用して最大並行データベース・マネージャー・エージェント数を制御するのではなく、接続コンセントレーターおよび DB2 ワークロード管理の機能を使用して、システム上のワークロードおよびリソースを管理してください。

表 12. 非推奨になった構成パラメーターの要約 (続き)

パラメーター名	説明	DB2 Connect に該当する DBM 構成パラ メーターか?	詳細および解決方法
<b>logretain</b>	ログ保持使用 可能	いいえ	これらのパラメーターは両方とも、1 次ログ・アーカイブ方式構成パラメーター <b>logarchmeth1</b> に置き換えられました。 <b>logretain</b> を使用してロールフォワード・リカバリーのためにアクティブ・ログ・ファイルを保持する代わりに、LOGRETAIN 値を <b>logarchmeth1</b> に指定します。同様に、 <b>userexit</b> を使用して、ユーザー出口プログラムを通してログ・アーカイブを有効にする代わりに、USEREXIT 値を <b>logarchmeth1</b> で指定します。  <ul style="list-style-type: none"> <li>• <b>logretain</b> をオンにすると LOGRETAIN 値が <b>logarchmeth1</b> に設定されます。</li> <li>• <b>userexit</b> をオンにすると USEREXIT 値が <b>logarchmeth1</b> に設定されます。<b>logretain</b> および <b>userexit</b> の両方をオンにすると、<b>logarchmeth1</b> に USEREXIT 値が設定されません。</li> </ul> 例えばコマンド <code>update db cfg using logretain on</code> を発行すると、LOGARCHMETH1 が LOGRETAIN に設定されます。
<b>userexit</b>	ユーザー出口 使用可能	いいえ	<ul style="list-style-type: none"> <li>• <b>userexit</b> をオンにすると USEREXIT 値が <b>logarchmeth1</b> に設定されます。<b>logretain</b> および <b>userexit</b> の両方をオンにすると、<b>logarchmeth1</b> に USEREXIT 値が設定されません。</li> </ul> 例えばコマンド <code>update db cfg using logretain on</code> を発行すると、LOGARCHMETH1 が LOGRETAIN に設定されます。
<b>query_heap_sz</b>	照会ヒープ・ サイズ	はい	この構成パラメーターは非推奨になりました。これは、DB2RA プロトコルを使用する DB2 Universal Database for Linux, Windows, and UNIX バージョン 7 (またはそれ以前) のクライアントをサポートするために使用されていましたが、このプロトコルはサポートされなくなったためです。
<b>numsegs</b>	SMS コンテナ のデフォル ト数	いいえ	CREATE DATABASE コマンドで SMS 表スペースの複数のコンテナを指定するため、この構成パラメーターは非推奨になりました。

以下の構成パラメーターは廃止されました。

表 13. 廃止された構成パラメーターの要約

パラメーター名	説明	DB2 Connect に該当する DBM 構成パ ラメーターか?	詳細および解決方法
<b>estore_seg_sz</b>	拡張ストレージ・メモリ ー・セグメント・サイズ	いいえ	拡張ストレージ・フィーチャーのサポートは廃止されました。拡張ストレージ構成パラメーターの使用は除去してください。メモリをさらに割り振ることができるようにする場合は、64 ビットの実オペレーティング・システムへのアップグレードを考慮してください。
<b>num_estore_segs</b>	拡張ストレージ・メモリ ー・セグメント数	いいえ	拡張ストレージ・フィーチャーのサポートは廃止されました。拡張ストレージ構成パラメーターの使用は除去してください。メモリをさらに割り振ることができるようにする場合は、64 ビットの実オペレーティング・システムへのアップグレードを考慮してください。



表 13. 廃止された構成パラメーターの要約 (続き)

パラメーター名	説明	DB2 Connect に該当する DBM 構成パ ラメーターか?	詳細および解決方法
priv_mem_thresh	専用メモリーしきい値	はい	現在データベース・マネージャーは マルチスレッド方式を使用するた め、このパラメーターは必要でなく なりました。

#### 関連資料

コマンド・リファレンスの『RESET DATABASE CONFIGURATION コマンド』  
データ・サーバー、データベース、およびデータベース・オブジェクトのガイド  
の『構成パラメーターのサマリー』

「マイグレーション・ガイド」の『DB2 サーバー動作の変更点』

## いくつかのレジストリー変数および環境変数の変更

バージョン 9.5 では、複数のレジストリー変数および環境変数を変更されました。

### 新規デフォルト

表 14. デフォルト値が新しくなったレジストリー変数

レジストリー変数	バージョン 9.1 のデフォルト 設定	バージョン 9.5 のデフォルト 設定
DB2INSTPROF	NULL	Windows Vista オペレーティ ング・システムでは ProgramData¥IBM¥DB2¥、お よび Windows 2003 または XP オペレーティング・シス テムでは Documents and Settings¥All Users¥Application Data¥IBM¥DB2¥。

## 新規値

表 15. 値が新しくなったレジストリー変数

レジストリー変数	新規値
<b>DB2_MDC_ROLLOUT</b>	この変数には新規値 DEFER、および新規記述 IMMEDIATE (現行の ON、YES、1、および TRUE 設定と同じ) があります。マルチディメンション・クラスタリング表のロールアウト削除が、即時索引クリーンアップ・ロールアウトを使用するか (デフォルト動作)、据え置き索引クリーンアップ・ロールアウトを使用するかを制御できるようになりました。詳しくは、55 ページの『索引の据え置きクリーンアップ・オプションを使用したより高速な MDC ロールアウト削除』を参照してください。
<b>DB2_WORKLOAD</b>	この変数には 2 つの新規値 1C および TPM があります。これらの設定により、1C によって提供されたアプリケーションおよび Tivoli Provisioning Manager 用のデータベースにあるレジストリー変数のセットを構成できるようになります。

## 変更された動作

表 16. 動作が変わったレジストリー変数

レジストリー変数	変更された動作
<ul style="list-style-type: none"> <li>• <b>DB2_RESOURCE_POLICY</b></li> <li>• <b>DB2_LARGE_PAGE_MEM</b></li> <li>• <b>DB2_PINNED_BP</b></li> <li>• <b>DB2PRIORITIES</b></li> </ul>	非ルート・インストールの場合、これらのレジストリー変数はサポートされません。これらに対する更新は許可されません。詳しくは、109 ページの『非ルート・ユーザーが DB2 製品のインストールと構成を実行できる (Linux および UNIX)』を参照してください。
<b>DB2CLIINIPATH</b>	DB2 バージョン 9.5 を Windows オペレーティング・システムにインストールする際にマイグレーション・アクションを選択した場合、 <b>DB2CLIINIPATH</b> は、マイグレーションしようとする DB2 コピー用の既存の db2cli.ini 構成ファイルの場所に設定されません。しかし、DB2 コピーをインストールした後、インスタンスをマイグレーションする場合、構成ファイルのマイグレーション前の位置を維持したい場合、この変数をその場所に設定してください。

表 16. 動作が変わったレジストリー変数 (続き)

レジストリー変数	変更された動作
<b>DB2MEMMAXFREE</b>	<p>このレジストリー変数を Linux および UNIX オペレーティング・システムに設定した場合、これはインスタンスのマイグレーションによって NULL に設定されます。現在データベース・マネージャーはスレッド化エンジン・モデルを使用し、専用メモリーはすべてのデータベース・マネージャー・スレッドによって共有されるので、この変数の設定は今後必要ありません。</p> <p>注: この変数を設定しないでください。設定すると、パフォーマンスが悪くなり、予期しない動作が発生する可能性があります。</p> <p>詳しくは、27 ページの『単純化されたマルチスレッド化アーキテクチャーにより総所有コスト (TCO) が減っている』を参照してください。</p>
<b>DB2_EXTENDED_IO_FEATURES</b>	<p>オンライン・バックアップの入出力優先順位を設定するのに、このレジストリー変数オプションを使うことはできなくなりました。入出力優先順位の設定に使われていた API はプロセス・ベースです。バージョン 9.5 はスレッド・ベース・モデルを使用し、現在のところスレッド・ベース入出力優先順位の設定を行うための相当する API はありません。</p> <p>詳しくは、27 ページの『単純化されたマルチスレッド化アーキテクチャーにより総所有コスト (TCO) が減っている』を参照してください。</p>
<b>DB2_USE_DB2JCCT2_JROUTINE</b>	<p>このレジストリー変数のデフォルト設定は、Java ストアード・プロシージャやユーザー定義関数のデフォルト・ドライバーが IBM Data Server Driver for JDBC and SQLJ であることを意味するようになりました。以前のリリースでは、デフォルトの JDBC ドライバーは DB2 JDBC Type 2 ドライバーでした。詳しくは、156 ページの『変更された Java ルーチン用のデフォルト JDBC ドライバー』を参照してください。</p>

## 新規変数

以下はバージョン 9.5 の新しいレジストリー変数です。

表 17. 追加されたレジストリー変数

レジストリー変数	説明
<b>DB2_CAPTURE_LOCKTIMEOUT</b>	このレジストリー変数はロック・タイムアウト発生時にそれに関する記述情報をログに記録するのに指定します。各ロック・タイムアウトに対して、テキスト・レポートが書き込まれ、ファイルに保管されます。詳しくは、38 ページの『ロック・タイムアウト診断が拡張された』を参照してください。
<b>DB2_EVMON_EVENT_LIST_SIZE</b>	このレジストリー変数は特定のイベント・モニターに書き込むためにキューに入れることのできる最大バイト数を指定します。この限度に達すると、イベント・モニター・レコードを送信しようとしているエージェントはキュー・サイズがこのしきい値よりも小さくなるまで待機します。詳しくは、41 ページの『ワークロード管理の改善による制御の向上』を参照してください。
<b>DB2FODC</b>	このレジストリー変数は、以前 <b>DB2FFDC</b> レジストリー変数で使用可能だった機能を拡張することによって、First Occurrence Data Collection (FODC) で使用されるトラブルシューティング関連のパラメーターのセットを制御します。これはユーザーおよび IBM サービス分析者が、FODC シナリオ中に DB2 製品によって収集されるものを制御することを意図して設計されたものです。 <b>DB2FODC</b> を使用して、障害状態時にさまざまな分野のデータ収集を制御してください。詳しくは、123 ページの『データ収集ツールで予期しないエラーを追跡する』を参照してください。

表 17. 追加されたレジストリー変数 (続き)

レジストリー変数	説明
<b>DB2_MEMORY_PROTECT</b>	<p>このレジストリー変数は、ストレージ・キーを使用して無効なメモリー・アクセスによって引き起こされるバッファ・プール内のデータ破損を防ぐメモリー保護フィーチャーを有効にします。DB2 エンジン・スレッドがバッファ・プール・メモリーへのアクセスを行おうとするタイミングを識別することにより、メモリー保護が働きます。</p> <p><b>DB2_MEMORY_PROTECT</b> を YES に設定すると、DB2 エンジン・スレッドがバッファ・プール・メモリーに不正にアクセスを行おうとするたびに、そのエンジン・スレッドはトラップします。詳しくは、125 ページの『予期しないエラー時のデータベース回復力が改善されている』を参照してください。</p>
<b>DB2_OPTSTATS_LOG</b>	<p><b>DB2_OPTSTATS_LOG</b> は、統計収集関連のアクティビティーのモニターや分析に使用される統計イベント・ロギング・ファイルの属性を指定します。</p> <p><b>DB2_OPTSTATS_LOG</b> を設定しないか、これを ON に設定すると、統計イベント・ロギングが有効になり、システム・パフォーマンスをモニターしたり、よりの確な問題判別を行うために履歴を維持したりできるようになります。詳しくは、26 ページの『リアルタイム統計収集により確実に最新の統計が最適化に使用される』を参照してください。</p>
<b>DB2_SET_MAX_CONTAINER_SIZE</b>	<p>このレジストリー変数はデータベースの最大コンテナ・サイズを設定します。このレジストリー変数を使用すると、自動ストレージ管理の表スペースが指定のコンテナ限度に達したとき、既存のストレージ・パスに基づいて新しいコンテナがデータベースに作成されます。詳しくは、38 ページの『表スペースのスペース使用効率が向上している』を参照してください。</p>
<b>DB2_SYSTEM_MONITOR_SETTINGS</b>	<p>このレジストリー変数は、DB2 モニターのさまざまな分野の動作を変更することを可能にするパラメーターのセットを制御します。例えば、パラメーター <b>OLD_CPU_USAGE</b> は、インスタンスが Linux プラットフォームで CPU 時間を取得する方法を制御します。</p>

表 17. 追加されたレジストリー変数 (続き)

レジストリー変数	説明
<b>DB2_THREAD_SUSPENSION</b>	このレジストリー変数は、DB2 スレッド中断フィーチャーを有効または無効にできます。この変数によって、DB2 インスタンスが障害のあるエンジン・スレッド (不正にバッファ・プール・メモリーにアクセスしようとしたスレッド) を中断することによってトラップを維持するかどうかを制御できるようになります。詳しくは、125 ページの『予期しないエラー時のデータベース回復力が改善されている』を参照してください。
<b>DB2_UPDDBCFG_SINGLE_DBPARTITION</b>	このレジストリー変数によって、データベース更新およびリセットが特定のパーティションにのみ影響を与えるようにするかどうか指定できます。変数を設定しなかった場合、データベース構成への更新および変更はすべてのデータベース・パーティションに適用されます。詳しくは、28 ページの『複数のパーティション間のデータベース構成が単純化された』を参照してください。

#### 関連概念

173 ページの『レジストリー変数および環境変数のいくつかは推奨されない』

187 ページの『いくつかのレジストリー変数および環境変数の廃止』

## 並行入出力および直接入出力はデフォルトで使用可能になる (AIX、Linux、Solaris、および Windows)

バージョン 9.5 より前では、FILE SYSTEM CACHING は、CREATE TABLESPACE ステートメントおよび CREATE DATABASE コマンドで作成された表スペースすべてのデフォルト属性でした。バージョン 9.5 では、NO FILE SYSTEM CACHING 属性が、使用できるところでシステム構成に暗黙的に使用されます。

### 詳細

FILE SYSTEM CACHING および NO FILE SYSTEM CACHING 属性は、入出力操作をファイル・システム・レベルでキャッシュするかどうかを指定します。これらの属性は、次のインターフェースを使用して指定します。CREATE TABLESPACE ステートメント、CREATE DATABASE コマンド、および sqlcrea() API (SQLETSDESC 構造の **sqlfscaching** フィールドを使用)。

バージョン 9.5 では、CREATE TABLESPACE ステートメントまたは CREATE DATABASE コマンドで指定しない場合、多くのシステム構成で、NO FILE SYSTEM CACHING がデフォルトで使用されます。CIO、DIO、またはファイル・システム・キャッシングをサポートするシステム構成のリストは、『ファイル・システム・キャッシング構成』を参照してください。

## 解決方法

新規の動作を使用しない場合は、表スペースの作成時に FILE SYSTEM CACHING を指定します。

新規の動作で性能低下が生じ、セルフチューニング・メモリーが使用不可である場合は、以下の回避策のいずれかを行ってください。

- セルフチューニング・メモリーを使用可能にし、バッファ・プール・サイズおよび **database\_memory** 構成パラメーターを AUTOMATIC に設定する。
- バッファ・プールのサイズを手動で大きくする。
- ALTER TABLESPACE ステートメントを FILE SYSTEM CACHING 属性と共に使用して、DIO および CIO を使用不可にする。

### 関連概念

データベース・パフォーマンスのチューニングの『複数のデータベース・バッファ・プールの管理』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『セルフチューニング・メモリー』

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『ファイル・システム・キャッシングを使用しない表スペース』

### 関連資料

SQL リファレンス 第 2 巻の『ALTER BUFFERPOOL ステートメント』

SQL リファレンス 第 2 巻の『ALTER TABLESPACE ステートメント』

SQL リファレンス 第 2 巻の『CREATE TABLESPACE ステートメント』

管理 API リファレンスの『sqlcrea API - データベースの作成』

コマンド・リファレンスの『CREATE DATABASE コマンド』

## 拡張セキュリティーでは、ユーザーが DB2ADMNS または DB2USERS グループに属する必要がある (Windows Vista)

Windows Vista で拡張セキュリティーを有効にした場合、ユーザーがローカル DB2 コマンドとアプリケーションを実行するためには、ユーザーは DB2ADMNS または DB2USERS グループに属している必要があります。これは、ローカル管理者がデフォルトで持つ特権を制限する特別なセキュリティー・フィーチャー (ユーザー アクセス制御) があるためです。

### 詳細

ユーザーがこれらのグループの 1 つに属していない場合、ローカル DB2 構成またはアプリケーション・データに対する読み取りアクセス権限が与えられません。

### 解決方法

- 拡張セキュリティーを有効にした場合、ローカル DB2 アプリケーションやツールを実行する必要があるユーザーを DB2ADMNS または DB2USERS グループに追加してください。グループ・メンバーシップを変更すると、変更内容はそのユーザーが次回にログインするとき有効になります。

- オペレーティング・システムのローカル管理者権限を必要とする DB2 コマンドおよびツールを起動するには、「**DB2 コマンド・ウィンドウ - 管理者 (DB2 Command Window - Administrator)**」ショートカットを使用してください。

#### 関連概念

データベース・セキュリティー・ガイドの『DB2ADMNS と DB2USERS グループの使用による拡張 Windows セキュリティー』

#### 関連資料

DB2 サーバー機能 概説およびインストールの『DB2 サーバー製品のインストールに必要なユーザー・アカウント (Windows)』

## 構成およびランタイム・データ・ファイルのデフォルト・ロケーションが変更された (Windows)

インスタンス・ディレクトリーおよび db2cli.ini ファイルといった、すべての構成およびランタイムのデータ・ファイルのデフォルト・ロケーションが変更され、Windows Vista の認定の要件に準拠するようになりました。

### 詳細

デフォルト・ロケーションは以下のとおりです。

- Windows XP および Windows 2003 オペレーティング・システム: Documents and Settings¥All Users¥Application Data¥IBM¥DB2¥Copy name
- Windows Vista オペレーティング・システム: ProgramData¥IBM¥DB2¥ Copy name

## フィックスパックのインストール後に手動のステップの必要がない (Linux および UNIX)

デフォルトで、installFixPack コマンドにより、特定のインストール・パスに関連したインスタンスおよび DAS が自動的に更新されるようになりました。データベースの再接続時またはアプリケーションの再始動時に、BIND コマンドも自動的に起動されるようになりました。

### 詳細

以前のリリースでは、フィックスパックの適用後にインスタンスを更新する必要があり、手動で行うステップがありました。また、手動でパッケージをバインドしなければなりませんでした。

### 解決方法

フィックスパックのインストール後にインスタンスおよび DAS を更新するデプロイメント・スクリプトがあれば、変更します。

#### 関連タスク

問題判別ガイドの『フィックスパックの適用』



## いくつかの構成パラメーターは単純化されたメモリー構成によって影響を受ける

バージョン 9.5 の単純化されたメモリー構成により、データベース・マネージャーはいくつかの構成パラメーターを `AUTOMATIC` に設定するようになりました。特に、これは、インスタンスのマイグレーションまたは作成中およびデータベースのマイグレーションまたは作成中に行われます。

### 詳細

`AUTOMATIC` 設定は、データベース構成パラメーターがシステム・リソースに応じて自動的にチューニングされることを示します。以下の表に、影響を受ける構成パラメーターをリストします。

表 18. バージョン 9.5 で `AUTOMATIC` に設定される構成パラメーター

構成パラメーター	インスタンス・マイグレーションまたはインスタンス作成時に <code>AUTOMATIC</code> に設定される	データベースのマイグレーション時に <code>AUTOMATIC</code> に設定される	データベースの作成時に <code>AUTOMATIC</code> に設定される
<code>applheapsz</code>		X	X
<code>dbheap</code>		X	X
<code>instance_memory</code>	X		
<code>mon_heap_sz</code>	X		
<code>stat_heap_sz</code>		X	X
<code>stmtheap</code>			X

単純化されたメモリー構成の一部として、以下のエレメントは非推奨になりました。

- 構成パラメーター `appgroup_mem_sz`、`groupheap_ratio`、`app_ctl_heap_sz`、および `query_heap_sz`。これらの構成パラメーターは、新規の構成パラメーター `appl_memory` で置き換えられます。
- `db2mtrk` メモリー・トラッカー・コマンドの `-p` パラメーター。専用エージェント・メモリー・ヒープをリストするこのパラメーターは、すべてのアプリケーション・メモリー消費量をリストする `-a` パラメーターに置き換わりました。

#### 関連概念

28 ページの『メモリー構成が単純化された』

169 ページの『`db2mtrk` コマンドが変更された』

140 ページの『いくつかの構成パラメーターが変更された』

#### 関連資料

コマンド・リファレンスの『`db2mtrk` - メモリー・トラッカー・コマンド』

## Information Integrator 製品 ID 値の変更

バージョン 9.5 では、Information Integrator 製品の製品 ID 値が、バージョン 9.5 Information Integrator 製品の名前に合うように変更されました。

## 詳細

製品 ID 値とは、ライセンス管理ツール (db2licm ユーティリティ) が使用する入力パラメーターです。db2licm コマンドに **-i** オプションを指定して、製品 ID 値をリストできます。

バージョン 9.5 では、以下の Information Integrator 製品の製品 ID 値が変更されました。

表 19. 変更された製品 ID 値

製品名	バージョン 9.5 の製品 ID	バージョン 9.1 の製品 ID
WebSphere Data Event Publisher	wsep	wsiip
WebSphere Federation Server	wsfs	wsiif
WebSphere Replication Server	wsrs	wsiir

## 解決方法

db2licm コマンドの出力を解析するすべてのスクリプトおよびアプリケーションを更新します。

### 関連資料

コマンド・リファレンスの『db2licm - ライセンス管理ツール・コマンド』

---

## アプリケーション開発の変更のサマリー

### 変更された Java ルーチン用のデフォルト JDBC ドライバー

Java ストアド・プロシージャやユーザー定義関数などの Java ルーチン用のデフォルト・ドライバーが、IBM Data Server Driver for JDBC and SQLJ に変更されました。

## 詳細

バージョン 9.5 より前には、IBM DB2 Driver for JDBC and SQLJ (バージョン 9.5 では IBM Data Server Driver for JDBC and SQLJ という名前に変更) を Java ルーチン用に使用したい場合、**DB2\_USE\_DB2JCCT2\_JROUTINE** 環境変数を設定する必要がありました。今後は IBM Data Server Driver for JDBC and SQLJ がデフォルト・ドライバーになったため、この環境変数の定義はもはや必要なくなりました。

## 解決方法

推奨されなくなった Linux、UNIX、および Windows 用 DB2 JDBC Type 2 ドライバーを使って Java ルーチンの SQL 要求を処理したい場合には、**DB2\_USE\_DB2JCCT2\_JROUTINE** を OFF に設定してください。なお、将来のリリースでのサポート上の問題を未然に防ぐには、推奨されないこのドライバーを使用するアプリケーションを IBM Data Server Driver for JDBC and SQLJ にマイグレーションすることを検討してください。

### 関連概念

pureXML ガイドの『Java ルーチン用のドライバーの指定』

77 ページの『JDBC および SQLJ のサポートの機能拡張』

#### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『その他の変数』

## ResultSetMetaData が IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 に対して異なる値を返す

IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 では、`ResultSetMetaData.getColumnLabel` および `ResultSetMetaData.getColumnName` で返される値が JDBC 4.0 規格に準拠するよう変更されました。これらの値は、IBM Data Server Driver for JDBC and SQLJ バージョン 3.50 およびこれ以前の JDBC ドライバーに対して返される値とは異なります。

### 詳細

以下の要因に応じて結果は異なります。

- どのデータ・ソース・タイプおよびバージョンを使用しているか。DB2 for z/OS and OS/390 バージョン 7 および DB2 for i5/OS® V5R2 はこの変更に影響されません。これらのデータベース製品のそれ以降のバージョン、また DB2 Database for Linux, UNIX, and Windows および IBM Informix Dynamic Server のすべてのバージョンのみが影響を受けます。
- 照会の SELECT リスト内の列に AS 節があるかどうか。IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 より前の JDBC ドライバーでは、照会の SELECT リスト内の列に AS 節が含まれる場合、`ResultSetMetaData.getColumnName` によってその AS 節の引数が返されます。IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 では、`ResultSetMetaData.getColumnName` はその表列名を返します。
- 照会の SELECT リスト内の列に AS 節はないが、LABEL ステートメントからのラベルがあるかどうか。DB2 for z/OS および DB2 for System i は、ラベルを列に割り当てる LABEL ステートメントをサポートします。IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 より前の JDBC ドライバーでは、照会の SELECT リスト内の列にラベルがある場合、`ResultSetMetaData.getColumnName` はその表列名を返し、`ResultSetMetaData.getColumnLabel` はその LABEL ステートメントからの列ラベルを返します。IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 では、`ResultSetMetaData.getColumnName` および `ResultSetMetaData.getColumnLabel` はその表列名を返します。LABEL ステートメントからの列ラベルは使用されません。
- SELECT リスト内の列に、LABEL ステートメントからのラベルおよび AS 節があるかどうか。IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 より前の JDBC ドライバーでは、照会の SELECT リスト内の列にラベルおよび AS 節がある場合、`ResultSetMetaData.getColumnName` はその AS 節の引数を返し、`ResultSetMetaData.getColumnLabel` はその LABEL ステートメントからの列ラベルを返します。IBM Data Server Driver for JDBC and SQLJ バージョン 4.0 では、`ResultSetMetaData.getColumnName` はその表列名を返し、

`ResultSetMetaData.getColumnLabel` はその AS 節の引数を返します。LABEL ステートメントからの列ラベルは使用されません。

## 解決方法

新規 `ResultSetMetaData` の動作に準拠するようにアプリケーションを変更することはできないが、JDBC 4.0 の他のフィーチャーが必要である場合には、`useJDBC4ColumnNameAndLabelSemantics` Connection または `DataSource` プロパティを `DB2BaseDataSource.NO (2)` に設定して旧動作を維持します。

### 関連資料

Java アプリケーションの開発の『DB2 for z/OS および DB2 Database for Linux, UNIX, and Windows のための共通 IBM Data Server Driver for JDBC and SQLJ プロパティ』

## 自動的に生成されるキーでのバッチ更新が `SQLException` の原因となる

IBM Data Server Driver for JDBC and SQLJ バージョン 3.50 以降では、自動的に生成されるキーの検索のために SQL ステートメントを準備し、バッチ更新に `PreparedStatement` オブジェクトを使用すると `SQLException` が生じます。

### 詳細

IBM Data Server Driver for JDBC and SQLJ バージョン 3.50 以前のバージョンでは、自動的に生成されるキーを返すために準備された `PreparedStatement` オブジェクトにアプリケーションが `addBatch` または `executeBatch` メソッドを呼び出しても、`SQLException` はスローされません。しかし、`PreparedStatement` オブジェクトは自動的に生成されるキーを返しません。IBM Data Server Driver for JDBC and SQLJ バージョン 3.50 以降のバージョンでは `SQLException` がスローされ、アプリケーションが正しくない操作を試行していることを示します。

## 解決方法

アプリケーションを変更して、自動的に生成されるキーを返すように準備された `PreparedStatement` オブジェクトに対してバッチ更新を行わないようにします。

## unfenced ルーチン、unfenced ラッパー・ライブラリー、およびセキュリティー・プラグインはスレッド・セーフである必要がある (Linux および UNIX)

新規マルチスレッド・データベース・マネージャーでは、unfenced ルーチン (ストアード・プロシージャやユーザー定義関数など)、unfenced ラッパー・ライブラリー、およびセキュリティー・プラグインがスレッド・セーフであることが必要です。

### 詳細

バージョン 9.5 以前では、Linux および UNIX システム上で unfenced ルーチンを実行したエージェントは、それぞれ別個のプロセスでした。マルチスレッドのデータベース・マネージャーで、スレッド・セーフでない unfenced ストアード・プロシ

ージャー、unfenced ユーザー定義関数、unfenced ラッパー・ライブラリー、またセキュリティ・プラグインを実行すると、正しくない結果、データベース破壊、またはデータベース・マネージャーの異常終了が生じるおそれがあります。

サード・パーティーのクライアント・ライブラリー (Sybase や Teradata ラッパーなど) をロードする unfenced ラッパー・ライブラリーもスレッド・セーフでなければなりません。DB2 付属の unfenced ラッパー・ライブラリーは既にスレッド・セーフであるため、これは、ユーザー定義の unfenced ラッパー・ライブラリーに該当します。同様に、ユーザー定義のセキュリティ・プラグインもスレッド・セーフである必要があります。

スレッド・セーフとは、コードの特定のプロパティを指します。スレッド・セーフであるコードでは、同じプロセス内で複数のオペレーティング・システムのスレッドがみな同じコードを同時に実行している場合、各スレッドが正しい結果をもたらし、他のスレッドの操作に悪い影響は与えません。DB2 サーバーはマルチスレッドになったため、同じ unfenced ストアード・プロシージャのコードを複数のデータベース・エージェントで並行して実行することが可能です。スレッド・セーフティを保証するのは困難なタスクであり、コード検査によってのみ可能です。以下に、避ける必要のある、スレッド・セーフでないコードの一般的な例を挙げます。

- 例えばセマフォといった、何らかの同期の手段によって正しく保護されていないグローバル変数の使用。ルーチン・コード内のホスト変数は、グローバル変数の一例です。
- スレッド・セーフでないライブラリー関数、または影響が (呼び出し側のスレッドだけでなく) プロセス全体に及ぶライブラリー関数への呼び出し。この例としては、現行作業ディレクトリーを変更するライブラリー関数、またはプロセスのロケールを変更するライブラリー関数が挙げられます。
- シグナル・ハンドラーのインストール、またはシグナル・マスクへの変更。DB2 サーバーは独自のシグナル・ハンドラーをインストールします。DB2 サーバーの整合性を保証するためには、これらのシグナル・ハンドラーは変更してはなりません。
- 新規スレッドまたはプロセスの作成。

## 解決方法

コードのスレッド・セーフティについて疑いがある場合、またはソース・コードへのアクセスが不可能な場合には、ルーチンまたはラッパー・ライブラリーを FENCED および NOT THREADSAFE としてカタログしてください。注意深く検査した結果、これらがスレッド・セーフであると分かり、さらにこのコードを fenced モードで実行することがパフォーマンス上受け入れられない場合に限り、これらを unfenced で実行してください。

## ID の長さ制限の増加

ID のサポート拡大により、他の DBMS ベンダー製アプリケーションの移植がさらに容易になります。また、ID を短くする必要がないため、データ定義言語 (DDL) のマイグレーションがさらに容易になるはずで

最大長が長くなった ID は、以下の表にリストされています。

表 20. バージョン 9.1 および 9.5 での ID の長さの制限

ID 名	バージョン 9.1 での長さ (バイト)	バージョン 9.5 での長さ (バイト)
属性	18	128
許可 ID (Authid)	30	128
列	30	128
制約	18	128
カーソル	18	128
データベース・パーティション・グループ	18	128
イベント・モニター	18	128
グループ	30	128
パッケージ	8	128
スキーマ	30	128
特定の名前	18	128
SQL パス (FUNCPATH BIND オプションおよび CURRENT PATH 特殊レジスターによる指定)	254	2048
ステートメント	18	128
トリガー	18	128
ユーザー定義タイプ	18	128

現在でも SQLDA は、ユーザー定義タイプ (UDT) では 8 バイトのスキーマ名、UDT では 18 バイトの名前、さらに列では 30 バイトの名前に制限されているため、128 バイトの制限は非組み込み SQL のみに適用される点に注意してください。

128 バイトの制限は、データベース・マネージャーによりシステム・カタログに保管される制限を指します。アプリケーション内で ID を示すために使用されるコード・ページはさまざまである場合があるため、アプリケーション側の制限は定義されていません。アプリケーションのコード・ページに関係なく、アプリケーションおよびサーバー両方の側の DB2 ユーティリティーが 128 バイトの制限を使用します。

役立つサンプル・ファイル、checkv9limits が、samples/admin\_scripts にあります。このファイルを使用すると、データベースで、より大きいバージョン 9.5 の制限を使用している可能性がある ID を見つけることができます。

#### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『SQL および XML 制限』

## 列およびアプリケーション・バッファに、より大きいデフォルト値が必要

USER および SCHEMA 特殊レジスターは、128 バイトまでの長さの値を返すようになりました。このより長くなった ID のサポートのため、列およびアプリケーション・バッファのデフォルトの長さの値を 128 バイトに増加する必要があります。

### 詳細

WITH DEFAULT オプションを使用して表を作成または変更し、ユーザー特殊レジスター (CURRENT USER、SESSION\_USER、SYSTEM\_USER) または CURRENT SCHEMA を指定したときに、ターゲットの列が小さすぎる場合、以下の例に示すように警告が返されます。

```
SQL20114W Column "COL1" in table "TAB1" is not long enough
for the defined length of the USER default value. SQLSTATE=01642
```

### 解決方法

より長くなった ID の名前を使用するかどうかに関わらず、USER または SCHEMA 特殊レジスター値に保管される可能性のある列およびアプリケーション・バッファのデフォルトの長さの値を変更して、新しい最大長を受け入れるようにする必要があります。

#### 関連概念

69 ページの『ID の長さ制限の増加』

## 一部の CLI/ODBC アプリケーションは消費メモリーが多くなることがある

BlockLobs を 1 に設定し、LOB 値を直接バッファにバインドする CLI/ODBC クライアントは、旧リリースより多くのメモリーを消費することがあります。

CLI/ODBC アプリケーションが消費する追加のメモリーの量は、要求に応じて取得するデータの量によって異なります。CLI/ODBC クライアントは、

**MaxLOBBlockSize** 構成キーワードを指定して、1 つの要求に応じて戻される LOB データの量を制限できます。代わりに、接続属性

**SQL\_ATTR\_MAX\_LOB\_BLOCK\_SIZE** または DB2 レジストリー変数

**DB2\_MAX\_LOB\_BLOCK\_SIZE** を設定することもできます。

#### 関連概念

コール・レベル・インターフェース ガイドおよびリファレンス 第 2 巻の『db2cli.ini 初期設定ファイル』

#### 関連資料

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『その他の変数』

コール・レベル・インターフェース ガイドおよびリファレンス 第 2 巻の『ステートメント属性 (CLI) のリスト』

コール・レベル・インターフェース ガイドおよびリファレンス 第 2 巻の『BlockLobs CLI/ODBC 構成キーワード』

## db2Load および db2Import パラメーターが変更され、より長い ID をサポートするようになった

長くなった名前をサポートするため、新規入力アクション・ストリング **piLongActionString** が db2Load および db2Import API に追加されました。これは sqlchar データ構造ではなく、sqllob を使用します。

### 詳細

**piActionString** データ構造は推奨されなくなりました。将来のリリースでは除去される可能性があります。代わりに、新規データ構造 **piLongActionString** を使用してください。

### 解決方法

API は、データ構造の一方のみが初期化されているかどうかを検査します。両方を初期化した場合、データ構造は互いに排他的であることを示すメッセージ SQL3009N が返されます。

#### 関連資料

データ移動ユーティリティ ガイドおよびリファレンスの『db2Import API - 表、階層、ニックネーム、ビューへのデータのインポート』

データ移動ユーティリティ ガイドおよびリファレンスの『db2Load API - 表へのデータのロード』

## ID が長すぎる場合にエラーと警告がより早く返される

バージョン 9.5 では、ID に対して境界および長さの追加の検査が行われます。ID が制限を超えた場合、エラーまたは警告が生成されますが、これは、プリコンパイル、バインディング、またはアプリケーションの実行中に、以前のリリースの DB2 製品よりも早い時点で行われます。

### 詳細

例えば、128 バイトを超える長さの AUTHID に対する GRANT ステートメントを含む組み込み SQL ステートメントに対して、SQL0102N プリコンパイル・エラーが返されるようになりました。前のバージョンの DB2 製品では、AUTHID が 128 バイトを上回る場合、GRANT ステートメントの実行時にエラーが生成されていました。

### 解決方法

ID 名を訂正して、許容される長さにします。

#### 関連概念

69 ページの『ID の長さ制限の増加』



## ダウン・レベルのユーティリティーおよび API はより長くなった ID を正しく扱うことができない可能性がある

以前の DB2 バージョン付属のユーティリティーおよび API では、バージョン 9.5 のより長い ID を許容できない可能性があります。

### 詳細

ユーティリティーまたは API が長い ID 名を処理するとき、データの長形式がどのように扱われ、どのような結果になるかは、以下のとおりです。

- すべて正常に処理される。
- ユーティリティーまたは API が、その長いデータに関して警告またはエラー・メッセージを返す。
- ユーティリティーまたは API が警告またはエラー・メッセージを返して失敗する。
- メッセージなしに、長いデータが切り捨てられる。

### 解決方法

バージョン 9.5 データベースにラージ ID が含まれる場合、それらのデータベースへのアクセスには、バージョン 9.5 のクライアントおよびユーティリティーのみを使用します。ユーティリティーがラージ ID にアクセスする必要がある場合には、そのユーティリティーのバージョン 9.5 レベルのみを使用します。

#### 関連概念

69 ページの『ID の長さ制限の増加』

## 非修飾 SYSFUN 関数が SYSIBM エラー・メッセージを返すことがある

いくつかの SYSFUN 関数が、SYSIBM 関数として使用可能になりました。エラーの発生時に、SYSIBM バージョンは、SYSFUN バージョンとは異なる SQLCODES を返します。

### 詳細

以下の 8 つの SYSFUN 関数は、SYSIBM スキーマ内の組み込み関数として使用可能になりました。すなわち、LN (または LOG)

、LOG10、DEGREES、RADIANS、SIGN、SQRT、POWER、および EXP です。これらの 8 つの関数のうちのいずれかへの非修飾参照は、SYSIBM スキーマに解決され、その結果、期待されたものとは異なる SQLCODE が返されることがあります。

例えば、`values (sqrt(-1))` といった非修飾の関数呼び出しを行ってエラーが生じると、以下のような SQLCODE を受け取ることになります。

```
1-----  
SQL0802N Arithmetic overflow or other arithmetic exception occurred.  
SQLSTATE=22003
```

`values (sysfun.sqrt(-1))` のように、SYSFUN バージョンを明示的に呼び出す完全修飾関数呼び出しは、異なる種類の SQLCODE を返します。

```
1-----  
SQL0443N Routine "SYSFUN.SQRT" (specific name "SQRT") has returned an error SQLSTATE  
with diagnostic text "SYSFUN:01".  
SQLSTATE=38552
```

SQL0443N はユーザー定義関数に固有であり、この関数の SYSIBM バージョンは組み込み関数としてインプリメントされるようになったため、異なるエラー・コードが返されます。非修飾呼び出しのエラー・コードには、失敗した関数または障害のタイプに関する情報が含まれないことに注意してください。

## 解決方法

これらの関数の SYSFUN バージョンを呼び出すためには、常に完全修飾で関数呼び出しを行ってください。

## 特殊レジスターが長くなった

バージョン 9.5 では、CURRENT DEFAULT TRANSFORM GROUP および CURRENT PATH 特殊レジスターが長くなりました。

### 詳細

CURRENT DEFAULT TRANSFORM GROUP 特殊レジスターの長さが、18 バイトから 128 バイトに延長されました。CURRENT PATH 特殊レジスターの長さは、254 バイトから 2048 バイトに延長されました。これらの特殊レジスターの延長された長さが、その長さを収めることのできないアプリケーション・バッファーまたは列に割り当てられた場合、エラーが返されます。

### 解決方法

列またはアプリケーション・バッファーが、割り当てられた特殊レジスターの値を収めることができない場合には、列またはアプリケーション・バッファーの長さを延長します。

#### 関連概念

69 ページの『ID の長さ制限の増加』

---

## CLP およびシステム・コマンドの変更点のサマリー

### コマンド行プロセッサ (CLP) の出力が変更された

いくつかの CLP コマンドの出力が変更され、異なる情報を表示し、ラージ ID (128 バイト、SQL パスの場合は 2048 バイト) を表示できるようになりました。この出力の変更により、CLP コマンド出力の出力形式を解析したり、それに依存したりするアプリケーションに影響する場合があります。

### 詳細

コマンドに **SHOW DETAIL** オプションがあるのにそれを指定しなかった場合、ID フィールドが現在の長さに切り捨てられ、> 文字が名前の最後の部分に表示されます。これは、フィールドが切り捨てられたことを示す、CLP によって使用される規則です。**SHOW DETAIL** オプションを指定した場合は、名前がすべて表示されます。**SHOW DETAIL** オプションがない場合には、全長が表示されます。

以下のコマンド出力が次のように変更されました。

- **DESCRIBE**: このコマンドの出力は **SQLDA** フィールドを表示しなくなり、**sqlname** の代わりに **Column name** などの一般的な用語を使用します。さらに、コマンド・パラメーター **TABLE** は暗黙的な隠し列に関する情報を返すようになりました。コマンド・パラメーター **OUTPUT** は、記述される照会の **SELECT** リスト内で列を指定した場合に限り、暗黙的な隠し列に関する情報を返すようになりました。
- **GET DB CFG**: この出力は非推奨のデータベース構成パラメーターを表示しません。
- **GET SNAPSHOT: DB2** アプリケーション・メモリー・モデルが変更されたため、コマンド出力がこれまでとは少し異なります。

## 解決方法

この **CLP** コマンドの出力形式を解析し、この出力形式に依存するアプリケーションを更新することが必要になる場合があります。

### 関連概念

69 ページの『ID の長さ制限の増加』

140 ページの『いくつかの構成パラメーターが変更された』

## REDISTRIBUTE DATABASE PARTITION GROUP コマンドが変更された

バージョン 9.5 フィックスパック 1 からは、**REDISTRIBUTE DATABASE PARTITION GROUP** コマンド・イベント・ログ・ファイルが新しいディレクトリーに生成されるようになり、英語のみでなく、DB2 インスタンスで指定する言語で提供されるようになりました。

## 詳細

マイグレーション後に初めて **REDISTRIBUTE DATABASE PARTITION GROUP** コマンドを実行すると、メッセージ・ファイルを記録するための新しいディレクトリーが作成されます。このコマンドは、新しい形式を使用して新しいメッセージを新しいディレクトリー内に記録するとともに、以前の形式を使って以前のリリースのときと同じディレクトリーにもメッセージを記録します。

## 解決方法

新しいディレクトリーは次のとおりです。

- **Linux** および **UNIX** オペレーティング・システムでは `homeinst/sqllib/redistribute`
- **Windows** オペレーティング・システムでは `DB2INSTPROF¥instance¥redistribute`

### 関連概念

32 ページの『データ再配分の機能拡張により容量が増えるコストとロード・バランスのタスクが減っている』

### 関連資料

パーティションおよびクラスタリングのガイドの『**REDISTRIBUTE DATABASE PARTITION GROUP** コマンド』

## 複数のデータベース・パーティションを同時にバックアップするバックアップ操作

BACKUP DATABASE コマンド、db2Backup API、または BACKUP DATABASE パラメーターを指定した ADMIN\_CMD プロシージャを 1 回だけ呼び出すことにより、複数のデータベース・パーティションを一度にバックアップできるようになりました。この新しい機能をサポートするためのバックアップ・ユーティリティーの変更は、既存のアプリケーションまたはスクリプトに影響を与えることはありません。

### 詳細

バージョン 9.1 では、パーティション・データベースをバックアップするために、データベース・パーティションごとにバックアップ・ユーティリティーを呼び出す必要がありました。バージョン 9.5 では、カタログ・データベース・パーティションに対して単一のシステム視点 (SSV) バックアップを実行することにより、複数のデータベース・パーティションを一度にバックアップできます。カタログ・データベース・パーティションからバックアップ操作を実行するとき、**ON DBPARTITIONNUMS** パラメーターまたは **iAllNodeFlag** オプションを使用して、バックアップに含めるパーティションを指定できます。指定されたパーティションは同時にバックアップされ、すべての指定されたパーティションに関連付けられたバックアップ・タイム・スタンプは同じになります。

この機能の結果として、IBM Tivoli Storage Manager (TSM) 戻りコード 41 (マウント・ポイントの超過) および -51 (接続試行のタイムアウト) は、使用可能なセッションがもはや存在しない場合を除き、致命的エラーではなくなりました。

### 解決方法

TSM を使用している場合、アプリケーションまたはスクリプトを変更して、戻りコードの変更に対処してください。

#### 関連概念

データ・リカバリーと高可用性 ガイドおよびリファレンスの『バックアップの概要』

#### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの『バックアップの使用』

データ・リカバリーと高可用性 ガイドおよびリファレンスの『パーティション・データベースのバックアップ』

#### 関連資料

コマンド・リファレンスの『BACKUP DATABASE コマンド』

管理 API リファレンスの『db2Backup API - データベースまたは表スペースのバックアップ』

管理ルーチンおよびビューの『BACKUP DATABASE コマンド (ADMIN\_CMD プロシージャを使用)』

## db2audit コマンドが変更された

バージョン 9.5 で監査機能に新しいフィーチャーが追加されたため、db2audit コマンドが部分的に変更されました。

### 詳細

バージョン 9.5 の監査機能では、インスタンス・レベルおよび個々のデータベース・レベルの両方の監査が可能で、すべてのインスタンス・レベルのアクティビティおよびデータベース・レベルのアクティビティがそれぞれ個別のログに独立して記録されます。(SYSADM 権限を持つ) システム管理者は db2audit ツールを使用して、インスタンス・レベルの監査を構成し、その監査の情報を収集するタイミングを制御できます。システム管理者は db2audit ツールを使用して、インスタンス監査ログおよびデータベース監査ログをアーカイブし、いずれかの種類のアーカイブ・ログから監査データを抽出することもできます。

(SECADM 権限を持つ) セキュリティー管理者は、SQL ステートメント AUDIT で監査ポリシーを使用して、個々のデータベースの監査要件を構成および制御することができます。セキュリティー管理者は SYSPROC.AUDIT\_ARCHIVE および SYSPROC.AUDIT\_DELIM\_EXTRACT ストアード・プロシージャと SYSPROC.AUDIT\_LIST\_LOGS 表関数を使用して、監査ログをアーカイブし、適切なログを見つけ、区切り付きファイルにデータを抽出して分析することができます。

この新機能を提供するために、db2audit コマンドの以下のコマンド・パラメーターが変更されました。

- 整理 (**prune**) コマンド・パラメーターが除去されました。

以前のリリースでは、まず監査データを区切り文字付き ASCII ファイルに抽出した後、それを表にロードする必要がありました。その後、**prune** コマンド・パラメーターを使って db2audit を実行することにより、監査ログをクリーンアップできました。バージョン 9.5 では、監査ログを定期的に (例えば 1 日または 1 週間に一度) アーカイブして、必要なデータをアーカイブ・ファイルから抽出した後、アーカイブ・ファイルを削除するか、オフラインで保管することができます。

- 構成 (**configure**) コマンド・パラメーターの構文が変更されました。

バージョン 9.5 では、それぞれの監査カテゴリーの成功と失敗を指定できます。すべてのカテゴリーの成功または失敗を指定する必要はなくなりました。さらに、コマンドで指定したカテゴリーだけが変更されます。他のすべてのカテゴリーは不変です。以前のリリースでは、カテゴリーを指定しない場合、false (「監査しない」) に設定されました。

**configure** コマンド・パラメーターは、インスタンス・レベルの監査にのみ使用できます。データベース・レベルの監査には使用できません。データベース・レベルの監査については、セキュリティー管理者が監査ポリシーを使って監査を構成することができます。

- 抽出 (**extract**) コマンド・パラメーターの構文が変更されました。

各データベースの監査ログがそれぞれ個別のファイルに保管されるようになったため、**database** パラメーターがなくなりました。また、アーカイブされた監査ログ・ファイルの名前を提供する必要があります。

**extract** コマンド・パラメーターを使用しても、抽出操作が完了するまでインスタンスが停止することがなくなりました。これは、現行の `db2audit.log` ファイルではなくアーカイブされた監査ログ・ファイルを使用するようになったためです。しかし、**extract** コマンド・パラメーターを使用する前に監査ログをアーカイブする必要があります。また、以前のリリースほど頻繁に抽出を実行する必要もありません。バージョン 9.5 では、監査データを表示するときのみ、これを実行する必要があります。

**extract** コマンド・パラメーターでは、抽出対象のカテゴリを指定したり、成功/失敗イベントのどちらを (または両方を) 抽出するか指定したりできるようになりました。

- **describe** コマンド・パラメーターによって表示される項目が、**configure** コマンド・パラメーターの新しい構文をサポートできるように変更されました。

以前のリリースでは、有効範囲に示されたすべてのカテゴリに対して、状況を **SUCCESS** または **FAILURE** のいずれかにしか指定できませんでした。今後、各カテゴリに対して **SUCCESS**、**FAILURE**、**NONE**、または **BOTH** の値で状況を指定できるようになりました。以下の表は、以前のリリースのカテゴリ・イベント、ログ・エラー、およびログ成功の値 (標準出力で表示) をバージョン 9.5 の値 (標準出力で表示) にマップしたものです。

表 21. 標準出力で表示された以前のリリースの値から、バージョン 9.5 の値へのマッピング

カテゴリ・イベント、ログ・エラー、ログ成功 (以前のリリース)	カテゴリ・イベント (バージョン 9.5)
TRUE, FALSE, FALSE	NONE
TRUE, FALSE, TRUE	SUCCESS
TRUE, TRUE, FALSE	FAILURE
TRUE, TRUE, TRUE	BOTH
FALSE, いずれかの値、いずれかの値	NONE

- 開始 (**start**) および停止 (**stop**) コマンド・パラメーターは、インスタンスのレベルの監査だけを操作します。データベース・レベルの監査は対象とはなりません。

## 解決方法

新しい構文の `db2audit` コマンドを使用してください。

### 関連概念

データベース・セキュリティー・ガイドの『監査ポリシー』

データベース・セキュリティー・ガイドの『監査ログの保管と分析』

### 関連資料

コマンド・リファレンスの『`db2audit` - 監査機能管理者ツール・コマンド』

## db2ckmig コマンドが変更された

db2ckmig コマンドは新たに、データベースがリストア・ペンディング状態かどうか、および DB2 エンジン・ライブラリーに依存しない Linux および UNIX 作動プラットフォームに対する外部 unfenced ルーチンがデータベース内に存在するかどうかを検査するようになりました。

### 詳細

db2ckmig コマンドを使用して、データベースがマイグレーション可能であるかどうかを確認できます。データベースがリストア・ペンディング状態にある場合、このコマンドは失敗するようになりました。このコマンドの使用法および出力についての全体の説明は、「マイグレーション・ガイド」の『データベースがマイグレーションできる状態にあることの確認』を参照してください。

DB2 エンジン・ライブラリーに依存しない Linux および UNIX オペレーティング・システムに対する外部 unfenced ルーチンがデータベース内に存在する場合、このコマンドは警告メッセージ SQL1349W を返し、データベースのマイグレーション時に FENCED および NOT THREADSAFE として再定義されるすべての外部 unfenced ルーチンをリストしたファイルを生成します。

### 解決方法

データベースのリストア・ペンディング状態を除去するには、データベース・リストア操作を実行する必要があります。

警告メッセージ SQL1349W を受け取った場合、外部ルーチンを FENCED および NOT THREADSAFE として実行できるならば、データベースのマイグレーションを進めることができます。マイグレーション対象のデータベースで外部ルーチンを NOT FENCED および THREADSAFE として実行する必要がある場合、データベース・マイグレーション前にこれらが NOT FENCED および THREADSAFE として正常に実行されることを確認する必要があります。この確認を行う方法に関する情報は、「マイグレーション・ガイド」にある『32 ビット外部ルーチンを 64 ビット・インスタンスで実行するようにマイグレーションする』を参照してください。

#### 関連タスク

データ・リカバリーと高可用性 ガイドおよびリファレンスの『リストアの使用』

「マイグレーション・ガイド」の『ルーチンのマイグレーション』

## db2mtrk コマンドが変更された

メモリーの状況を詳細に報告する db2mtrk コマンドが変更されました。(専用エージェント・メモリー・ヒープをリストする) **-p** オプションは推奨されなくなり、(すべてのアプリケーション・メモリー消費量をリストする) **-a** オプションに置き換わりました。

### 詳細

DB2 アプリケーション・メモリー・モデルが変更されたため、コマンド出力がこれまでとは異なり、より多くの情報を示すようになりました。

## 解決方法

db2mtrk コマンドの出力を解析するスクリプトが存在する場合、新しい形式を扱えるよう解析条件を修正します。

### 関連資料

コマンド・リファレンスの『db2mtrk - メモリー・トラッカー・コマンド』

## カスタマイズされたコールアウト・スクリプトが検索対象になる (Linux および UNIX)

データベース・マネージャーは、db2cos、db2cos\_datacorruption、db2cos\_hang、および db2cos\_trap スクリプトのカスタマイズされたバージョンの有無を検査するようになりました。何も存在しない場合、デフォルトのバージョンが使用されます。

### 詳細

db2cos、db2cos\_datacorruption、db2cos\_hang、および db2cos\_trap スクリプトは、トラップ、ハング、またはデータ破壊障害が生じた場合に、トラブルシューティング情報収集のために実行されます。

Linux および UNIX オペレーティング・システムでは、データベース・マネージャーはまず、INSTHOME/sqllib/adm/ (INSTHOME はインスタンスのホーム・ディレクトリー) でコールアウト・スクリプトのカスタマイズされたバージョンが存在するかどうかを検査し、それらのスクリプトを実行するようになりました。スクリプトが見つからない場合、データベース・マネージャーは INSTHOME/sqllib/bin/ ディレクトリーのシステム提供のスクリプトを実行します。

## 解決方法

システム提供の設定を変更したり、システム提供のスクリプトに変更を加えたりしないでください。

### 関連概念

問題判別ガイドの『db2cos (コールアウト・スクリプト) 出力ファイル』

## OS プロセスおよびスレッドのリストが変更された (Linux および UNIX)

バージョン 9.5 ではマルチスレッド方式に移行したため、ps コマンドの出力が変更されました。また、db2pd コマンドに、データベース・パーティションでのエンジン・ディスパッチ可能単位 (EDU) をすべてリストする新規オプション **-edus** ができました。

### 詳細

UNIX および Linux オペレーティング・システム上のバージョン 9.5 では、DB2 インスタンス内のほとんどすべてのオペレーティング・システム・プロセスは、オペレーティング・システム・スレッドで、すべて、そのインスタンスの単一のプロセス内にあります。これにより、DB2 オペレーティング・システム・プロセス数が削減され、システムでの問題のトラブルシューティングを容易に行えるようになります。



ps コマンドを **-fu** *instancename* オプションで呼び出す場合には、以下の例に示すように、出力には 2 つの DB2 プロセス (db2sysc および db2acd) だけがリスト表示されます。

## 解決方法

db2sysc プロセスに関連した個々のスレッドを表示するには、ps コマンドで適切なスレッド・オプションを使用する必要があります。例えば Linux オペレーティング・システムでは、**-lfp** オプションを使用できます。AIX オペレーティング・システムでは、**-m -o THREAD** オプションを使用します。

## 例

ps **-fu** コマンドは以下の例に示すように、2 つのプロセスのみを示すようになりました。

```
$ ps -fu lpham
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
lpham	25996	25946	0	12:19	pts/12	00:00:00	-ksh
lpham	26567	26552	0	12:19	pts/12	00:00:00	ksh
lpham	<b>27688</b>	27676	0	12:21	pts/12	00:01:46	db2sysc
lpham	27716	27676	0	12:21	pts/12	00:00:00	db2acd
lpham	27995	27994	0	12:24	pts/13	00:00:00	-ksh
lpham	29321	26567	0	12:30	pts/12	00:00:00	ps -fu lpham

プロセス ID 27688 の詳細を表示するには、ps コマンドを新規の **-lfp** オプションで呼び出します。以下に例を示します。

```
$ps -lfp 27688
```

```
(try ps -m -o THREAD -p 27688 on AIX)
```

F	S	UID	PID	PPID	LWP	C	NLWP	PRI	NI	ADDR	SZ	WCHAN	STIME	TTY	TIME	CMD
5	S	lpham	27688	27676	27688	0	21	76	0	-	264903	msgrcv	12:21	pts/12	00:00:01	db2sysc
1	S	lpham	27688	27676	27694	0	21	75	0	-	264903	schedu	12:21	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	27695	0	21	76	0	-	264903	semtim	12:21	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	27696	0	21	79	0	-	264903	schedu	12:21	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	27697	0	21	76	0	-	264903	msgrcv	12:21	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	27714	0	21	76	0	-	264903	schedu	12:21	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	27827	1	21	75	0	-	264903	semtim	12:21	pts/12	00:00:06	db2sysc
1	S	lpham	27688	27676	27943	27	21	77	0	-	264903	schedu	12:22	pts/12	00:01:39	db2sysc
1	S	lpham	27688	27676	28150	0	21	75	0	-	264903	schedu	12:25	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	28153	0	21	76	0	-	264903	schedu	12:25	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	28156	0	21	75	0	-	264903	schedu	12:25	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30290	0	21	76	0	-	264903	schedu	12:36	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30291	0	21	75	0	-	264903	schedu	12:36	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30292	0	21	76	0	-	264903	semtim	12:36	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30293	0	21	76	0	-	264903	schedu	12:36	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30295	0	21	77	0	-	264903	semtim	12:36	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30296	0	21	77	0	-	264903	semtim	12:36	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30297	0	21	77	0	-	264903	semtim	12:36	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30298	0	21	76	0	-	264903	msgrcv	12:36	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30299	0	21	76	0	-	264903	msgrcv	12:36	pts/12	00:00:00	db2sysc
1	S	lpham	27688	27676	30300	0	21	76	0	-	264903	msgrcv	12:36	pts/12	00:00:00	db2sysc

以下の例では、**-edus** オプションによって提供される情報を示します。

```
$ db2pd -edus
```

```
>>>> List of all EDUs for database partition 0 <<<<
```

```
db2sysc PID: 27688
db2wdog PID: 27676
db2acd PID: 27716
```

EDU ID	TID	Kernel TID	EDU Name
60	183282690400	30300	db2pfchr (TESTDB)
59	183278496096	30299	db2pfchr (TESTDB)
58	183291079008	30298	db2pfchr (TESTDB)

57	183295273312	30297	db2pclnr (TESTDB)
56	183286884704	30296	db2pclnr (TESTDB)
55	183299467616	30295	db2pclnr (TESTDB)
54	183307856224	30293	db2dlock (TESTDB)
53	183320439136	30292	db2lfr (TESTDB)
52	183303661920	30291	db2loggw (TESTDB)
51	183316244832	30290	db2loggr (TESTDB)
50	183257524576	28156	db2evmli (DB2DETAILDEADLOCK)
49	183261718880	28153	db2taskd (TESTDB)
46	183274301792	28150	db2w1md (TESTDB)
26	183312050528	27943	db2stmm (TESTDB)
17	183324633440	27827	db2agent (TESTDB)
16	183328827744	27714	db2resync
15	183333022048	27697	db2ipccm
14	183337216352	27696	db2lcc
13	183341410656	27695	db2thcln
12	183345604960	27694	db2alarm
1	183085558112	27688	db2sysc

### 関連概念

27 ページの『単純化されたマルチスレッド化アーキテクチャーにより総所有コスト (TCO) が減っている』

### 関連資料

コマンド・リファレンスの『db2pd - DB2 データベースのモニターおよびトラブルシューティング・コマンド』

## 第 17 章 推奨されない機能

推奨されない機能とは、このリリースでサポートされているものの、将来のリリースでは除去される可能性のある機能です。例えば、あるレジストリー変数によって起動される動作がこのリリースではデフォルトで有効になったため、その廃止されたレジストリー変数は将来のリリースで除去される場合があります。そのようなレジストリー変数はこのリリースでは推奨されません。

このセクションでは、バージョン 9.5 で推奨されなくなった機能について詳しく説明し、将来の変更に備えて計画を行います。

### レジストリー変数および環境変数のいくつかは推奨されない

複数のレジストリー変数および環境変数が、バージョン 9.5 で推奨されなくなりました。現在も、これらの変数は使用できますが、今後の製品のバージョンで除去される予定のため、使用しないことをお奨めします。

以下の表では、推奨されなくなったレジストリー変数および環境変数がリストされています。これらは別のフィーチャーに置き換えられたか、これらがサポートしていた機能が廃止されました。

表 22. バージョン 9.5 で推奨されなくなったレジストリー変数および環境変数

レジストリー変数または環境変数	詳細
<b>DB2_ALLOCATION_SIZE</b>	この変数は、推奨されておらず、今後のリリースでは除去される可能性があります。
<b>DB2ATLD_PORTS</b>	この変数は、推奨されておらず、今後のリリースでは除去される可能性があります。
<b>DB2_ASYNC_IO_MAXFILOP</b>	この変数は、スレッド化されたデータベース・マネージャーによって保守される共用ファイル・ハンドル表のゆえに、廃止されました。バージョン 9.5 でこれを設定することはできますが、効果はありません。
<b>DB2_BAR_AUTONOMIC_DISABLE</b>	この変数が必要なのは IBM の内部使用の場合だけです (テストなど)。
<b>DB2BPVARS</b>	この変数は、推奨されておらず、今後のリリースでは除去される可能性があります。
<b>DB2COUNTRY</b>	この変数は、 <b>DB2TERRITORY</b> レジストリー変数に置き換えられました。 <b>DB2TERRITORY</b> によって、クライアント・アプリケーションの地域またはテリトリー・コード (日付および時刻の形式に影響を与える) を指定できるようになります。 <b>DB2TERRITORY</b> は <b>DB2COUNTRY</b> と同じ値を受け入れます。例えば、 <b>DB2COUNTRY</b> を 68 に設定するのは、 <b>DB2TERRITORY</b> を 68 に設定するのと同じです。

表 22. バージョン 9.5 で推奨されなくなったレジストリー変数および環境変数 (続き)

レジストリー変数または環境変数	詳細
<b>DB2DEFPREP</b>	IBM サービスから推奨されたときだけ、この変数を使用します。
<b>DB2DMNBCKCTRL</b>	アクティブ・ディレクトリーのバックアップ・ドメイン・コントローラーは Windows NT プラットフォーム上にのみ存在し、Windows 2003 および Windows XP プラットフォーム上には存在しないので、この変数は必要なくなりました。バージョン 9.5 は Windows NT プラットフォーム上では稼働しません。
<b>DB2_ENABLE_SINGLE_NIS_GROUP</b>	Network Information Services (NIS および NIS+) のサポートが今後のリリースで除去される予定なので、この変数は推奨されていません。詳しくは、Network Information Services サポートは推奨されない (Linux および UNIX) を参照してください。
<b>DB2FFDC</b>	この変数は、 <b>DB2FODC</b> レジストリー変数に置き換えられました。 <b>DB2FODC</b> の DUMPCORE オプションを使用するなら <b>DB2FFDC</b> が提供していたのと同じ機能が得られます。デフォルトでは、DUMPCORE オプションは、コア・ファイルの生成ができるように、また以前のリリースとの互換性を保守するために ON に設定されています。
<b>DB2_HASH_JOIN</b>	この変数は、DB2 フィーチャーのコントロールを提供するために作成されましたが、このレジストリー・コントロールが必要なくなったため、不要です。
<b>DB2_INDEX_FREE</b>	この変数は CREATE INDEX ステートメント内の PCTFREE 節と同じ機能を持っています。PCTFREE 節は、索引作成時に各「索引」ページの何パーセントをフリー・スペースとして残すかを指定します。例えば、 <b>DB2_INDEX_FREE</b> を 20 に設定することは、CREATE INDEX <i>IndexName</i> ON <i>TableName</i> ( <i>Columns</i> ) PCTFREE 20 とするのと同じです。PCTFREE の値は、索引作成時または索引再ビルド時にのみ効果があり、索引の存続期間中同じ値を維持します。PCTFREE 節は、すべての索引に影響を与える <b>DB2_INDEX_FREE</b> とは異なり、作成中の索引にのみ影響を与えます。

表 22. バージョン 9.5 で推奨されなくなったレジストリー変数および環境変数 (続き)

レジストリー変数または環境変数	詳細
<b>DB2_MAP_XML_AS_CLOB_FOR_DLC</b>	XML 値にアクセスする大多数の既存の DB2 アプリケーションは XML を使用できるクライアント (バージョン 9.1 以降) を使ってそれを行うので、この変数は推奨されません。表データを総括的に取り出し、UTF-8 XML データを BLOB に解析できなかった以前のアプリケーションに対してのみこの変数は必要です。
<b>DB2MEMMAXFREE</b>	現在データベース・マネージャーはスレッド化エンジン・モデルを使用するので、この変数は今後は必要なくなります。詳しい情報については、マルチスレッド化アーキテクチャーにより総所有コストが減っているを参照してください。  注: この変数を設定しないでください。設定してしまうと、パフォーマンスが下がり、予期しない動作の原因となることがあります。
<b>DB2_NO_FORK_CHECK</b>	バージョン 9.5 では、現行処理 ID (PID) の取得手順が改善されたので、この変数は今後必要ありません。
<b>DB2NTNOCACHE</b>	この変数は、DB2 UDB バージョン 8.2 以降推奨されていません。このレジストリー変数が行うように設計されたすべての操作は、CREATE TABLESPACE および ALTER TABLESPACE SQL ステートメントを使用することにより、実行できます。
<b>DB2_PARTITIONEDLOAD_DEFAULT</b>	同じ動作を実行するのに使用できるさまざまなオプションが LOAD コマンドにあるため、この変数は推奨されません。
<b>DB2PRIORITIES、DB2NTPRICLASS</b>	これらの変数は推奨されません。DB2 サービス・クラスを使用してエージェント優先順位およびプリフェッチ優先順位を調整してください。
<b>DB2ROUTINE_DEBUG</b>	このストアード・プロシージャ・デバッガーは Unified Debugger に置き換えられたため、この変数は今後必要ありません。
<b>DB2_RR_TO_RS</b>	IBM サービスから推奨されたときだけ、この変数を使用します。
<b>DB2_SNAPSHOT_NOAUTH</b>	SYSMON 権限グループを使用して同じ機能性を実行できるので、この変数は不要です。
<b>DB2_TRUSTED_BINDIN</b>	この変数は有用でなくなったので推奨されていません。
<b>DB2_UPDATE_PART_KEY</b>	この変数は、推奨されておらず、今後のリリースでは除去される可能性があります。パーティション・キー更新はデフォルトで許可されているので、これは廃止されます。

表 22. バージョン 9.5 で推奨されなくなったレジストリー変数および環境変数 (続き)

レジストリー変数または環境変数	詳細
<b>DB2_VENDOR_INI</b>	この変数が含まれる環境変数設定を <b>DB2_DJ_INI</b> 変数によって指定されているファイルに置けるので、この変数は必要なくなりました。
<b>DB2YIELD</b>	この変数は Windows 3.1 でしか使用されていませんでしたが、Windows 3.1 はバージョン 9.5 ではサポートされていません。

### 関連概念

147 ページの『いくつかのレジストリー変数および環境変数の変更』

187 ページの『いくつかのレジストリー変数および環境変数の廃止』

---

## 推奨されなくなった GET AUTHORIZATIONS コマンド

バージョン 9.5 では GET AUTHORIZATIONS コマンドが推奨されなくなりました。代わりに、AUTH\_LIST\_AUTHORITIES\_FOR\_AUTHID 表関数を使用してください。

### 詳細

GET AUTHORIZATIONS コマンドは、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・ビュー (SYSCAT.DBAUTH) から検索した値を使用して、現在のユーザーの権限について報告します。DB2 許可モデルの内部的な変更のために、このコマンドはバージョン 9.5 では推奨されなくなりました。

このコマンドは、ロールがユーザーとグループのどちらに付与されているかに基づき、ロールを介して直接的または間接的に保持されている権限を報告します。

### 解決方法

特定のユーザーの権限を返すには、AUTH\_LIST\_AUTHORITIES\_FOR\_AUTHID 表関数を使用します。

#### 関連資料

管理ルーチンおよびビューの『AUTH\_LIST\_AUTHORITIES\_FOR\_AUTHID 表関数』

---

## 推奨されなくなった sqludau API

バージョン 9.5 では sqludau API が推奨されなくなりました。代わりに、AUTH\_LIST\_AUTHORITIES\_FOR\_AUTHID 表関数を使用してください。

### 詳細

DB2 許可モデルの内部的な変更のために、sqludau API は推奨されなくなりました。この API は現在のユーザーの権限を返します。このコマンドは、ロールが誰に付与されているかに基づき、ロールを介して直接的または間接的に保持されている権限を報告します。

## 解決方法

squadau によって提供されるのと同じ情報を得るには、`AUTH_LIST_AUTHORITIES_FOR_AUTHID` 表関数を使用します。

### 関連資料

管理ルーチンおよびビューの『`AUTH_LIST_AUTHORITIES_FOR_AUTHID` 表関数』

---

## 推奨されなくなっていくつかのモニター・エレメント

モニター・エレメントのサブセットが、バージョン 9.5 の機能の変更を反映して推奨されなくなりました。

### 詳細

以下のモニター・エレメントは、もはや勧められておらず、将来のリリースで除去される可能性があります。

- `agents_waiting_top` - エージェント最大待機数
- `agents_waiting_on_token` - トークン待ちエージェント
- `authority_lvl` - ユーザー許可レベル
- `cat_cache_size_top` - カタログ・キャッシュ最高水準点
- `db_heap_top` - 割り振られた最大データベース・ヒープ
- `max_agents_overflows` - 最大エージェント・オーバーフロー回数
- `pkg_cache_size_top` - パッケージ・キャッシュの最高水準点
- `priv_workspace_num_overflows` - 専用ワークスペースのオーバーフロー回数
- `priv_workspace_section_inserts` - 専用ワークスペース・セクション挿入
- `priv_workspace_section_lookups` - 専用ワークスペース・セクション検索
- `priv_workspace_size_top` - 専用ワークスペースの最大サイズ
- `shr_workspace_num_overflows` - 共有ワークスペースのオーバーフロー回数
- `shr_workspace_section_inserts` - 共有ワークスペース・セクション挿入数
- `shr_workspace_section_lookups` - 共有ワークスペース・セクション検索
- `shr_workspace_size_top` - 最大共有ワークスペース・サイズ

## 解決方法

推奨されないエレメントはスナップショット出力から除去されて、SNAPDBM 管理ビューおよび `SNAP_GET_DBM` 表関数に定義されている可能性があります。エレメントが見つかった場合、その値は有効ではありません。db2GetSnapshot API は、これらの推奨されないエレメントに対する `SQLM_DBMON_VERSION6` 以降の `iVersion` (収集するデータベース・モニター・データのバージョン ID) を使用した要求には値を返しません、`SQLM_DBMON_VERSION5_2` 以前についてはゼロの値を返します。使用可能なときには、代替エレメント・モニターを使用してください。

推奨されないモニター・エレメント	変更
<ul style="list-style-type: none"> <li>• <b>agents_waiting_top</b> - エージェント最大待機数</li> <li>• <b>agents_waiting_on_token</b> - トークン待ちエージェント</li> <li>• <b>max_agents_overflows</b> - 最大エージェント・オーバーフロー回数</li> </ul>	<p>プロセス・モデル・パラメーターの構成メカニズムがバージョン 9.5 で単純化されたので、これらのモニター・エレメントは必要なくなりました。それらを使用しても、エラーは生成されません。ただし、有効な値は返されません。</p>
<p><b>authority_lvl</b> - ユーザー許可レベル</p>	<p>代わりに <b>authority_bitmap</b> モニター・エレメントを使用してください。これは、ユーザーおよびユーザーの属するグループに対して与えられた権限と特権を示します。これらの権限と特権には、ユーザーおよびユーザーの属するグループに対して与えられたロールが持っている権限や特権が含まれます。</p>
<p><b>cat_cache_size_top</b> - カタログ・キャッシュ最高水準点</p>	<p>このモニター・エレメントを使用しても、エラーは生成されません。ただし、有効な値は返されません。メモリー・プール・エレメント <b>pool_watermark</b> および <b>pool_id</b> エレメントを <b>SQLM_HEAP_CAT_CACHE</b> の値と共に使用します。</p>
<p><b>db_heap_top</b> - 割り振られた最大データベース・ヒープ</p>	<p>このモニター・エレメントを使用しても、エラーは生成されません。ただし、有効な値は返されません。メモリー・プール・エレメント <b>pool_watermark</b> および <b>pool_id</b> エレメントを <b>SQLM_HEAP_DATABASE</b> の値と共に使用します。</p>
<p><b>pkg_cache_size_top</b> - パッケージ・キャッシュの最高水準点</p>	<p>このモニター・エレメントを使用しても、エラーは生成されません。ただし、有効な値は返されません。メモリー・プール・エレメント <b>pool_watermark</b> および <b>pool_id</b> エレメントを <b>SQLM_HEAP_PACKAGE_CACHE</b> の値と共に使用します。</p>



推奨されないモニター・エレメント	変更
<ul style="list-style-type: none"> <li>• <b>priv_workspace_num_overflows</b> - 専用ワークスペースのオーバーフロー回数</li> <li>• <b>priv_workspace_section_inserts</b> - 専用ワークスペース・セクション挿入</li> <li>• <b>priv_workspace_section_lookups</b> - 専用ワークスペース・セクション検索</li> <li>• <b>priv_workspace_size_top</b> - 専用ワークスペースの最大サイズ</li> <li>• <b>shr_workspace_num_overflows</b> - 共有ワークスペースのオーバーフロー回数</li> <li>• <b>shr_workspace_section_inserts</b> - 共有ワークスペース・セクション挿入数</li> <li>• <b>shr_workspace_section_lookups</b> - 共有ワークスペース・セクション検索</li> <li>• <b>shr_workspace_size_top</b> - 最大共有ワークスペース・サイズ</li> </ul>	<p>メモリー構成がバージョン 9.5 で単純化されたので、これらのモニター・エレメントは必要なくなりました。それらを使用しても、エラーは生成されません。ただし、有効な値は返されません。</p>

#### 関連概念

27 ページの『単純化されたマルチスレッド化アーキテクチャーにより総所有コスト (TCO) が減っている』

#### 関連資料

システム・モニター ガイドおよびリファレンスの『pool\_id - メモリー・プール ID : モニター・エレメント』

システム・モニター ガイドおよびリファレンスの『pool\_watermark - メモリー・プール水準点 : モニター・エレメント』

システム・モニター ガイドおよびリファレンスの『authority\_bitmap - ユーザー許可レベル : モニター・エレメント』

## ログ制御ファイル SQLOGCTL.LFH の名前変更とコピー

バージョン 9.1 では、データベース・マネージャーは SQLOGCTL.LFH という 1 つのログ制御ファイルを維持していました。バージョン 9.5 では、データベース・マネージャーは 2 つのログ制御ファイルのコピー (SQLOGCTL.LFH.1 と SQLOGCTL.LFH.2) を維持します。

### 詳細

障害後にデータベースが再始動するとき、データベース・マネージャーはログ・ファイルに保管されたトランザクション情報を適用して、データベースを整合性のある状態に戻します。ログ・ファイル内のどの項目を適用する必要があるか判別するために、データベース・マネージャーはログ制御ファイルを使用します。

ログ制御ファイルが壊れている場合、データベース・マネージャーはデータベースを整合性のある状態に戻すことができない可能性があります。しかし、ログ制御ファイルのコピーが 2 つあれば、ログ制御ファイルの 1 つのコピーが壊れた場合に、データベース・マネージャーは再始動中にもう一方のコピーを使用できるので、データベースの回復力を増強できます。

## 解決方法

ログ制御ファイルを参照するアプリケーションまたはスクリプトを実行する場合、ログ制御ファイルのいずれかの複製を参照するようにアプリケーションやスクリプトを更新してください。db2flsn のようなツールでは、**-path** パラメーターを使用して、両方のログ制御ファイルのパスを指定してください。こうすれば、データベース・マネージャーは、1 つのログ制御ファイルが欠落、損傷、または不整合状態にある場合でも対処できます。

## 例

db2flsn コマンドで **-file** パラメーターを使用する場合、SQLOGCTL.LFH.1 または SQLOGCTL.LFH.2 のいずれか 1 つのログ制御ファイルの名前をコマンドに渡します。詳しくは、「コマンド・リファレンス」の『db2flsn - ログ・シーケンス番号の検出』のトピックを参照してください。

### 関連概念

データ・リカバリーと高可用性 ガイドおよびリファレンスの『ログ制御ファイル』

### 関連資料

コマンド・リファレンスの『db2flsn - ログ・シーケンス番号の検出コマンド』

---

## 推奨されなくなった IMPORT コマンド・オプション CREATE および REPLACE\_CREATE

IMPORT コマンドの **CREATE** および **REPLACE\_CREATE** オプションは推奨されなくなりました。将来のリリースでは除去される可能性があります。

## 詳細

**CREATE** および **REPLACE\_CREATE** オプションを使用すれば、表を作成してそれにデータを挿入することができます。ただし、**CREATE** および **REPLACE\_CREATE** の使用時に表のすべてのプロパティが再作成されるとは限らないため、これらのオプションは推奨されなくなりました。

## 解決方法

**CREATE** および **REPLACE\_CREATE** オプションを使用する代わりに、db2look コマンドを 2 段階の処理で使用してください。まず、db2look を使用して、元の表定義をキャプチャーして表を再作成します。表を再作成した後、LOAD または IMPORT コマンドを発行して表にデータを追加します。表のすべてのプロパティが db2look コマンドによって保持されます。その後、別個の IMPORT または LOAD 操作を実行すれば、より良い方法で表を再作成できます。

### 関連タスク

データ・サーバー、データベース、およびデータベース・オブジェクトのガイドの『既存の表の類似表の作成』

#### 関連資料

データ移動ユーティリティ ガイドおよびリファレンスの『db2look - DB2 統計および DDL 抽出ツール・コマンド』

---

## 推奨されなくなった XML Extender

バージョン 9.5 から、XML Extender が提供した機能は、pureXML フィーチャーで置き換えられました。その結果、XML Extender は推奨されなくなりました。

### 詳細

DB2 バージョン 9.1 での pureXML フィーチャーの導入により、XML Extender は推奨されなくなりました。

pureXML フィーチャーは、XML データ・タイプを持つデータベース表列に整形 XML 文書を保管することを可能にします。XML データを XML 列に保管することにより、データをテキストとして保管したり異なるデータ・モデルにマップしたりするのではなく、ネイティブの階層形式のまま保持することができます。XMLQUERY や XSLTRANSFORM といったデータベース関数は、XML データ・タイプを持つデータベース表に直接適用できます。データベースに XML ツールのセットが十分提供されているため、XML Extender 機能は必要でなくなりました。

---

## 推奨されなくなった静的データ・ストリームのスナップショット出力

バージョン 5.2 (またはそれ以前) の入力バージョンがスナップショット・モニター API に指定された場合、スナップショット出力は静的なサイズの構造 (sqlmon.h ファイルにその記述がある) で渡して返します。スナップショット出力形式は推奨されなくなりました。将来のリリースでは除去される可能性があります。

### 詳細

以下の入力バージョンは推奨されなくなりました。将来のリリースではこれらのサポートが除去される可能性があります。

- SQLM\_DBMON\_VERSION1
- SQLM\_DBMON\_VERSION2
- SQLM\_DBMON\_VERSION5
- SQLM\_DBMON\_VERSION5\_2

バージョン 6 以降のスナップショット・モニターは、静的構造ではなく、セルフ記述型データ・ストリームを使用します。

### 解決方法

推奨されなくなった入力バージョンを使用するモニター・アプリケーションがあれば、新しいバージョンを使用するよう変更するとともに、セルフ記述型のスナップショット・モニター形式を使用するように変更します。セルフ記述型モニター形式を使用するモニター・アプリケーションの例は、dbsnap.c (C サンプル) または

dbsnap.C (C++ サンプル) を参照してください。

---

## 推奨されなくなった Web Object Runtime Framework (WORF)

Web Object Runtime Framework (WORF) は推奨されなくなりました。将来のリリースでは除去される可能性があります。このリリースから始まり、データ・サーバー開発者ツールによって、Web サービスの迅速な開発およびデプロイのための、よりシンプルでより直感的な開発環境を提供します。

### 詳細

Web Objects Runtime Framework (WORF) は、DADX 文書を Web サービスとして作成し呼び出すためのツールおよびランタイム・サポートを提供します。

WORF は、データ・サーバー開発者ツール内の新規フィーチャーに置き換わりました。このフィーチャーでは、文書アクセス定義拡張 (DADX) ファイルを作成せずに Web サービスを作成することが可能です。さらに、このデータ・サーバー開発者ツールを使用して、Web サービスの操作の基本となる SQL ステートメントおよびストアド・プロシージャを作成できます。そして最後に、多くのシナリオで Web サービスのデプロイが、一回のマウス・クリックで行えるようになりました。

このフィーチャーに関する詳細情報は、以下をお読みください。Web サービスの開発とデプロイ

### 解決方法

WORF Web サービスをデータ・サーバー開発者ツールの Web サービスにマイグレーションしてください。マイグレーションの指示は、以下にあります。Web Object Runtime Framework (WORF) 用に開発された Web アプリケーションのマイグレーション

---

## 推奨されなくなった db2Import および db2Load API の piActionString データ構造

db2Import および db2Load API の **piActionString** データ構造は推奨されなくなりました。将来のリリースでは除去される可能性があります。

### 詳細

長くなった名前をサポートするため、新規入力アクション・ストリング **piLongActionString** が db2Load および db2Import API に追加されました。これは sqlchar データ構造ではなく、sqllob を使用します。

### 解決方法

API は、データ構造の一方のみが初期化されているかどうかを検査します。両方を初期化した場合、データ構造は互いに排他的であることを示すメッセージ SQL3009N が返されます。

#### 関連資料

データ移動ユーティリティー ガイドおよびリファレンスの『db2Import API - 表、階層、ニックネーム、ビューへのデータのインポート』

データ移動ユーティリティー ガイドおよびリファレンスの『db2Load API - 表へのデータのロード』

---

## Network Information Services サポートは推奨されない (Linux および UNIX)

Network Information Services (NIS) および Network Information Services Plus (NIS+) の各フィーチャーのサポートは、推奨されません。

### 詳細

ユーザー認証 のための NIS および NIS+ のサポートは、Linux および UNIX オペレーティング・システムで推奨されません。これに伴い、

**DB2\_ENABLE\_SINGLE\_NIS\_GROUP** レジストリー変数は推奨されなくなり、将来のリリースでは除去される可能性があります。

### 解決方法

一元化されたユーザー管理サービスに推奨されるソリューションは、Lightweight Directory Access Protocol (LDAP) です。バージョン 9.5 は、LDAP セキュリティー・プラグイン・モジュールの使用による LDAP に基づく認証およびグループ検索機能をサポートします。

NIS および NIS+ フィーチャーを DB2 環境で使用する方法については、バージョン 9.1 インフォメーション・センターを参照してください。

### 関連概念

データベース・セキュリティ・ガイドの『LDAP ベースの認証およびグループ参照のサポート』



---

## 第 18 章 廃止された機能

廃止された機能とは、もはや使用できなくなった機能のことです。前のリリースでそのような機能を使用していた場合、何らかの変更が必要です。

このセクションでは、バージョン 9.5 で廃止された機能について詳しく説明します。

---

### 拡張ストレージ (ESTORE) フィーチャー・サポートの廃止

拡張ストレージ (ESTORE) フィーチャー・サポートが廃止されました。

#### 詳細

バッファー・プールの「拡張ストレージ」オプションは、バージョン 9.1 で廃止されました。バージョン 9.5 では、関連する構成パラメーター、モニター・エレメント、および ESTORE への他のインターフェースが除去されました。64 ビット・オペレーティング・システムを使用するコンピューターには、32 ビット・オペレーティング・システムのようなメモリー制限がありませんが、こうしたコンピューターが普及してきたため、ESTORE フィーチャーの必要性がなくなりました。

ESTORE の廃止により、これをサポートしていたすべての 32 ビット・オペレーティング・システムのパフォーマンスが影響を受け、DB2 機能に次のような影響があります。

- **ESTORE\_SEG\_SZ** および **NUM\_ESTORE\_SEGS** 構成パラメーターが廃止されました。
- ESTORE モニター・エレメント **pool\_index\_to\_estore**、**pool\_data\_to\_estore**、**pool\_index\_from\_estore**、および **pool\_data\_from\_estore** が廃止され、スナップショットやイベント・モニター出力に表示されなくなりました。
- 推奨されない表関数 **SNAP\_GET\_TBSP**、**SNAP\_GET\_DB**、**SNAPSHOT\_BP**、**SNAPSHOT\_TBS**、**SNAPSHOT\_DATABASE**、および **SNAPSHOT\_APPL** は、廃止されたモニター・エレメントに関して **NULL** 値を返します。
- **ALTER BUFFERPOOL** および **CREATE BUFFERPOOL** ステートメントでは、**EXTENDED STORAGE** および **NOT EXTENDED STORAGE** オプションを使用できなくなりました。

#### 解決方法

より多くのメモリーを割り振るには、64 ビットのハードウェア、オペレーティング・システム、および DB2 製品にアップグレードする必要があります。さらに、アプリケーションやスクリプトを修正して、廃止された機能への参照を除去する必要があります。

---

## アドレス・ウィンドウ操作拡張 (AWE) フィーチャー・サポートの廃止 (Windows)

Address Windowing Extensions (アドレス・ウィンドウ操作拡張、AWE) フィーチャーのサポートが廃止されました。

### 詳細

以前のバージョンの DB2 では AWE フィーチャーがサポートされていました。このフィーチャーは、32 ビット DB2 サーバー上で稼働するアプリケーションが一定の制限を超えてメモリーを操作できるようにする、メモリー管理拡張機能のセットです。例えば、**DB2\_AWE** レジストリー変数を設定することにより、最大で 64 GB のメモリーを使用するバッファー・プールを割り振ることができました。しかし、64 ビット・プラットフォームが普及してきたため、AWE フィーチャーの必要性が低くなりました。

AWE の廃止により、32 ビット Windows プラットフォームではシステム・メモリーに対するアクセスが制限され、パフォーマンスが影響を受けます。DB2 機能には、次のような影響があります。

- **DB2\_AWE** レジストリー変数が廃止されました。
- **physical\_page\_maps** モニター・エレメントが廃止されました。
- SNAPBP 管理ビューには **physical\_page\_maps** 列が含まれなくなりました。
- SNAP\_GET\_BP 表関数は **physical\_page\_maps** 列に関して NULL 値を返します。

### 解決方法

現在 AWE を使用して使用可能なシステム・メモリー容量を増やしている 32 ビット・サーバーが存在する場合、64 ビットのハードウェア、オペレーティング・システム、および DB2 製品にアップグレードする必要があります。さらに、**DB2\_AWE** レジストリー変数または **physical\_page\_maps** モニター・エレメントへの参照を含んでいるスクリプトを更新する必要もあります。

---

## db2icrt、db2ilist、および db2iupdt の -w オプションの廃止 (Linux および UNIX)

db2icrt、db2iupdt、および db2ilist コマンドの **-w** WordWidth パラメーターが廃止されました。

### 詳細

**db2icrt**、**db2ilist**、および **db2iupdt** コマンドのビット幅オプション (**-w**) は無効になり、エラーを返します。このオプションは、AIX 5L、HP-UX、Linux、および Solaris オペレーティング・システムでのみ有効でした。サポートされる Linux および UNIX オペレーティング・システムでは、DB2 製品のインストール先のオペレーティング・システムによってインスタンスのビット・サイズが決定されるようになりました。



## 解決方法

db2icrt、db2ilist、および db2iupdt コマンドでは、**-w** オプションを使用しないでください。このオプションを使用すると、バージョン 9.1 では警告メッセージが返されましたが、バージョン 9.5 では構文エラーが返されます。

---

## DB2 Web ツール・サポートの廃止

DB2 Web ツールのサポートが廃止されました。

### 詳細

以前のバージョンの DB2 では DB2 Web ツールがサポートされました。これは、HTTP クライアント用に提供されている DB2 Web コマンド・センターおよび DB2 Web ヘルス・センターで構成されるスイートです。

### 解決方法

アプリケーションやスクリプトを修正して、廃止された機能への参照を除去します。データ・サーバー管理者ツールは、操作管理のための新しい Web ベースのツールです。

#### 関連概念

25 ページの『データ・サーバーの管理および開発用インターフェースの追加』

---

## いくつかのレジストリー変数および環境変数の廃止

バージョン 9.5 で廃止されたレジストリー変数が複数あります。これらに対する参照をすべて除去する必要があります。

以下のレジストリー変数および環境変数が、バージョン 9.5 で廃止されました。

表 23. バージョン 9.5 で廃止されたレジストリー変数

レジストリー変数または環境変数	詳細
<b>DB2_ASYNC_APPLY</b>	この変数は、DataJoiner (現在はサポートされていない) をサポートするのに使われていたので、バージョン 9.5 で廃止されました。
<b>DB2_AWE</b>	AWE バッファ・プールを使用していた場合、バージョン 9.5 の 64 ビット・データベース製品にマイグレーションして、仮想アドレス可能メモリー限度を除去することを考慮してください。詳細は、アドレス・ウィンドウ操作拡張 (AWE) フィーチャー・サポートの廃止 (Windows) を参照してください。
<b>DB2_BLOCK_ON_LOG_DISK_FULL</b>	この変数は、 <b>blk_log_dsk_ful</b> 構成パラメーターに置き換えられたので、バージョン 9.5 で廃止されました。
<b>DB2CCMSRV</b>	この変数は、必要なくなったのでバージョン 9.5 で廃止されました。

表 23. バージョン 9.5 で廃止されたレジストリー変数 (続き)

レジストリー変数または環境変数	詳細
<b>DB2_FORCE_FCM_BP</b>	AIX オペレーティング・システムの 64 ビット・カーネルのみがサポートされますが、これらには共用メモリー・セグメント・サイズ制限がないので、この変数はバージョン 9.5 で廃止されました。デフォルトで論理ノード間の共用メモリー通信が使用可能になります。これによって、パフォーマンスが向上し、他のプラットフォームとの整合性が得られます。
<b>DB2_LGPAGE_BP</b>	この変数はバージョン 9.5 で廃止されました。大規模ページ・サポートを使用可能にするには、 <b>DB2_LARGE_PAGE_MEM</b> レジストリー変数を使用してください。
<b>DB2LINUXAIO</b>	Linux オペレーティング・システムの非同期入出力 (AIO) 機能が、バージョン 9.5 ではデフォルトで使用可能になったので、このレジストリー変数は廃止されました。
<b>DB2_MIGRATE_TS_INFO</b>	この変数は、バージョン 5 よりも前のリリースからバージョン 5 へマイグレーションするのにのみ使用が必要とされるものだったので、バージョン 9.5 で廃止されました。
<b>DB2_NR_CONFIG</b>	この変数は、今後不要な機能なので、バージョン 9.5 で廃止されました。
<b>DB2_NEWLOGPATH2</b>	この変数は、データベース構成パラメーター <b>mirrorlogpath</b> (ログ・パスのミラーを設定するのにより柔軟性のあるパラメーター) に置き換えられたので、バージョン 9.5 で廃止されました。
<b>DB2_OLAP_BUFFER_SIZE</b>	バージョン 9.5 では、OLAP 関数はソート・ヒープ・メモリーを使用するので、このレジストリー変数は廃止されました。以前のリリースでは OLAP 関数はアプリケーション・ヒープ・メモリーを使用していて、 <b>DB2_OLAP_BUFFER_SIZE</b> はそれらが使用できる量を制限していました。ソート・ヒープ・メモリーはセルフチューニングが可能なので、この変数のロールは必要なくなりました。
<b>DB2UPMPR</b>	この変数は OS/2 (バージョン 9.5 はサポートしない) 上でのみ使われていたので、廃止されました。
<b>DB2UPMSINGLE</b>	この変数は、必要なくなったのでバージョン 9.5 で廃止されました。

#### 関連概念

173 ページの『レジストリー変数および環境変数のいくつかは推奨されない』

147 ページの『いくつかのレジストリー変数および環境変数の変更』

---

## db2undgp コマンドの廃止

db2undgp (実行特権の取り消し) コマンドは使用できなくなりました。

### 詳細

以前のバージョンでは、db2undgp を使用して、特権を持たないユーザーが SQL オブジェクトにアクセスしないようにできました。

### 解決方法

DB2 UDB バージョン 8 へのデータベース・マイグレーション中に、既存の関数、メソッド、および外部ストアド・プロシージャすべてに対する EXECUTE 特権が、すべてのユーザー (PUBLIC) に付与されました。db2undgp コマンドを使用して、特権を持たないユーザーが SQL オブジェクトにアクセスできないようにすることが可能でした。バージョン 9.5 では、EXECUTE 特権を PUBLIC グループから取り消すことができます。

---

## db2licm コマンドの -n オプションの廃止

db2licm コマンドの -n オプションは廃止されました。

### 詳細

過去には、-n オプションを使用して、1 つの DB2 製品と共に使用できるプロセッサの数を更新できました。ライセンスは現在、物理プロセッサ数ではなく、Value Unit (バリュー・ユニット) 数によって決定されます。したがって、このオプションをバージョン 9.5 またはそれ以降の製品で使用しても効果はありません。

### 解決方法

廃止されたこのオプションは使用しないでください。ライセンスのあるプロセッサ数を更新する必要はありません。

#### 関連資料

コマンド・リファレンスの『db2licm - ライセンス管理ツール・コマンド』

---

## CLISchema CLI キーワードの廃止

バージョン 9.5 データベース・サーバーに接続するバージョン 9.5 クライアントの場合に、CLISchema キーワードは廃止されました。

### 詳細

CLISchema キーワードの設定は、主に DB2 for z/OS に接続するクライアント・アプリケーションで、パフォーマンスを向上させました。バージョン 9.1 では、このキーワードのサポートは、DB2 Version 9.1 for Linux, UNIX, and Windows データベース・サーバーに接続するバージョン 9.1 クライアントに対して非推奨になり、DB2 for z/OS データベース・サーバーに接続するバージョン 9.1 クライアントに対しては廃止されました。

## 解決方法

代替スキーマを示すためには、代わりに **SysSchema** キーワードを使用できます。

### 関連資料

コール・レベル・インターフェース ガイドおよびリファレンス 第 2 巻の  
『SysSchema CLI/ODBC 構成キーワード』

---

## 第 19 章 DB2 バージョン 9.5 の使用に影響する DB2 バージョン 9.1 フィックスパックの変更

バージョン 9.1 フィックスパック 3 (およびこれ以前のフィックスパック) には、バージョン 9.5 の使用に影響する可能性のあるフィーチャーおよび機能の変更が含まれています。

### 詳細

バージョン 9.1 フィックスパック 3 以前のフィックスパックを適用しなかった場合、あるいはバージョン 9.1 が使用可能になって以来、ローカルのインフォメーション・センターを更新しなかった場合には、DB2 バージョン 9.5 の使用に影響する可能性のある変更について確認する必要があります。

### 解決方法

バージョン 9.1 フィックスパックに含まれている技術的な変更点に通じていない場合は、以下のトピックを参照してください。フィックスパックは累積的であるため、以前のフィックスパックの変更および機能性のすべてが含まれています。

#### DB2 バージョン 9.1 フィックスパック 1

フィックスパック 1 には、既存の機能に対する以下の変更点が含まれています。

- アドレス・ウィンドウ操作拡張 (AWE) サポートは推奨されていません (Windows)
- 連続ストリーミングを使用する JDBC アプリケーションでの LOB または XML 値の使用可能性が変更された
- 製品 ID の修正レベルには英数字が使用できる

フィックスパック 1 には、以下の機能拡張も含まれています。

- DB2Binder ユーティリティーに 2 つの新規オプションが含まれる
- 最適化プロファイル表はストアード・プロシージャによって作成できる

#### DB2 バージョン 9.1 フィックスパック 2

フィックスパック 2 には、フィックスパック 1 の機能と、既存の機能への以下の変更点も含まれます。

- java\_heap\_sz の変更された構成パラメーターのデフォルト値
- Network Information Services (NIS および NIS+) サポートは推奨されない (Linux および UNIX)
- db2sampl コマンドの -schema オプションの廃止

フィックスパック 2 には、以下の機能拡張も含まれています。

- アノテーション付き XML スキーマ分解のパフォーマンスが向上した

- C および C++ 組み込み SQL アプリケーションにおける BINARY、VARBINARY、および DECFLOAT データ・タイプのサポートが追加された
- .NET Framework 2.0 に対する DB2 .NET Data Provider の機能拡張とサポート
- HP-UX での DRDA および Informix ラッパー・サポートの追加
- Visual Studio 2005 用の IBM Database Add-In の機能拡張
- Solaris オペレーティング環境での IBM Software Development Kit (SDK) for Java 5.x のサポートの追加
- 新しいキーワードで、DPF 環境での照会パフォーマンスを向上できる
- Secure Sockets Layer (SSL) プロトコルはデータベース・サーバーによりサポートされる
- Windows Vista サポートの追加 (Windows)

### **DB2 バージョン 9.1 フィックスパック 3**

フィックスパック 3 には、フィックスパック 2 の機能と、既存の機能への以下の変更点も含まれます。

- LDAP セキュリティー・プラグインは更新されており、Open LDAP サーバーのサポートが含まれる

フィックスパック 3 には、以下の機能拡張も含まれています。

- サポートされるテリトリー・コードおよびコード・ページの追加
- パスワード変更サポートの追加 (Linux)
- COLLATION\_KEY\_BIT スカラー関数の追加
- JDBC および SQLJ の機能拡張
- HP-UX での Query Patroller サポートの追加

---

## 第 3 部 付録





---

## 付録 A. ファイル・システム・キャッシング構成

オペレーティング・システムは、デフォルトでは、ディスクとの間で読み書きされるファイル・データをキャッシュに入れます。

標準的な読み取り操作では、物理ディスク・アクセスにより、データがディスクからファイル・システム・キャッシュに読み取られ、そのデータがキャッシュからアプリケーション・バッファーにコピーされます。同様に、書き込み操作では、物理ディスク・アクセスにより、データがアプリケーション・バッファーからファイル・システム・キャッシュにコピーされ、そのデータがキャッシュから物理ディスクにコピーされます。CREATE TABLESPACE ステートメントの FILE SYSTEM CACHING 節には、このファイル・システム・レベルでのデータ・キャッシング動作が反映されます。データベース・マネージャーは、自身のデータ・キャッシングをバッファー・プールを使って管理します。バッファー・プールのサイズが適切に調整されていれば、ファイル・システム・レベルのキャッシングは必要ありません。

**注:** データベース・マネージャーは、ページのキャッシュを無効化することによって、既にほとんどの DB2 データのキャッシングを防止していますが、AIX の一時データと LOB は例外です。

ファイル・システム・レベルとバッファー・プールの両方でキャッシングを行うと、二重キャッシングに余分の CPU サイクルが必要なため、性能が低下する場合があります。この二重キャッシングを回避するため、大抵のファイル・システムにはファイル・システム・レベルでのキャッシングを使用不可にするフィーチャーがあります。これは一般に、バッファリングのない I/O と呼ばれています。UNIX では、このフィーチャーは一般に、*Direct I/O (DIO)* として知られています。

Windows では、これは FILE\_FLAG\_NO\_BUFFERING フラグを立ててファイルを開くことに相当します。さらに、IBM JFS2 や Symantec VERITAS VxFS などの一部のファイル・システムは、拡張された Direct I/O である高性能な *Concurrent I/O (CIO)* フィーチャーもサポートしています。データベース・マネージャーは、NO FILE SYSTEM CACHING 表スペース節を使用してこのフィーチャーをサポートしています。これが設定されているとき、データベース・マネージャーは、CIO が存在するファイル・システムで、自動的にこのフィーチャーを利用します。このフィーチャーは、ファイル・システム・キャッシュのメモリー要件を削減して他の用途に使用できるメモリーを増やすのに役立つことがあります。

バージョン 9.5 より前は、NO FILE SYSTEM CACHING も FILE SYSTEM CACHING も指定されていない場合は、キーワード FILE SYSTEM CACHING が暗黙指定されていました。バージョン 9.5 では、どちらのキーワードも指定されていない場合、デフォルトの NO FILE SYSTEM CACHING が使用されます。この変更は、新規に作成された表スペースにのみ影響します。バージョン 9.5 より前に作成された既存の表スペースは、影響を受けません。この変更は、AIX、Linux、Solaris、および Windows に適用されますが、以下の例外についてはデフォルトの動作は FILE SYSTEM CACHING のままです。

- AIX JFS

- Solaris 非 VxFS
- Linux for System z™
- すべての SMS TEMPORARY 表スペース・ファイル
- SMS 永続表スペース・ファイル。ただし、ロング・フィールド (LF) データ・ファイルとラージ・オブジェクト (LOB) データ・ファイルを除く。

デフォルト設定をオーバーライドするには、FILE SYSTEM CACHING または NO FILE SYSTEM CACHING を指定します。

## サポートされる構成

表 24 は、ファイル・システム・キャッシングなしで表スペースを使用する際にサポートされる構成を示しています。これは以下の点も示しています。(a) それぞれの場合に DIO と拡張 DIO のどちらが使用されるか。(b)表スペースでプラットフォームおよびファイル・システム・タイプに応じて NO FILE SYSTEM CACHING と FILE SYSTEM CACHING のどちらも指定されない場合のデフォルトの動作。

表 24. ファイル・システム・キャッシングを使用しない表スペースのサポートされる構成

プラットフォーム	ファイル・システムのタイプおよび必要な最小レベル	NO FILE SYSTEM CACHING が指定されているときに、データベース・マネージャーがサブミットした DIO 要求と CIO 要求	NO FILE SYSTEM CACHING と FILE SYSTEM CACHING のどちらも指定されない場合のデフォルトの動作
AIX 5.3+	Journal File System (JFS)	DIO	FILE SYSTEM CACHING (注 1 を参照。)
AIX 5.3+	Concurrent Journal File System (JFS2)	CIO	NO FILE SYSTEM CACHING
AIX 5.3+	VERITAS Storage Foundation for DB2 4.1 (VxFS)	CIO	NO FILE SYSTEM CACHING
HP-UX 11i (PA-RISC)	VERITAS Storage Foundation 4.1 (VxFS)	CIO	FILE SYSTEM CACHING
HP-UX バージョン 11i v2 (Itanium®)	VERITAS Storage Foundation 4.1 (VxFS)	CIO	FILE SYSTEM CACHING
Solaris 9	UNIX File System (UFS)	DIO	FILE SYSTEM CACHING (注 2 を参照。)
Solaris 10	UNIX File System (UFS)	CIO	FILE SYSTEM CACHING (注 2 を参照。)
Solaris 9、10	VERITAS Storage Foundation for DB2 4.1 (VxFS)	CIO	NO FILE SYSTEM CACHING
Linux ディストリビューション SLES 9+ および RHEL 4+  (対応アーキテクチャー: x86、x86_64、IA64、POWER)	ext2、ext3、reiserfs	DIO	NO FILE SYSTEM CACHING

表 24. ファイル・システム・キャッシングを使用しない表スペースのサポートされる構成 (続き)

プラットフォーム	ファイル・システムのタイプおよび必要な最小レベル	NO FILE SYSTEM CACHING が指定されているときに、データベース・マネージャーがサブミットした DIO 要求と CIO 要求	NO FILE SYSTEM CACHING と FILE SYSTEM CACHING のどちらも指定されない場合のデフォルトの動作
Linux ディストリビューション SLES 9+ および RHEL 4+  (対応アーキテクチャー: x86, x86_64, IA64, POWER)	VERITAS Storage Foundation 4.1 (VxFS)	CIO	NO FILE SYSTEM CACHING
Linux ディストリビューション SLES 9+ および RHEL 4+  (対応アーキテクチャー: zSeries®)	ファイバー・チャンネル・プロトコル (FCP) を使用する SCSI (Small Computer System Interface) ディスク上の ext2, ext3, または reiserfs	DIO	FILE SYSTEM CACHING
Windows	特定の要件なし。DB2 がサポートするすべてのファイル・システムを処理可能	DIO	NO FILE SYSTEM CACHING

**注:**

1. AIX JFS では、FILE SYSTEM CACHING がデフォルトです。
2. Solaris UFS では、FILE SYSTEM CACHING がデフォルトです。
3. データベース・マネージャー対応 VERITAS Storage Foundation では、オペレーティング・システムの前提条件が異なることがあります。上記にリストしたプラットフォームは、現行のリリースでサポートされているプラットフォームです。前提条件に関する情報については、VERITAS Storage Foundation for DB2 のサポートにご相談ください。
4. 上記の最小レベルの代わりに SFDB2 5.0 を使用する場合は、SFDB2 5.0 MP1 RP1 リリースを使用する必要があります。このリリースには、5.0 バージョンに固有のフィックスが含まれています。
5. データベース・マネージャーでデフォルト設定に対して NO FILE SYSTEM CACHING を選択したくない場合は、関連する SQL、コマンド、または API で FILE SYSTEM CACHING を指定してください。

**例**

**例 1:** デフォルトでは、この新規表スペースはバッファリングのない I/O を使用して作成されます。NO FILE SYSTEM CACHING 節が暗黙指定されています。

```
CREATE TABLESPACE table space name...
```

**例 2:** 以下のステートメントの NO FILE SYSTEM CACHING 節は、この特定の表スペースではファイル・システム・レベルのキャッシングが OFF になることを示しています。

```
CREATE TABLESPACE table space name ... NO FILE SYSTEM CACHING
```

**例 3:** 以下のステートメントは、既存の表スペースについてファイル・システム・レベルのキャッシングを使用不可にします。

```
ALTER TABLESPACE table space name ... NO FILE SYSTEM CACHING
```

**例 4:** 以下のステートメントは、既存の表スペースについてファイル・システム・レベルのキャッシングを使用可能にします。

```
ALTER TABLESPACE table space name ... FILE SYSTEM CACHING
```

---

## 付録 B. DB2 技術情報の概説

DB2 技術情報は、以下のツールと方法を介して利用できます。

- DB2 インフォメーション・センター
  - トピック (タスク、概念、およびリファレンス・トピック)
  - DB2 ツールのヘルプ
  - サンプル・プログラム
  - チュートリアル
- DB2 資料
  - PDF ファイル (ダウンロード可能)
  - PDF ファイル (DB2 PDF DVD に含まれる)
  - 印刷資料
- コマンド行ヘルプ
  - コマンド・ヘルプ
  - メッセージ・ヘルプ

**注:** DB2 インフォメーション・センターのトピックは、PDF やハードコピー資料よりも頻繁に更新されます。最新の情報を入手するには、資料の更新が発行されたときにそれをインストールするか、[ibm.com](http://ibm.com)<sup>®</sup> にある DB2 インフォメーション・センターを参照してください。

技術資料、ホワイト・ペーパー、IBM Redbooks<sup>®</sup> 資料などのその他の DB2 技術情報には、オンライン ([ibm.com](http://ibm.com)) でアクセスできます。DB2 Information Management ソフトウェア・ライブラリー・サイト (<http://www.ibm.com/software/data/sw-library/>) にアクセスしてください。

### 資料についてのフィードバック

DB2 の資料についてのお客様からの貴重なご意見をお待ちしています。DB2 の資料を改善するための提案については、[db2docs@ca.ibm.com](mailto:db2docs@ca.ibm.com) まで E メールを送信してください。DB2 の資料チームは、お客様からのフィードバックすべてに目を通しますが、直接お客様に返答することはありません。お客様が関心をお持ちの内容について、可能な限り具体的な例を提供してください。特定のトピックまたはヘルプ・ファイルについてのフィードバックを提供する場合は、そのトピック・タイトルおよび URL を含めてください。

DB2 お客様サポートに連絡する場合には、この E メール・アドレスを使用しないでください。資料を参照しても、DB2 の技術的な問題が解決しない場合は、お近くの IBM サービス・センターにお問い合わせください。

## DB2 テクニカル・ライブラリー (ハードコピーまたは PDF 形式)

以下の表は、DB2 ライブラリーについて説明しています。DB2 ライブラリーに関する詳細な説明については、[www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order) にある IBM Publications Center にアクセスしてください。英語の DB2 バージョン 9.5 のマニュアル (PDF 形式) とその翻訳版は、[www.ibm.com/support/docview.wss?rs=71&uid=swg2700947](http://www.ibm.com/support/docview.wss?rs=71&uid=swg2700947) からダウンロードできます。

この表には印刷資料が入手可能かどうかを示されていますが、国または地域によっては入手できない場合があります。

表 25. DB2 の技術情報

資料名	資料番号	印刷資料が入手可能かどうか
管理 API リファレンス	SC88-4431-00	入手可能
管理ルーチンおよびビュー	SC88-4435-00	入手不可
コール・レベル・インターフェース ガイドおよびリファレンス 第 1 巻	SC88-4433-00	入手可能
コール・レベル・インターフェース ガイドおよびリファレンス 第 2 巻	SC88-4434-00	入手可能
コマンド・リファレンス	SC88-4432-00	入手可能
データ移動ユーティリティガイドおよびリファレンス	SC88-4421-00	入手可能
データ・リカバリーと高可用性ガイドおよびリファレンス	SC88-4423-00	入手可能
データ・サーバー、データベース、およびデータベース・オブジェクトのガイド	SC88-4259-00	入手可能
データベース・セキュリティ・ガイド	SC88-4418-00	入手可能
ADO.NET および OLE DB アプリケーションの開発	SC88-4425-00	入手可能
組み込み SQL アプリケーションの開発	SC88-4426-00	入手可能
Java アプリケーションの開発	SC88-4427-00	入手可能
Perl および PHP アプリケーションの開発	SC88-4428-00	入手不可
SQL および 外部ルーチンの開発	SC88-4429-00	入手可能
データベース・アプリケーション 開発の基礎	GC88-4430-00	入手可能
DB2 インストールおよび管理概説 (Linux および Windows 版)	GC88-4439-00	入手可能
国際化対応ガイド	SC88-4420-00	入手可能

表 25. DB2 の技術情報 (続き)

資料名	資料番号	印刷資料が入手可能かどうか
メッセージ・リファレンス 第 1 巻	GI88-4109-00	入手不可
メッセージ・リファレンス 第 2 巻	GI88-4110-00	入手不可
マイグレーション・ガイド	GC88-4438-00	入手可能
<i>Net Search Extender</i> 管理および ユーザーズ・ガイド	SC88-4630-00	入手可能
注: この資料の内容は、DB2 イ ンフォメーション・センターに は含まれていません。		
パーティションおよびクラスタ リングのガイド	SC88-4419-00	入手可能
<i>Query Patroller</i> 管理およびユー ザーズ・ガイド	SC88-4611-00	入手可能
IBM データ・サーバー・クライ アント機能 概説およびインス トール	GC88-4441-00	入手不可
DB2 サーバー機能 概説および インストール	GC88-4440-00	入手可能
<i>Spatial Extender and Geodetic Data Management Feature</i> ユー ザーズ・ガイドおよびリファレ ンス	SC88-4629-00	入手可能
SQL リファレンス 第 1 巻	SC88-4436-00	入手可能
SQL リファレンス 第 2 巻	SC88-4437-00	入手可能
システム・モニター ガイドお よびリファレンス	SC88-4422-00	入手可能
テキスト検索ガイド	SC88-4424-00	入手可能
問題判別ガイド	GI88-4108-00	入手不可
データベース・パフォーマンス のチューニング	SC88-4417-00	入手可能
<i>Visual Explain</i> チュートリアル	SC88-4449-00	入手不可
新機能	SC88-4445-00	入手可能
ワークロード・マネージャー ガイドおよびリファレンス	SC88-4446-00	入手可能
<i>pureXML</i> ガイド	SC88-4447-00	入手可能
XQuery リファレンス	SC88-4448-00	入手不可

表 26. DB2 Connect 固有の技術情報

資料名	資料番号	印刷資料が入手可能かどうか
DB2 Connect Personal Edition 概説およびインストール	GC88-4443-00	入手可能

表 26. DB2 Connect 固有の技術情報 (続き)

資料名	資料番号	印刷資料が入手可能かどうか
DB2 Connect サーバー機能 概説およびインストール	GC88-4444-00	入手可能
DB2 Connect ユーザーズ・ガイド	SC88-4442-00	入手可能

表 27. Information Integration の技術情報

資料名	資料番号	印刷資料が入手可能かどうか
Information Integration: フェデレーテッド・システム 管理ガイド	SC88-4166-01	入手可能
Information Integration: レプリケーションおよびイベント・パブリッシングのための ASNCLP プログラム・リファレンス	SC88-4167-02	入手可能
Information Integration: フェデレーテッド・データ・ソース 構成ガイド	SC88-4185-01	入手不可
Information Integration: SQL レプリケーション ガイドおよびリファレンス	SC88-4168-01	入手可能
Information Integration: レプリケーションとイベント・パブリッシング 概説	GC88-4187-01	入手可能

## DB2 の印刷資料の注文方法

DB2 の印刷資料が必要な場合、オンラインで購入することができますが、すべての国および地域で購入できるわけではありません。DB2 の印刷資料については、IBM 営業担当員にお問い合わせください。DB2 PDF ドキュメンテーション DVD の一部のソフトコピー・ブックは、印刷資料では入手できないことに留意してください。例えば、「DB2 メッセージ・リファレンス」はどちらの巻も印刷資料としては入手できません。

DB2 PDF ドキュメンテーション DVD で利用できる DB2 の印刷資料の大半は、IBM に有償で注文することができます。国または地域によっては、資料を IBM Publications Center からオンラインで注文することもできます。お客様の国または地域でオンライン注文が利用できない場合、DB2 の印刷資料については、IBM 営業担当員にお問い合わせください。DB2 PDF ドキュメンテーション DVD に収録されている資料の中には、印刷資料として提供されていないものもあります。

注: 最新で完全な DB2 資料は、DB2 インフォメーション・センター (<http://publib.boulder.ibm.com/infocenter/db2luw/v9r5>) で参照することができます。

DB2 の印刷資料は以下の方法で注文することができます。



- 日本 IBM 発行のマニュアルはインターネット経由でご購入いただけます。詳しくは <http://www.ibm.com/shop/publications/order> の「ご注文について」をご覧ください。資料の注文情報にアクセスするには、お客様の国、地域、または言語を選択してください。その後、各ロケーションにおける注文についての指示に従ってください。
- DB2 の印刷資料を IBM 営業担当員に注文するには、以下のようになります。
  1. 以下の Web サイトのいずれかから、営業担当員の連絡先情報を見つけてください。
    - IBM Directory of world wide contacts ([www.ibm.com/planetwide](http://www.ibm.com/planetwide))
    - IBM Publications Web サイト (<http://www.ibm.com/shop/publications/order>)  
国、地域、または言語を選択し、お客様の所在地に該当する Publications ホーム・ページにアクセスしてください。このページから、「このサイトについて」のリンクにアクセスしてください。
  2. 電話をご利用の場合は、DB2 資料の注文であることをご指定ください。
  3. 担当者に、注文する資料のタイトルと資料番号をお伝えください。タイトルと資料番号は、200 ページの『DB2 テクニカル・ライブラリー (ハードコピーまたは PDF 形式)』でご確認いただけます。

---

## コマンド行プロセッサから SQL 状態ヘルプを表示する

DB2 は、SQL ステートメントの結果の原因になったと考えられる条件の SQLSTATE 値を戻します。SQLSTATE ヘルプは、SQL 状態および SQL 状態クラス・コードの意味を説明します。

SQL 状態ヘルプを呼び出すには、コマンド行プロセッサを開いて以下のように入力します。

```
? sqlstate or ? class code
```

ここで、*sqlstate* は有効な 5 桁の SQL 状態を、*class code* は SQL 状態の最初の 2 桁を表します。

例えば、? 08003 を指定すると SQL 状態 08003 のヘルプが表示され、? 08 を指定するとクラス・コード 08 のヘルプが表示されます。

---

## 異なるバージョンの DB2 インフォメーション・センターへのアクセス

DB2 バージョン 9.5 のトピックを扱っている DB2 インフォメーション・センターの URL は、<http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>です。

DB2 バージョン 9 のトピックを扱っている DB2 インフォメーション・センターの URL は <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>です。

DB2 バージョン 8 のトピックについては、バージョン 8 のインフォメーション・センターの URL <http://publib.boulder.ibm.com/infocenter/db2luw/v8/>にアクセスしてください。

## DB2 インフォメーション・センターにおける特定の言語でのトピックの表示

DB2 インフォメーション・センターでは、ブラウザの設定で指定した言語でのトピックの表示が試みられます。トピックがその指定言語に翻訳されていない場合は、DB2 インフォメーション・センターでは英語でトピックが表示されます。

• Internet Explorer Web ブラウザーで、指定どおりの言語でトピックを表示するには、以下のようにします。

1. Internet Explorer の「ツール」->「インターネット オプション」->「言語...」ボタンをクリックします。「言語の優先順位」ウィンドウがオープンします。
2. 該当する言語が、言語リストの先頭の項目に指定されていることを確認します。
  - リストに新しい言語を追加するには、「追加...」ボタンをクリックします。

注: 言語を追加しても、特定の言語でトピックを表示するのに必要なフォントがコンピューターに備えられているとはかぎりません。

- リストの先頭に新しい言語を移動するには、その言語を選択してから、その言語が言語リストに先頭に行くまで「上に移動」ボタンをクリックします。
3. ブラウザー・キャッシュを消去してから、ページをリフレッシュし、使用する言語で DB2 インフォメーション・センターを表示します。
- Firefox または Mozilla Web ブラウザーの場合に、使いたい言語でトピックを表示するには、以下のようにします。
1. 「ツール」->「オプション」->「詳細」ダイアログの「言語」セクションにあるボタンを選択します。「設定」ウィンドウに「言語」パネルが表示されます。
  2. 該当する言語が、言語リストの先頭の項目に指定されていることを確認します。
    - リストに新しい言語を追加するには、「追加...」ボタンをクリックしてから、「言語を追加」ウィンドウで言語を選択します。
    - リストの先頭に新しい言語を移動するには、その言語を選択してから、その言語が言語リストに先頭に行くまで「上に移動」ボタンをクリックします。
  3. ブラウザー・キャッシュを消去してから、ページをリフレッシュし、使用する言語で DB2 インフォメーション・センターを表示します。

ブラウザとオペレーティング・システムの組み合わせによっては、オペレーティング・システムの地域の設定も希望のロケールと言語に変更しなければならない場合があります。

## コンピューターまたはイントラネット・サーバーにインストールされた DB2 インフォメーション・センターの更新

DB2 インフォメーション・センターをローカルにインストールしている場合は、IBM から提供される更新をダウンロードおよびインストールすることができます。

ローカルにインストールされた DB2 インフォメーション・センターを更新するには、以下のことを行う必要があります。

1. コンピューター上の DB2 インフォメーション・センターを停止し、インフォメーション・センターをスタンドアロン・モードで再始動します。インフォメーション・センターをスタンドアロン・モードで実行すると、ネットワーク上の他のユーザーがそのインフォメーション・センターにアクセスできなくなります。これで、更新をダウンロードして適用できるようになります。
2. 「更新」機能を使用することにより、どんな更新が利用できるかを確認します。インストールする更新がある場合は、「更新」機能を使用してそれをダウンロードおよびインストールできます。

**注:** ご使用の環境において、インターネットに接続されていないマシンに DB2 インフォメーション・センターの更新をインストールする必要がある場合は、インターネットに接続されていて DB2 インフォメーション・センターがインストールされているマシンを使用して、更新サイトをローカル・ファイル・システムにミラーリングする必要があります。ネットワーク上の多数のユーザーが資料の更新をインストールする場合にも、更新サイトをローカルにミラーリングして、更新サイト用のプロキシーを作成することにより、個々のユーザーが更新を実行するのに要する時間を短縮できます。

更新パッケージが入手可能な場合、「更新」機能を使用してパッケージをダウンロードします。ただし、「更新」機能は、スタンドアロン・モードでのみ使用できます。

3. スタンドアロンのインフォメーション・センターを停止し、コンピューター上の DB2 インフォメーション・センターを再開します。

**注:** Windows Vista の場合、下記のコマンドは管理者として実行する必要があります。完全な管理者特権でコマンド・プロンプトまたはグラフィカル・ツールを起動するには、ショートカットを右クリックしてから、「管理者として実行」を選択します。

コンピューターまたはイントラネット・サーバーにインストール済みの DB2 インフォメーション・センターを更新するには、以下のようになります。

1. DB2 インフォメーション・センターを停止します。
  - Windows では、「スタート」→「コントロール パネル」→「管理ツール」→「サービス」をクリックします。次に、「DB2 インフォメーション・センター」サービスを右クリックして「停止」を選択します。
  - Linux では、以下のコマンドを入力します。

```
/etc/init.d/db2icdv95 stop
```
2. インフォメーション・センターをスタンドアロン・モードで開始します。
  - Windows の場合:
    - a. コマンド・ウィンドウを開きます。

- b. インフォメーション・センターがインストールされているパスにナビゲートします。デフォルトでは、DB2 インフォメーション・センターは <Program Files>¥IBM¥DB2 Information Center¥Version 9.5 ディレクトリーにインストールされています (<Program Files> は「Program Files」ディレクトリーのロケーション)。
- c. インストール・ディレクトリーの doc¥bin ディレクトリーにナビゲートします。
- d. 次のように help\_start.bat ファイルを実行します。

```
help_start.bat
```

• Linux の場合:

- a. インフォメーション・センターがインストールされているパスにナビゲートします。デフォルトでは、DB2 インフォメーション・センターは /opt/ibm/db2ic/V9.5 ディレクトリーにインストールされています。
- b. インストール・ディレクトリーの doc/bin ディレクトリーにナビゲートします。
- c. 次のように help\_start スクリプトを実行します。

```
help_start
```

システムのデフォルト Web ブラウザーが起動し、スタンドアロンのインフォメーション・センターが表示されます。

3. 「更新」ボタン (🔄) をクリックします。インフォメーション・センターの右側のパネルで、「更新の検索 (Find Updates)」をクリックします。既存の文書に対する更新のリストが表示されます。
4. ダウンロード・プロセスを開始するには、ダウンロードする更新をチェックして選択し、「更新のインストール (Install Updates)」をクリックします。
5. ダウンロードおよびインストール・プロセスが完了したら、「完了」をクリックします。
6. スタンドアロンのインフォメーション・センターを停止します。
  - Windows の場合は、インストール・ディレクトリーの doc¥bin ディレクトリーにナビゲートしてから、次のように help\_end.bat ファイルを実行します。

```
help_end.bat
```

注: help\_end バッチ・ファイルには、help\_start バッチ・ファイルを使用して開始したプロセスを安全に終了するのに必要なコマンドが含まれています。Ctrl-C または他の方法を使用して、help\_start.bat を終了しないでください。

- Linux の場合は、インストール・ディレクトリーの doc/bin ディレクトリーにナビゲートしてから、次のように help\_end スクリプトを実行します。

```
help_end
```

注: help\_end スクリプトには、help\_start スクリプトを使用して開始したプロセスを安全に終了するのに必要なコマンドが含まれています。他の方法を使用して、help\_start スクリプトを終了しないでください。

7. DB2 インフォメーション・センターを再開します。

- Windows では、「スタート」→「コントロール パネル」→「管理ツール」→「サービス」をクリックします。次に、「DB2 インフォメーション・センター」サービスを右クリックして「開始」を選択します。
- Linux では、以下のコマンドを入力します。

```
/etc/init.d/db2icdv95 start
```

更新された DB2 インフォメーション・センターに、更新された新しいトピックが表示されます。

---

## DB2 チュートリアル

DB2 チュートリアルは、DB2 製品のさまざまな機能について学習するのを支援します。この演習をとおして段階的に学習することができます。

### はじめに

インフォメーション・センター (<http://publib.boulder.ibm.com/infocenter/db2help/>) から、このチュートリアルの XHTML 版を表示できます。

演習の中で、サンプル・データまたはサンプル・コードを使用する場合があります。個々のタスクの前提条件については、チュートリアルを参照してください。

### DB2 チュートリアル

チュートリアルを表示するには、タイトルをクリックします。

「*pureXML* ガイド」の『**pureXML**』

XML データを保管し、ネイティブ XML データ・ストアに対して基本的な操作を実行できるように、DB2 データベースをセットアップします。

「*Visual Explain* チュートリアル」の『**Visual Explain**』

Visual Explain を使用して、パフォーマンスを向上させるために SQL ステートメントを分析し、最適化し、調整します。

---

## DB2 トラブルシューティング情報

DB2 製品を使用する際に役立つ、トラブルシューティングおよび問題判別に関する広範囲な情報を利用できます。

### DB2 ドキュメンテーション

トラブルシューティング情報は、DB2 問題判別ガイド、または DB2 インフォメーション・センターの「サポートおよびトラブルシューティング」セクションにあります。ここでは、DB2 診断ツールおよびユーティリティーを使用して、問題を切り分けて識別する方法、最も頻繁に起こる幾つかの問題に対するソリューションについての情報、および DB2 製品を使用する際に発生する可能性のある問題の解決方法についての他のアドバイスがあります。

### DB2 Technical Support の Web サイト

現在問題が発生していて、考えられる原因とソリューションを検索したい場合は、DB2 Technical Support の Web サイトを参照してください。

Technical Support サイトには、最新の DB2 資料、TechNotes、プログラム

診断依頼書 (APAR またはバグ修正)、フィックスパック、およびその他のリソースへのリンクが用意されています。この知識ベースを活用して、問題に対する有効なソリューションを探し出すことができます。

DB2 Technical Support の Web サイト (<http://www.ibm.com/software/data/db2/udb/support.html>) にアクセスしてください。

---

## ご利用条件

これらの資料は、以下の条件に同意していただける場合に限りご使用いただけます。

**個人使用:** これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布 (頒布、送信を含む) または表示 (上映を含む) することはできません。

**商業的使用:** これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。

お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入関連法規を含む、すべての関連法規を遵守するものとします。

IBM は、これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態を提供され、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任なしで提供されます。

---

## 付録 C. 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

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

**以下の保証は、国または地域の法律に沿わない場合は、適用されません。** IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書は、IBM 以外の Web サイトおよびリソースへのリンクまたは参照を含む場合があります。IBM は、本書より参照もしくはアクセスできる、または本書からリンクされた IBM 以外の Web サイトもしくは第三者のリソースに対して一切の責任を負いません。IBM 以外の Web サイトにリンクが張られていることにより IBM が当該 Web サイトを推奨するものではなく、またその内容、使用もしくはサイトの所有者について IBM が責任を負うことを意味するものではありません。また、IBM は、お客様が IBM Web サイトから第三者の存在を知ることになった場合にも (もしくは、IBM Web サイトから第三者へのリンクを使用した場合にも)、お客様と第三者との間のいかなる取引に対しても一切責任を負いません。従って、お客様は、IBM が上記の外部サイトまたはリソースの利用について責任を負うものではなく、また、外部サイトまたはリソースからアクセス可能なコンテンツ、サービス、

製品、またはその他の資料一切に対して IBM が責任を負うものではないことを承諾し、同意するものとします。第三者により提供されるソフトウェアには、そのソフトウェアと共に提供される固有の使用条件が適用されます。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Canada Limited  
Office of the Lab Director  
8200 Warden Avenue  
Markham, Ontario  
L6G 1C7  
CANADA

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

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

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

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

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:



本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生した創作物には、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. \_年を入れる\_. All rights reserved.

## 商標

DB2 バージョン 9.5 ドキュメンテーション・ライブラリーの資料に記載されている会社名、製品名、またはサービス名は、IBM Corporation の商標である可能性があります。IBM Corporation の商標については、<http://www.ibm.com/legal/copytrade.shtml> を参照してください。

以下は、それぞれ各社の商標または登録商標です。

Microsoft、Windows、Windows NT<sup>®</sup>、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。

Intel<sup>®</sup>、Intel ロゴ、Intel Inside<sup>®</sup> ロゴ、Intel Centrino<sup>®</sup>、Intel Centrino ロゴ、Celeron<sup>®</sup>、Intel Xeon<sup>®</sup>、Intel SpeedStep<sup>®</sup>、Itanium および Pentium<sup>®</sup> は、Intel 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です。

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

Linux は、Linus Torvalds の米国およびその他の国における商標です。

Adobe<sup>®</sup>、Adobe ロゴ、PostScript<sup>®</sup>、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。



# 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]

- アドレス・ウィンドウ操作拡張 (AWE)
  - 廃止されたサポート 186
- アプリケーション
  - エラー許容の改善 125
  - 新規サンプルのサマリー 90
  - Windows でのデプロイメントの機能拡張 23
  - Windows での配布の単純化 23
- アプリケーション開発
  - 機能拡張のサマリー 16, 69
  - 新規サンプルのサマリー 90
  - フェデレーションの機能強化 113
  - 変更のサマリー 156
- JDBC 3.0 の機能拡張 78
- JDBC 4.0 サポート 84
- Linux および UNIX 158
- SQLJ の機能拡張 78, 84
- アプリケーション・セーブポイント
  - フェデレーション・サポート 113
- イベント・モニター
  - 表書き込みの変更 131
- イベント・ログ・アクセス 124
- 印刷資料
  - 注文 202
- インスタンス
  - プラットフォーム間で適用可能なスタンドアロン・タイプ 111
- インストーラ
  - 機能拡張のサマリー 107
  - 非ルート・ユーザー 109
- インフォメーション・センター
  - 更新 205
  - バージョン 203
  - 別の言語で表示する 204
- エラー条件
  - db2fodc ユーティリティ 123
- オートノミックの機能拡張
  - 要約 3
- 応答ファイル
  - キーワードの追加 109
- オプティミスティック・ロック
  - 機能拡張 54

## [カ行]

- 外部ユーザー・マッピング・リポジトリ
    - C ベースおよび C++ ベースのインターフェース 114
  - 拡張ストレージ (ESTORE)
    - 廃止されたサポート 185
  - 拡張セキュリティー
    - Windows Vista 153
  - カタログ・ビュー
    - 追加 132
    - 変更 132
  - 各国語サポート
    - 機能拡張のサマリー 119
  - 環境変数
    - 推奨されない機能 173
    - 廃止されたサポート 187
    - 変更 147
  - 監査機能
    - 機能拡張 49, 167
    - 権限要件の変更 130
  - 関数
    - 推奨されない機能 132
    - スカラー関数
      - 発行 67
      - 文字ベースの処理 120
    - 追加 132
  - 表関数
    - 推奨されないスナップショット機能 185
    - ENV\_GET\_FEATURE\_INFO 38
    - 変更 132
  - DB2 XQuery
    - 機能拡張 66
    - 時刻 67
    - 日付 67
    - lower-case 66
    - upper-case 66
  - OLAP (Online Analytical Processing)
    - 機能拡張 56
  - SYSFUN 関数の SYSIBM バージョン 163
  - XSLTRANSFORM 63
- 管理
- 機能拡張のサマリー 3, 25
  - データ・サーバー開発者ツール 25
  - 非ルート・ユーザー 109
  - 変更のサマリー 129
- 管理ビュー
- 追加 36, 132
  - 変更 132
  - ENV\_FEATURE\_INFO 38
- 管理ルーチン
- 追加 36, 132

管理ルーチン (続き)

変更 132

キャスト

XQuery データ・タイプ 67

キャスト可能式

XQuery 67

キャッシング

ファイル・システム、表スペース用 195

組み込み関数

追加 132

変更 132

組み込みルーチン

追加 132

変更 132

クラスター

管理 100, 107

クラスター管理 API 105

構成 100, 105

グローバル変数

概要 74

コード・ページ

Unicode デフォルト 129

コールアウト・スクリプト

カスタマイズされたバージョン 170

高可用性

機能拡張のサマリー 14, 97

高可用性災害時リカバリー (HADR)

ピア・ウィンドウ 101

更新

インフォメーション・センター 205

DB2 インフォメーション・センター 205

更新式

概要 61

更新の検出

機能拡張 54

構成

非ルート・ユーザー 109

フェデレーションの機能強化 115

複数パーティション・データベース 28

構成パラメーター

機能拡張 30

推奨されない機能 140, 155

追加 140

廃止されたサポート 140

変更 140, 155

app1\_memory 28

auto\_del\_rec\_obj 99

db2Import

機能拡張 162

db2Load

機能拡張 162

enable\_xmlchar 65

estore\_seg\_sz

廃止されたサポート 185

hadr\_peer\_window 101

構成パラメーター (続き)

num\_estore\_segs

廃止されたサポート 185

コマンド

変更のサマリー 164

BACKUP DATABASE

機能拡張 166

BIND

インストール後の自動化 154

db2audit

変更 167

db2ckmig

機能拡張 169

db2icrt

廃止されたオプション 186

db2ilist

廃止されたオプション 186

db2iupdt

廃止されたオプション 186

db2licm

廃止されたオプション 189

変更 156

db2look

DDL 生成の拡張機能 36

db2mtrk

変更 169

db2pd

EDU の機能拡張 170

db2undgp

廃止されたサポート 189

IMPORT

推奨されないオプション 180

INSPECT

索引検査の機能拡張 124

installFixPack

機能拡張 154

LOAD

pureXML データ 62

XML データ 62

ps

出力の変更 170

REDISTRIBUTE DATABASE PARTITION GROUP

機能拡張 33

メッセージ・ファイルの変更 165

ROLLFORWARD

最小リカバリー時間の機能拡張 103

UPDATE XMLSCHEMA 66

コマンド行プロセッサ (CLP)

コマンドの変更 164

スクリプト 92

ご利用条件

資料の使用 208

コンプレッション・ディクショナリー

自動作成 30, 130

コンポーネント

名前の変更 24

## [サ行]

- サーバー・フィックスバック・イメージ
  - 製品の更新の単純化 108
- 再配分
  - 機能拡張 33
- 索引
  - 作成の改善 56
  - 増加したサイズ 138
  - データ不整合の許容の機能拡張 125
- 作成されたセッション・グローバル変数
  - 概要 74
- サンプル
  - 追加のサマリー 90
- 時刻関数
  - DB2 XQuery 67
- システム・カタログ・ビュー
  - 追加 132
  - 変更 132
- システム・コマンド
  - 変更のサマリー 164
- 自動保守
  - 構成ポリシー 97
  - ポリシー情報収集 97
- 照会
  - パフォーマンス向上 53, 57
- 照合
  - Unicode 言語認識サポート 119
- 資料
  - 使用に関するご利用条件 208
  - PDF および印刷資料 200
- 資料の概説 199
- 新機能
  - 要約 1
- 新規フィーチャー
  - 要約 1
- 推奨されない機能
  - 要約 127, 173
- スカラー関数
  - アプリケーションの移植性 95
  - 概要 96
  - ストリング 120
  - 発行 67
  - ビット単位の操作 96
  - 文字ベースの処理 120
- スケラビリティ
  - 機能拡張のサマリー 7
- ストアド・プロシージャ
  - 自動保守ポリシー 97
  - XSR\_UPDATE 66
- ストリング・リテラル
  - Unicode 119
- ストレージ
  - pureXML
    - スペースの縮小 65

- ストレージ (続き)
  - XML
    - スペースの縮小 65
- ストレージ・キー
  - バッファ・プール保護 124
- スナップショット・バックアップ
  - 概要 104
- スナップショット・モニター
  - 推奨されない形式 181
- 整合変更データ (CCD) 表
  - CD-UOW 結合なし 117
- 製品
  - 可用性 23
  - パッケージ 23
- セキュリティ
  - 機能拡張のサマリー 12, 47
  - トラステッド・コンテキスト 47
  - フェデレーションの機能強化 114
  - LBAC 機能拡張 52
- セキュリティ管理者 (SECADM) 権限
  - データベース監査 130
- 設計
  - データ・サーバー開発者ツール 25
- セルフチューニング・メモリー
  - 機能拡張 28

## [タ行]

- 単一システム・ビュー (SSV)
  - バックアップ 102
- チュートリアル
  - トラブルシューティングと問題判別 207
  - Visual Explain 207
- 直接入出力 (DIO)
  - デフォルトの使用 152
- データ
  - 圧縮の機能拡張 30, 130
  - 再配分の機能拡張 33
  - 整合性の検査の機能拡張 124
- データベース
  - 開発 25
  - 管理 25
  - 言語認識照合のサポート 119
  - 設計 25
  - セットアップ変更のサマリー 140
  - デフォルト・コード・ページ 129
  - デプロイメント 25
  - 複数パーティション構成 28
  - NO FILE SYSTEM CACHING デフォルト 152
  - Unicode 119, 129
- データ・サーバー開発者ツール
  - インストール 108
  - 概要 25
  - Windows Vista のサポート 111
- データ・サーバー管理者ツール
  - インストール 108

データ・サーバー管理者ツール (続き)

概要 25

データ・タイプ

10 進浮動小数点 76

ARRAY 76

DECFLOAT 76

レプリケーション・サポート 117

XQuery データ・タイプへの値のキャスト 67

データ・リカバリー

機能拡張のサマリー 14

ログインの改善 101

ディクショナリー自動作成 (ADC)

概要 30

デプロイメント

データ・サーバー開発者ツール 25

統計

フェデレーションの機能強化 115

リアルタイムの収集 26

統計プロファイル

リセット 39

動的ステートメント・キャッシュ

無効化 139

特殊レジスター

長さの変更 164

SCHEMA

戻り値の変更 161

USER

戻り値の変更 161

特記事項 209

特権

ロール 51

トラステッド・コンテキスト

概要 47

フェデレーション・サポート 114

トラブルシューティング

オンライン情報 207

機能拡張のサマリー 123

チュートリアル 207

db2fodc ユーティリティ 123

## [ナ行]

ニックネーム

フェデレーションの機能強化 115

入出力

pureXML での効率の改善 65

XML での効率の改善 65

## [ハ行]

パーティション・データベース環境

単一システム・ビュー (SSV) を使用したバックアップ 102

廃止された機能

要約 127, 185

バックアップ

機能拡張のサマリー 97

スナップショット 98, 104

特権の変更 137

パーティション・データベースの機能拡張 102

ファイル許可の変更 137

複数パーティション・データベースの機能拡張 102, 166

発行

XMLGROUP スカラー関数 67

XMLROW スカラー関数 67

XSLTRANSFORM スカラー関数 67

バッファ・プール

保護 124

パフォーマンス

機能拡張のサマリー 7, 53

索引作成の機能拡張 56

照会の機能拡張 53, 57

pureXML の機能拡張 62, 65

XML の機能拡張 62, 65

日付関数

DB2 XQuery 67

ビット単位のスカラー関数

概要 96

ビュー

追加 132

変更 132

SNAPBP

変更 186

表書き込みイベント・モニター

ターゲット表の変更 131

表関数

推奨されない機能 132

ENV\_GET\_FEATURE\_INFO 38

SNAP\_GET\_BP

変更 186

表スペース

スペース再利用の機能拡張 38

ファイル・システム・キャッシングなし 195

並行入出力 (CIO) デフォルト 57

CIO (並行入出力) デフォルト 57

NO FILE SYSTEM CACHING デフォルト 152

非ルート・インストール

概要 109

ファイル・システム

表スペースのキャッシング 195

ファイル・システム・キャッシュ

メモリーの縮小 57

フィックスパック

インストール後の自動化 108, 154

機能拡張のサマリー 107

製品の更新の単純化 108

バージョン 9.5 に影響するバージョン 9.1 の変更 191

フェイルオーバー

高可用性災害時リカバリー (HADR)

ピア・ウィンドウ 101

ピア・ウィンドウ 101

- フェイルオーバー (続き)
  - HADR (高可用性災害時リカバリー)
    - ピア・ウィンドウ 101
- フェデレーション
  - アプリケーション開発の機能拡張 113
  - 機能拡張のサマリー 113
  - 構成の機能拡張 115
  - セキュリティの向上 114
- 複数パーティション・データベース
  - 構成 28
  - 単一システム・ビュー (SSV) を使用したバックアップ 102
- プロシージャ
  - 自動保守ポリシー 97
  - ADMIN\_CMD
    - 機能拡張 166
- 並行性の制御
  - 機能拡張 54
- 並行入出力 (CIO)
  - デフォルトの使用 57, 152
- ヘルプ
  - 表示 204
  - SQL ステートメントの 203
- 変更された機能
  - 要約 127, 129
- 変数
  - グローバル 74

## [マ行]

- マージ・モジュール
  - 非 DB2 サポート 110
- マイグレーション
  - 機能拡張のサマリー 107
  - Windows
    - インターフェースの変更 138
- マルチスレッド化アーキテクチャ
  - 利点 27
- マルチディメンション・クラスタリング (MDC) 表
  - 据え置き索引クリーンアップのロールアウト 55
  - ロールアウト削除 55
- マルチバイト文字
  - Perl ドライバー 72
- メモリー
  - 構成の機能拡張 28
  - メモリー・ビジュアライザーの機能拡張 136
  - CLI/ODBC アプリケーションによる消費量の増加 161
  - NO FILE SYSTEM CACHING キーワードを使用して縮小 57
- メモリー・ビジュアライザー
  - 情報の機能拡張 136
- モニター
  - 改善 37
  - データ・サーバー管理ツール 25
- モニター・エレメント
  - 機能拡張のサマリー 37
  - 推奨されない機能 177

- モニター・エレメント (続き)
  - ESTORE のサポート廃止 185
  - physical\_page\_maps
    - 廃止されたサポート 186
- 問題判別
  - オンライン情報 207
  - 機能拡張のサマリー 123
  - チュートリアル 207

## [ヤ行]

- ユーティリティ
  - 非互換性 163
  - db2fodc 123

## [ラ行]

- ライセンス交付
  - モニターの改善 38
- ラベル・ベースのアクセス制御 (LBAC)
  - 機能拡張 52
- ランチパッド
  - 「マイグレーション」ボタン 138
- リアルタイムの統計収集
  - 概要 26
- リカバリー
  - 機能拡張のサマリー 97
  - 単純化 99
  - ロギングの改善 179
- リテラル
  - Unicode スtring 119
- ルーチン
  - 追加 132
  - 変更 132
- レジストリー変数
  - 推奨されない機能 173
  - 廃止されたサポート 187
  - 変更 147
  - DB2\_AWE
    - 廃止されたサポート 186
  - DB2\_MEMORY\_PROTECT 124
- レプリケーション
  - 機能拡張のサマリー 117
  - DECFLOAT データ・タイプ 117
- ロード
  - データ
    - 圧縮の機能拡張 30
    - XML 62
  - ロード・ユーティリティ
    - pureXML データ 62
    - XML データ 62
- ロール
  - 概要 51
- ロギング
  - 機能拡張のサマリー 97

ログ  
二重制御ファイル 179  
ログ制御ファイル  
二重コピー 101, 179  
ロック・タイムアウト 38

## [ワ行]

ワークロード管理  
改善 41  
概要 41  
機能拡張のサマリー 41

## [数字]

10 進浮動小数点データ・タイプ  
概要 76  
レプリケーション・サポート 117

## A

ADC (自動ディクショナリー作成)  
概要 30  
ADMIN\_CMD プロシージャ  
複数パーティション・バックアップの機能拡張 166  
ADMIN\_GET\_TAB\_INFO 関数  
推奨されない機能 132  
agentpri 構成パラメーター  
推奨されない機能 140  
agents\_waiting\_on\_token モニター・エレメント  
推奨されない機能 177  
agents\_waiting\_top モニター・エレメント  
推奨されない機能 177  
ALTER BUFFERPOOL ステートメント  
ESTORE の変更 185  
ALTER SECURITY LABEL COMPONENT ステートメント  
概要 52  
ALTER SECURITY POLICY ステートメント  
概要 52  
ALTER TABLESPACE ステートメント  
スペース再利用の機能拡張 38  
API  
非互換性 163  
db2Backup  
機能拡張 166  
db2inspect  
索引検査の機能拡張 124  
db2Rollforward  
最小リカバリ時間の機能拡張 103  
piActionString データ構造  
推奨されない機能 182  
sqluadaw  
推奨されない機能 176  
appgroup\_mem\_sz 構成パラメーター  
推奨されない機能 140, 155

applheapsz 構成パラメーター  
機能拡張 30, 155  
変更 140  
appl\_memory 構成パラメーター  
概要 28, 140  
app\_ctl\_heap\_sz 構成パラメーター  
推奨されない機能 140, 155  
ARRAY データ・タイプ  
概要 76  
authority\_lvl モニター・エレメント  
推奨されない機能 177  
AUTOMAINT\_GET\_POLICY ストアード・プロシージャ  
概要 97  
AUTOMAINT\_GET\_POLICYFILE ストアード・プロシージャ  
概要 97  
AUTOMAINT\_SET\_POLICY ストアード・プロシージャ  
概要 97  
AUTOMAINT\_SET\_POLICYFILE ストアード・プロシージャ  
概要 97  
auto\_del\_rec\_obj 構成パラメーター  
概要 99, 140  
auto\_stmt\_stats 構成パラメーター  
概要 140  
AWE (アドレス・ウィンドウ操作拡張)  
廃止されたサポート 186

## B

BACKUP DATABASE コマンド  
複数パーティション・バックアップの機能拡張 166  
Big5-HKSCS  
Unicode 変換 120  
BIND コマンド  
インストール後の自動化 154

## C

catalogcache\_sz 構成パラメーター  
デフォルト値の変更 140  
cat\_cache\_size\_top モニター・エレメント  
推奨されない機能 177  
CCD (整合変更データ) 表  
CD-UOW 結合なし 117  
CIO (並行入出力)  
デフォルトの使用 57, 152  
CLIENT APPLNAME 特殊レジスター  
自動設定 92  
CLISchema CLI キーワード  
廃止されたサポート 189  
CLI/ODBC アプリケーション  
メモリー消費量の増加 161  
CLP (コマンド行プロセッサ)  
コマンドの変更 164  
スクリプト 92



cluster\_mgr 構成パラメーター  
 概要 140

comm\_bandwidth 構成パラメーター  
 デフォルト値の変更 140

CREATE BUFFERPOOL ステートメント  
 ESTORE の変更 185

CREATE DATABASE ステートメント  
 NO FILE SYSTEM CACHING デフォルト 152

CREATE INDEX ステートメント  
 パフォーマンス向上 56

CREATE TABLESPACE ステートメント  
 NO FILE SYSTEM CACHING デフォルト 152

CURRENT DEFAULT TRANSFORM GROUP 特殊レジスター  
 長さの変更 164

CURRENT PATH 特殊レジスター  
 長さの変更 164

## D

database\_memory 構成パラメーター  
 機能拡張 30  
 変更されたデフォルト値 140

DB2 ACS (拡張コピー・サービス) API  
 スナップショット・バックアップ 98

DB2 Connect  
 機能拡張のサマリー 19

DB2 Developer Workbench  
 機能拡張 93  
 名前の変更 24, 93

DB2 Driver for JDBC and SQLJ  
 名前の変更 24

DB2 Driver for ODBC and CLI  
 名前の変更 24

DB2 JDBC Type 2 Driver for Linux, UNIX, and Windows  
 推奨されない機能 156

DB2 Runtime Client  
 名前の変更 24  
 マージ・モジュール 110

DB2 Spatial Extender  
 照会パフォーマンスの改善 58

DB2 Web ツール  
 廃止されたサポート 187

DB2 XQuery 関数  
 時刻 67  
 日付 67  
 lower-case  
 ロケール・サポートの機能拡張 66  
 upper-case  
 ロケール・サポートの機能拡張 66

DB2 インフォメーション・センター  
 更新 205  
 バージョン 203  
 別の言語で表示する 204

DB2 拡張コピー・サービス (ACS) API  
 スナップショット・バックアップ 98

DB2 クライアント  
 名前の変更 24

DB2 資料の印刷方法 202

DB2 バージョン 9.5  
 ハイライト 3

DB2ATLTD\_PORTS レジストリー変数  
 推奨されない機能 173

db2audit コマンド  
 機能拡張 49  
 変更 130, 167

db2Backup API  
 複数パーティション・バックアップの機能拡張 166

DB2BPVARS レジストリー変数  
 推奨されない機能 173

DB2CCMSRV  
 廃止されたサポート 187

db2ckmig コマンド  
 機能拡張 169

DB2CLIINIPATH  
 変更 147

db2cos スクリプト  
 カスタマイズされたバージョン 170

db2cos\_datacorruption スクリプト  
 カスタマイズされたバージョン 170

db2cos\_hang スクリプト  
 カスタマイズされたバージョン 170

db2cos\_trap スクリプト  
 カスタマイズされたバージョン 170

DB2COUNTRY レジストリー変数  
 推奨されない機能 173

DB2DEFPREP レジストリー変数  
 推奨されない機能 173

DB2DMNBCKCTLR レジストリー変数  
 推奨されない機能 173

DB2FFDC レジストリー変数  
 推奨されない機能 173

DB2FODC  
 概要 147

db2haicu コーティリティー  
 概要 100

db2icrt コマンド  
 廃止されたオプション 186

db2ilist コマンド  
 廃止されたオプション 186

db2Import API  
 piActionString データ構造  
 推奨されない機能 182

db2Import 構成パラメーター  
 ID の機能拡張 162

db2inspect API  
 索引検査の機能拡張 124

DB2INSTPROF  
 変更されたデフォルト値 147

db2iupdt コマンド  
 廃止されたオプション 186

db2licm コマンド  
 廃止されたオプション 189  
 変更された製品 ID 値 156

DB2LINUXAIO  
 廃止されたサポート 187

db2Load API  
 piActionString データ構造  
 推奨されない機能 182

db2Load 構成パラメーター  
 ID の機能拡張 162

db2look コマンド  
 DDL 生成の拡張機能 36

DB2MEMMAXFREE レジストリー変数  
 推奨されない機能 173  
 変更 147

db2mtrk コマンド  
 変更 155, 169

DB2NTNOCACHE レジストリー変数  
 推奨されない機能 173

DB2NTPRICLASS レジストリー変数  
 推奨されない機能 173

db2pd コマンド  
 EDU の機能拡張 170

DB2PRIORITIES レジストリー変数  
 推奨されない機能 173  
 変更 147

db2Rollforward API  
 最小リカバリ時間の機能拡張 103

DB2ROUTINE\_DEBUG レジストリー変数  
 推奨されない機能 173

db2undgp コマンド  
 廃止されたサポート 189

DB2UPMPR  
 廃止されたサポート 187

DB2UPMPSINGLE  
 廃止されたサポート 187

DB2YIELD レジストリー変数  
 推奨されない機能 173

DB2\_ALLOCATION\_SIZE レジストリー変数  
 推奨されない機能 173

DB2\_ASYNC\_APPLY  
 廃止されたサポート 187

DB2\_ASYNC\_IO\_MAXFILOP レジストリー変数  
 推奨されない機能 173

DB2\_AWE レジストリー変数  
 廃止されたサポート 186, 187

DB2\_BAR\_AUTONOMIC\_DISABLE レジストリー変数  
 推奨されない機能 173

DB2\_BLOCK\_ON\_LOG\_DISK\_FULL  
 廃止されたサポート 187

DB2\_CAPTURE\_LOCKTIMEOUT  
 概要 147

DB2\_ENABLE\_SINGLE\_NIS\_GROUP レジストリー変数  
 推奨されない機能 173

DB2\_EVMON\_EVENT\_LIST\_SIZE  
 概要 147

DB2\_EXTENDED\_IO\_FEATURES  
 変更 147

DB2\_FORCE\_FCM\_BP  
 廃止されたサポート 187

DB2\_HASH\_JOIN レジストリー変数  
 推奨されない機能 173

DB2\_INDEX\_FREE レジストリー変数  
 推奨されない機能 173

DB2\_LARGE\_PAGE\_MEM  
 変更 147

DB2\_LGPAGE\_BP  
 廃止されたサポート 187

DB2\_MAP\_XML\_AS\_CLOB\_FOR\_DLC レジストリー変数  
 推奨されない機能 173

DB2\_MDC\_ROLLOUT  
 変更 147

DB2\_MEMORY\_PROTECT レジストリー変数  
 概要 147  
 バッファ・プール保護 124

DB2\_MIGRATE\_TS\_INFO  
 廃止されたサポート 187

DB2\_NEWLOGPATH2  
 廃止されたサポート 187

DB2\_NO\_FORK\_CHECK レジストリー変数  
 推奨されない機能 173

DB2\_NR\_CONFIG  
 廃止されたサポート 187

DB2\_OLAP\_BUFFER\_SIZE  
 廃止されたサポート 187

DB2\_OPTSTATS\_LOG  
 概要 147

DB2\_PARTITIONEDLOAD\_DEFAULT レジストリー変数  
 推奨されない機能 173

DB2\_PINNED\_BP  
 変更 147

DB2\_RESOURCE\_POLICY  
 変更 147

DB2\_RR\_TO\_RS レジストリー変数  
 推奨されない機能 173

DB2\_SET\_MAX\_CONTAINER\_SIZE レジストリー変数  
 概要 147

DB2\_SNAPSHOT\_NOAUTH レジストリー変数  
 推奨されない機能 173

DB2\_SYSTEM\_MONITOR\_SETTINGS  
 概要 147

DB2\_THREAD\_SUSPENSION  
 概要 147

DB2\_TRUSTED\_BINDIN レジストリー変数  
 推奨されない機能 173

DB2\_UPDATE\_PART\_KEY レジストリー変数  
 推奨されない機能 173

DB2\_UPDDBCFG\_SINGLE\_DBPARTITION  
 概要 147

DB2\_USE\_DB2JCCT2\_JROUTINE  
 変更 147

DB2\_VENDOR\_INI レジストリー変数  
推奨されない機能 173

DB2\_WORKLOAD  
変更 147

dbheap 構成パラメーター  
機能拡張 30, 155  
変更されたデフォルト値 140

db\_heap\_top モニター・エレメント  
推奨されない機能 177

DECFLOAT データ・タイプ  
概要 76  
レプリケーション・サポート 117

decflt\_rounding 構成パラメーター  
概要 140

DECODE スカラー関数  
アプリケーションの移植性 95

DEGREES 関数  
SYSIBM バージョン 163

DESCRIBE コマンド  
出力の変更 164

Developer Workbench  
機能拡張 93  
名前の変更 93

DIO (直接入出力)  
デフォルトの使用 152

## E

enable\_xmlchar 構成パラメーター  
概要 65, 140

ENV\_FEATURE\_INFO 管理ビュー  
概要 38

ENV\_GET\_FEATURE\_INFO 表関数  
概要 38

ESTORE (拡張ストレージ)  
廃止されたサポート 185

ESTORE\_SEG\_SZ 構成パラメーター  
廃止されたサポート 185

estore\_seg\_sz 構成パラメーター  
廃止されたサポート 140

EXP 関数  
SYSIBM バージョン 163

EXPLAIN 可能ステートメント  
REFRESH TABLE 59  
SET INTEGRITY 59

## F

federated\_async 構成パラメーター  
変更 140

fenced\_pool 構成パラメーター  
デフォルト値の変更 140

First Occurrence Data Capture ツール  
概要 123

## G

GET AUTHORIZATIONS コマンド  
推奨されない機能 176

GET DB CFG コマンド  
出力の変更 164

GET SNAPSHOT コマンド  
出力の変更 164

GRANT EXEMPTION ステートメント  
機能拡張 52

GRANT SECURITY LABEL ステートメント  
機能拡張 52

GREATEST スカラー関数  
アプリケーションの移植性 95

groupheap\_ratio 構成パラメーター  
推奨されない機能 140, 155

## H

HADR (高可用性災害時リカバリー)  
ピア・ウィンドウ 101

hadr\_peer\_window 構成パラメーター  
概要 101, 140

## I

IBM Data Server Driver for JDBC and SQLJ  
デフォルト・ドライバ 156

IBM Data Server Driver for ODBC, CLI, and .NET  
概要 23

IBM Data Server Provider for .NET  
データ・サーバー・サポート 89

IBM Tivoli System Automation for Multiplatforms  
アンインストール 107  
インストール 107  
更新 107

IBM\_DB2 PHP 拡張  
概要 70  
パッケージ 70

ID  
長さの検査の機能拡張 162  
長さの変更 69, 160, 163

IMPORT コマンド  
推奨されないオプション 180

INSERT スカラー関数  
Unicode 可変文字サイズ・サポート 120, 132

INSPECT コマンド  
索引検査の機能拡張 124

installFixPack コマンド  
機能拡張 154

instance\_memory 構成パラメーター  
機能拡張 30, 155  
変更 140

intra\_parallel 構成パラメーター  
変更 140

## J

### Java

デフォルト・ドライバーの変更 156

### JDBC

デフォルト・ドライバーの変更 156

#### 3.0

機能拡張 78

#### 4.0

機能拡張 84

ResultSetMetaData.getColumnLabel の変更 157

ResultSetMetaData.getColumnName の変更 157

JAR ファイル名 77

## L

LBAC (ラベル・ベースのアクセス制御)

機能拡張 52

LEAST スカラー関数

アプリケーションの移植性 95

LEFT スカラー関数

Unicode 可変文字サイズ・サポート 120, 132

LN 関数

SYSFUN 関数の SYSIBM バージョン 163

LOB 列

照会パフォーマンスの改善 53

LOG 関数

SYSFUN 関数の SYSIBM バージョン 163

LOG10 関数

SYSFUN 関数の SYSIBM バージョン 163

logretain 構成パラメーター

推奨されない機能 140

LOWER スカラー関数

ロケール依存 121

lower-case 関数

ロケール・サポートの機能拡張 66

## M

MAX スカラー関数

アプリケーションの移植性 95

maxagents 構成パラメーター

推奨されない機能 140

maxcagents 構成パラメーター

推奨されない機能 140

maxfilop 構成パラメーター

変更 140

max\_agents\_overflows モニター・エレメント

推奨されない機能 177

max\_connections 構成パラメーター

デフォルト値の変更 140

max\_coordagents 構成パラメーター

変更 140

MDC (マルチディメンション・クラスタリング) 表

据え置き索引クリーンアップのロールアウト 55

ロールアウト削除 55

## 222 新機能

MIN スカラー関数

アプリケーションの移植性 95

mon\_heap\_sz 構成パラメーター

機能拡張 30, 155

変更されたデフォルト値 140

## N

Network Information Services (NIS)

推奨されないサポート 183

Network Information Services Plus (NIS+)

推奨されないサポート 183

NIS (Network Information Services)

推奨されないサポート 183

NIS+ (Network Information Services Plus)

推奨されないサポート 183

numsegs 構成パラメーター

推奨されない機能 140

NUM\_ESTORE\_SEGS 構成パラメーター

廃止されたサポート 185

num\_estore\_segs 構成パラメーター

廃止されたサポート 140

num\_initagents 構成パラメーター

変更 140

num\_initfenced 構成パラメーター

変更 140

num\_poolagents 構成パラメーター

デフォルト値の変更 140

NVL スカラー関数

アプリケーションの移植性 95

## O

OLAP (Online Analytical Processing)

関数

機能拡張 56

OVERLAY スカラー関数

Unicode 可変文字サイズ・サポート 120, 132

## P

PDO\_IBM PHP 拡張

概要 70

パッケージ 70

PD\_GET\_DIAG\_HIST 関数 124

Perl ドライバー

マルチバイト文字サポート 72

pureXML サポート 72

PHP 拡張

概要 70

パッケージ 70

physical\_page\_maps モニター・エレメント

廃止されたサポート 186

piActionString データ構造

推奨されない機能 182

pkg\_cache\_size\_top モニター・エレメント  
 推奨されない機能 177

pool\_data\_from\_estore モニター・エレメント  
 廃止されたサポート 185

pool\_data\_to\_estore モニター・エレメント  
 廃止されたサポート 185

pool\_index\_from\_estore モニター・エレメント  
 廃止されたサポート 185

pool\_index\_to\_estore モニター・エレメント  
 廃止されたサポート 185

POWER 関数  
 SYSFUN 関数の SYSIBM パージョン 163

PreparedStatement オブジェクト  
 バッチ処理エラー 158

priv\_mem\_thresh 構成パラメーター  
 廃止されたサポート 140

priv\_workspace\_num\_overflows モニター・エレメント  
 推奨されない機能 177

priv\_workspace\_section\_inserts モニター・エレメント  
 推奨されない機能 177

priv\_workspace\_section\_lookups モニター・エレメント  
 推奨されない機能 177

priv\_workspace\_size\_top モニター・エレメント  
 推奨されない機能 177

ps コマンド  
 出力の変更 170

pureXML  
 機能拡張のサマリー 9, 61  
 ストレージ・スペースの縮小 65  
 チェック制約  
 機能拡張 63  
 データ  
 更新 61  
 非 Unicode データベース・ストレージ 65  
 ロード 62  
 XSLT を使用した変換 63  
 トリガー処理の機能拡張 63  
 発行関数 67  
 パフォーマンス向上 62, 65  
 パラメーターの引き渡しの改善 64  
 分解の機能拡張 68  
 文書の妥当性検査の機能拡張 63  
 ロード・ユーティリティ 62  
 Perl ドライバー 72  
 VALIDATED 述部 63

## Q

Query Patroller  
 Windows Vista のサポート 111

query\_heap\_sz 構成パラメーター  
 推奨されない機能 140, 155

## R

RADIANS 関数  
 SYSFUN 関数の SYSIBM パージョン 163

REDISTRIBUTE DATABASE PARTITION GROUP コマンド  
 機能拡張 33  
 メッセージ・ファイルの変更 165

REFRESH TABLE ステートメント  
 機能拡張 59  
 ロック・タイプの変更 139

ResultSetMetaData.getColumnLabel  
 JDBC 4.0 で変更された値 157

ResultSetMetaData.getColumnName  
 JDBC 4.0 で変更された値 157

REVOKE EXEMPTION ステートメント  
 機能拡張 52

REVOKE SECURITY LABEL ステートメント  
 機能拡張 52

RIGHT スカラー関数  
 Unicode 可変文字サイズ・サポート 120, 132

ROLLFORWARD コマンド  
 最小リカバリ時間の機能拡張 103

Ruby on Rails  
 サポートの概要 71

RUNSTATS ユーティリティ  
 UNSET PROFILE オプション 39

## S

SCHEMA 特殊レジスター  
 戻り値の変更 161

SECADM (セキュリティ管理者) 権限  
 データベース監査 130

SET INTEGRITY ステートメント  
 機能拡張 59  
 ロック・タイプの変更 139

SET 変数ステートメント  
 機能拡張 75

shr\_workspace\_num\_overflows モニター・エレメント  
 推奨されない機能 177

shr\_workspace\_section\_inserts モニター・エレメント  
 推奨されない機能 177

shr\_workspace\_section\_lookups モニター・エレメント  
 推奨されない機能 177

shr\_workspace\_size\_top モニター・エレメント  
 推奨されない機能 177

SIGN 関数  
 SYSFUN 関数の SYSIBM パージョン 163

SNAPBP 管理ビュー  
 変更 186

SNAP\_GET\_APPL\_INFO 表関数  
 推奨されない機能 132

SNAP\_GET\_APPL 表関数  
 推奨されない機能 132

SNAP\_GET\_BP 表関数  
 推奨されない機能 132

SNAP\_GET\_BP 表関数 (続き)  
変更 186

SNAP\_GET\_DBM 表関数  
推奨されない機能 132

SNAP\_GET\_DB\_V91 表関数  
推奨されない機能 132

SNAP\_GET\_DYN\_SQL\_V91 表関数  
推奨されない機能 132

sortheap 構成パラメーター  
変更 140

SQL  
管理ビューの追加 36  
管理ルーチンの追加 36  
XQuery パラメーターの引き渡しの改善 64

SQL ステートメント  
ヘルプを表示する 203  
ラベル・ベースのアクセス制御 (LBAC) の機能拡張 52

ALTER BUFFERPOOL  
ESTORE の変更 185

ALTER TABLESPACE  
スペース再利用の機能拡張 38

CREATE BUFFERPOOL  
ESTORE の変更 185

CREATE DATABASE  
NO FILE SYSTEM CACHING デフォルト 152

CREATE INDEX  
パフォーマンス向上 56

CREATE TABLESPACE  
NO FILE SYSTEM CACHING デフォルト 152

REFRESH TABLE  
機能拡張 59  
ロック・タイプの変更 139

SET INTEGRITY  
機能拡張 59  
ロック・タイプの変更 139

SET 変数  
機能拡張 75

SQLJ  
機能拡張 78, 84

SQLOGCTL.LFH ファイル  
二重コピー 101, 179

sqludau API  
推奨されない機能 176

SQRT 関数  
SYSFUN 関数の SYSIBM バージョン 163

SSV (単一システム・ビュー)  
バックアップ 102

stat\_heap\_sz 構成パラメーター  
機能拡張 30, 155  
デフォルト値の変更 140

stmheap 構成パラメーター  
機能拡張 30, 155  
変更されたデフォルト値 140

STRIP スカラー関数  
Unicode 可変文字サイズ・サポート 120, 132

SYSCAT ビュー  
追加 132  
変更 132

## T

TRIM スカラー関数  
Unicode 可変文字サイズ・サポート 120, 132

## U

Unicode  
ストリング・リテラル 119  
デフォルト・コード・ページ 129  
Big5-HKSCS 変換 120

Unicode データベース  
データベース  
言語認識照合のサポート 119

UPDATE XMLSCHEMA コマンド  
概要 66

UPPER スカラー関数  
ロケール依存 121

upper-case 関数  
ロケール・サポートの機能拡張 66

USER 特殊レジスター  
戻り値の変更 161

userexit 構成パラメーター  
推奨されない機能 140

## V

V9.1 フィックスパックの変更、DB2 Connect バージョン 9.5  
の使用に影響する事項 21

Visual Explain  
チュートリアル 207

Visual Studio 2005 用の IBM Database Add-in  
機能拡張 72

## W

Web Object Runtime Framework (WORF)  
推奨されない機能 182

WebSphere Federation Server  
機能拡張のサマリー 113

Windows  
マイグレーション  
インターフェースの変更 138

Windows Vista  
拡張セキュリティ要件 153  
機能拡張 111  
ファイル・ロケーションの変更 154

WITH HOLD カーソル  
フェデレーション・サポート 113

wlm\_collect\_int 構成パラメーター  
概要 140

## X

### XML

- 機能拡張のサマリー 9, 61
- 処理の改善 62
- ストレージ・スペースの縮小 65
- チェック制約
  - 機能拡張 63
- データ
  - 更新 61
  - 非 Unicode データベース・ストレージ 65
  - ロード 62
  - XSLT を使用した変換 63
- トリガー処理の機能拡張 63
- 発行関数 67
- パフォーマンス向上 65
- 文書の妥当性検査の機能拡張 63
- ロード・ユーティリティー 62

### XML Extender

- 推奨されない機能 181

### XML スキーマ

- 更新 66
- 再帰の機能拡張 68
- 挿入の順序の機能拡張 68

### XML データ・タイプ

- フェデレーション・サポート 113

### XML の分解

- 再帰的スキーマの機能拡張 68
- 挿入の順序の機能拡張 68

### XMLGROUP 関数

- 概要 67

### XMLROW 関数

- 概要 67

### XMLTRANSFORM 関数

- 概要 67

### XQuery

- キャスト可能式 67
- 更新式 61
- SQL パラメーターの引き渡しの改善 64
- XML データの更新 61

### XSLTRANSFORM 関数

- 概要 63

### XSR\_UPDATE ストアード・プロシージャ

- 概要 66









Printed in Japan

SC88-4445-00



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

Spine information:

**DB2 Version 9.5 for Linux, UNIX, and Windows**

**新機能**

