DB2 Version 9.5
for Linux, UNIX, and Windows

IBM

What's New
Updated March, 2008

DB2 Version 9.5
for Linux, UNIX, and Windows

IBM

What's New
Updated March, 2008

Edition Notice

This document contains proprietary information of IBM. It is provided under a license agreement and is protected by copyright law. The information contained in this publication does not include any product warranties, and any statements provided in this manual should not be interpreted as such.

You can order IBM publications online or through your local IBM representative.

- To order publications online, go to the IBM Publications Center at www.ibm.com/shop/publications/order
- To find your local IBM representative, go to the IBM Directory of Worldwide Contacts at www.ibm.com/planetwide

To order DB2 publications from DB2 Marketing and Sales in the United States or Canada, call 1-800-IBM-4YOU (426-4968).

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

About this book

This book provides information about the new and changed functionality included in the Version 9.5 release of the DB2® Database for Linux®, UNIX®, and Windows® and DB2 Connect™ products.

Who should use this book

This book is for database administrators, application programmers, and other DB2 database users who want to quickly find out what enhancements are available in DB2 Version 9.5 for Linux, UNIX, and Windows and in DB2 Connect Version 9.5 and what differences exist between Version 9.5 and Version 9.1 of those products.

This book provides overview information and does not contain detailed instructions for using the features described. To get additional information, use the references that are provided.

For information about the features and enhancements introduced in Version 9.5, read Part 1, "New features and functionality," on page 1.

For information about the changed, deprecated, or discontinued functionality in Version 9.5, read Part 2, "What's changed," on page 121. This information points out important changes that you need to know before using Version 9.5.

For DB2 Connect information, read Chapter 2, "DB2 Connect enhancements summary," on page 19.

If you are a Version 9.1 user, review Chapter 19, "DB2 Version 9.1 fix pack changes that affect DB2 Version 9.5 usage," on page 179 for a list of the changes introduced in Fix Pack 3 (and earlier fix packs) that are applicable also to Version 9.5 but are not described in other sections of this book.

This book was updated since it was first published to include details about Fix Pack 1. For a list of important changes included in that fix pack, see "DB2 Version 9.5 for Linux, UNIX, and Windows fix pack summary" on page 16.

How this book is structured

The following topics are covered:

Part 1: New features and functionality

Chapter 1, "Highlights of DB2 Version 9.5," on page 3
    This chapter provides overviews of the most important new features and enhancements included in DB2 Version 9.5 and in DB2 Version 9.5 fix packs.

Chapter 2, "DB2 Connect enhancements summary," on page 19
    This chapter describes the enhancements and changes in DB2 Version 9.5 that affect the functionality of DB2 Connect.

Chapter 3, "Product packaging enhancements," on page 25
    This chapter describes the product packaging changes introduced in Version 9.5.

Part 2: What's changed

functions or features that are supported but are no longer recommended and might be removed in a future release.

Chapter 18, "Discontinued functionality," on page 173
This chapter lists features and functionality that are unsupported in Version 9.5.

Chapter 19, "DB2 Version 9.1 fix pack changes that affect DB2 Version 9.5 usage," on page 179
This chapter describes features and functionality added or changed as part of Version 9.1 Fix Pack 3 (and earlier fix packs) that are applicable also to Version 9.5 but are not described in other sections of this book.

Part 3: Appendixes

File system caching configurations
This appendix provides supplementary information about file system caching configurations supported in Version 9.5.

Overview of the DB2 technical information
This appendix contains information about accessing and using the latest documentation for your DB2 database systems.

Notices
This appendix contains the legal requirements and limitations related to using the DB2 database product and its documentation.

Highlighting conventions

Topics that are associated with a specific fix pack include an "FPx" prefix at the beginning of the topic title, where *x* represents a fix pack level.

The following highlighting conventions are used in this book.

Bold	Indicates commands, keywords, and other items whose names are predefined by the system. Commands written in uppercase are CLP commands, whereas commands written in lowercase are system commands.
Italics	Indicates one of the following: • Names or values (variables) that must be supplied by the user • General emphasis • The introduction of a new term • A reference to another source of information
`Monospace`	Indicates one of the following: • Files and directories • Information that you are instructed to type at a command prompt or in a window • Examples of specific data values • Examples of text similar to what might be displayed by the system • Examples of system messages • Samples of programming code

Part 1. New features and functionality

This section describes the new features and functionality available with DB2 Version 9.5 for Linux, UNIX, and Windows.

Topics that are associated with a specific fix pack include an ″FPx″ prefix at the beginning of the topic title, where x represents a fix pack level.

Chapter 1, "Highlights of DB2 Version 9.5," on page 3
This chapter provides overviews of the most important new features and enhancements included in DB2 Version 9.5 and in DB2 Version 9.5 fix packs.

Chapter 2, "DB2 Connect enhancements summary," on page 19
This chapter describes the enhancements and changes in DB2 Version 9.5 that affect the functionality of DB2 Connect.

Chapter 3, "Product packaging enhancements," on page 25
This chapter describes the product packaging changes introduced in Version 9.5.

Chapter 4, "Manageability enhancements," on page 27
This chapter describes the new features and enhancements that help you spend less time managing your databases.

Chapter 5, "Workload management enhancements," on page 41
This chapter describes the new workload management features that extend the existing workload management capabilities provided in previous releases.

Chapter 6, "Security enhancements," on page 47
This chapter describes the new features and enhancements that help you protect and manage your sensitive data.

Chapter 7, "Performance enhancements," on page 53
This chapter describes the new features and enhancements that help ensure the highest performance when you access and update data.

Chapter 8, "pureXML enhancements," on page 59
This chapter describes the new pureXML features and enhancements.

Chapter 9, "Application development enhancements," on page 67
This chapter describes the new features and enhancements that simplify application development, improve application portability, and ease application deployment.

Chapter 10, "High availability, backup, logging, and recovery enhancements," on page 93
This chapter describes the new features and enhancements that help ensure that your data remains available for your users.

Chapter 11, "Installation, migration, and fix pack enhancements," on page 101
This chapter describes the new features and enhancements that make it faster to deploy DB2 products and easier to maintain them.

Chapter 12, "Federation enhancements," on page 107
This chapter describes the new features and enhancements for federated databases.

Chapter 13, "Replication enhancements," on page 111
> This chapter describes the new features and enhancements for SQL replication.

Chapter 14, "National language enhancements," on page 113
> This chapter describes the new features and enhancements that make it easier to work with data and database applications that work with multiple national languages.

Chapter 15, "Troubleshooting and problem determination enhancements," on page 117
> This chapter describes the new features and enhancements that you can use to generate diagnostic information when you encounter problems.

Chapter 1. Highlights of DB2 Version 9.5

DB2 Version 9.5 for Linux, UNIX, and Windows delivers important new features and enhancements that address the needs of your business, whether those needs are integrating business data from across your organization, reducing costs, creating business value, or providing a secure and resilient system for the valuable information assets of your company.

Manage your business, not your database

Version 9.5 introduces autonomic enhancements that reduce the time required to administer and tune your data servers and installation enhancements that allow you to set up and deploy your applications more quickly.

Version 9.5 includes the following key manageability features:

Data compression enhancements
> Data compression dictionaries can be automatically created during data population operations on tables for which you have defined the COMPRESS attribute. You no longer have to worry about when to create the dictionary, and it is easier to create packaged applications that use compression. For more information, see "Data compression has been enhanced" on page 31.

New workload management capabilities
> In Version 9.5, a new, comprehensive set of workload management features helps you to identify, manage, and monitor data server workloads. These features provide the first workload management solution truly integrated into the DB2 data server. Support for identity assertion enables you to provide workload management for individual users or groups in a multi-tier application environment.
>
> For more information, see "Workload management improvements provide better control" on page 41 and "Trusted contexts enhance security" on page 47.

Multithreaded architecture extensions
> DB2 data servers now use a multithreaded architecture on all platforms, which improves performance and simplifies configuration and optimization. Also, simplifications to memory management eliminate most agent-level configuration parameters and automate the rest. This use of a consistent threading architecture across all operating systems reduces the overall complexity and maintenance of your data server. Prior to Version 9.5, IBM® provided a multithreaded architecture on Windows operating systems only. Version 9.5 provides the benefits of a multithreaded architecture on other operating systems. For more information, see "Simplified multithreaded architecture reduces the Total Cost of Ownership (TCO)" on page 28.

Deployment improvements
> Deployment improvements simplify the process of installing and maintaining your DB2 data server. Version 9.5 includes the following key features:
> - The new IBM Data Server Driver for ODBC, CLI, and .NET client simplifies application deployment on Windows platforms. This client,

which has a small footprint, is designed to be redistributed by independent software vendors (ISVs) and to be used for application distribution in mass deployment scenarios typical of large enterprises. For more information, see "New DB2 client product simplifies deployment (Windows)" on page 25.

- Two previously manual steps required after applying fix packs, the running of the db2iupdt and dasupdt commands, are now automated. In addition, binding occurs automatically at the first connection. For more information, see "Fix pack post-installation tasks are automated (Linux and UNIX)" on page 102.
- Non-root users can now perform administrative tasks on Linux and UNIX operating systems. Tasks that non-root users can perform include installing, applying or rolling back fix packs, configuring instances, adding new features, and uninstalling. For more information, see "Non-root users can install and configure DB2 products (Linux and UNIX)" on page 102.

Easier management of partitioned database systems

Enhancements available in Version 9.5 ensure that partitioned database systems are easier to manage. Version 9.5 includes the following partitioned database system features:

- There is a single view of all database configuration elements across multiple partitions. With this new functionality, you can update or reset a database configuration across all database partitions by issuing a single SQL statement or a single administration command from any partition on which the database resides. For more information, see "Database configuration across multiple partitions has been simplified" on page 29.
- The BACKUP DATABASE command can now back up all partitions of a multi-partition database at once. For more information, see "Multiple database partitions can be backed up and restored at once with single system view backup" on page 97.

Automatic storage management enhancements

Automatic storage automatically grows the size of your database across disk and file systems. It eliminates the need to manage storage containers while taking advantage of the performance and flexibility of database-managed space. Version 9.5 introduces table space handling enhancements that ensure that you are able to reduce the size of a table space to an accurate High Water Mark (HWM). This allows you to reclaim unused space automatically. For more information, see "Table spaces use space more efficiently" on page 38.

Additional automatic configuration parameters

Version 9.5 includes more tuning parameters that the data server automatically handles, without requiring you to stop and restart your instance or database. For more information about new configuration parameters, see "Some database configuration parameters have changed" on page 139 and "Some database manager configuration parameters have changed" on page 123. For more information about enhancements to parameters that control memory allocation, see "More configuration parameters can be set to AUTOMATIC and configured dynamically" on page 31.

Automated maintenance improvements

If you want to take advantage of automatic maintenance but need to maintain tight control over the process and policies, you can use the new system stored procedures for collecting automated maintenance

configuration information (SYSPROC.AUTOMAINT_GET_POLICY and
SYSPROC.AUTOMAINT_GET_POLICYFILE) and configuring automated
maintenance (SYSPROC.AUTOMAINT_SET_POLICY and
SYSPROC.AUTOMAINT_SET_POLICYFILE). You can use these procedures
to configure and collect information about automated maintenance for the
following areas:

- Maintenance windows
- Automatic backups
- Automatic table and index reorganizations
- Automatic table RUNSTATS operations

For more information, see "New system stored procedures simplify
automated maintenance policy configuration" on page 93.

Related enhancements from Version 9.1

In Version 9.1, IBM introduced the following manageability and installation
enhancements that make it easier to install and maintain your database:

- Simpler memory management using adaptive, self-tuning memory allocation.
  Self-tuning memory provides a configuration that is dynamic and responsive to
  significant changes in workload characteristics.
- Automatic statistics collection enabled by default when you create databases.
  With automatic statistics collection enabled, the DB2 database product
  automatically runs the RUNSTATS utility in the background to ensure that the
  correct statistics are collected and maintained.
- Automatic storage support for partitioned databases.
- The ability to change some attributes of tables without having to drop and
  re-create the tables.
- New policy options that provide you with more table and index reorganization
  capabilities.
- The ability to copy database schemas and create model schemas. After you
  establish a model schema, you can use it as a template for creating new versions
  of the schema.
- New administrative SQL routines and views. The administrative routines and
  views provide a primary, easy-to-use programmatic interface to administer the
  DB2 database product through SQL.
- Dynamic fast communication manager (FCM) buffers and new configuration
  parameters that can be tuned automatically by the DB2 database manager.
- Easier management of product licenses using the License Center and the db2licm
  command.
- The ability to install multiple DB2 versions and fix packs on the same computer.
- New response file keywords that allow you to set up DB2 database products
  without interaction.
- Licensing changes for the DB2 Runtime Client, allowing you to freely distribute
  it.

  Related concepts

  "Fast communications manager (Linux and UNIX)" in Partitioning and
  Clustering Guide

  "Fast communications manager (Windows)" in Partitioning and Clustering
  Guide

  "Response file installation basics" in Quick Beginnings for DB2 Servers

"Automatic statistics collection" in Data Servers, Databases, and Database Objects Guide

"Self-tuning memory" in Data Servers, Databases, and Database Objects Guide

"IBM data server client types" in Quick Beginnings for IBM Data Server Clients

"Automatic reorganization" in Tuning Database Performance

"Automatic storage" in Data Servers, Databases, and Database Objects Guide

Related tasks

"Copying schemas" in Data Movement Utilities Guide and Reference

Related reference

"ALTER TABLE statement" in SQL Reference, Volume 2

"Supported administrative SQL routines and views" in Administrative Routines and Views

"Multiple DB2 copies overview" in Data Servers, Databases, and Database Objects Guide

Increased performance and scalability

Version 9.5 introduces performance and scalability enhancements to help you achieve the highest performance when accessing and updating large volumes of data. Improvements in performance and scalability continue to make the DB2 data server an industrial-strength data server solution that is suitable for any size of company.

Version 9.5 includes the following key performance and scalability features:

Query performance and automatic statistics collection improvements
Enhancements to the query optimizer and automatic statistics collection improve the efficiency and performance of your queries. Version 9.5 introduces the following enhancements:

- Real-time statistics gathering. This ensures that table statistics are available whenever they are needed to optimize and run a query. When you submit a query to the compiler, the optimizer determines whether real-time statistics need to be collected before the query is compiled and run. The compiler then uses any statistics that are collected to generate the best access plan for the query. The real-time, self-correcting statistics enhancement ensures that just enough statistics are updated for the optimizer to generate the best access plan for a query. For more information, see "Real-time statistics gathering ensures that the latest statistics are used for optimization" on page 27.

- Automatically refreshed nickname statistics. Nickname statistics are kept up-to-date by running the nickname statistics (NNSTAT) stored procedure. Access to the most current statistics enables the optimizer on the federated server to make informed choices of query plans that improve performance. For more information, see "Configuration has been enhanced for federation" on page 109.

- Complex queries have been optimized. For more information, see "Query optimizer has been enhanced" on page 56.

LOB management performance improvements
Enhancements to LOB management improve the performance of queries that return LOB data. Version 9.5 includes the following enhancements:

- Blocking of row data that contains references to LOB data types. When a result set contains LOB data, multiple rows of data can be blocked together and returned as a result set to the client for a single cursor request.
- Support for the Dynamic Data Format (also known as progressive streaming). This allows the server to return LOB values efficiently. DB2 clients automatically take advantage of the Dynamic Data Format when using progressive streaming to retrieve LOB values.

For more information, see "Queries that reference LOB columns run faster" on page 53.

Faster multidimensional clustered table (MDC) rollout
You can now defer the cleanup of record ID (RID) indexes until after a rollout delete on an MDC table is complete. Deferred cleanup of RID indexes significantly improves the speed of delete operations that occur on dimensional boundaries. For more information, see "MDC rollout deletions are faster with the deferred index cleanup option" on page 54.

Increased concurrency
Support for optimistic locking minimizes the time a resource is unavailable by limiting the time that lock are held while ensuring data integrity. Using the optimistic locking protocol, the server releases locks immediately after reading a row. When the row is updated at a later point in time, the server validates that the row had remained unchanged in the meantime. For more information, see "Optimistic concurrency control and update detection enhancements provide a scalable locking alternative" on page 53.

Related enhancements from Version 9.1

In Version 9.1, IBM introduced several performance and scalability enhancements that make it easier to manage large volumes of data. These enhancements introduced in Version 9.1 include:
- The ability to compress table data objects using data row compression
- Improved access plans for queries using statistical views
- Faster data-loading capabilities using customized scripts or programs
- Enhanced query execution for materialized query tables
- Larger RIDs, which allow more data pages per object and more records per page
- Index keys that can include up to 64 columns and that are up to 8 KB in size

Related concepts

"Database managed space" in Data Servers, Databases, and Database Objects Guide

"Materialized query tables" in Tuning Database Performance

"Statistical views" in Tuning Database Performance

"Moving data using a customized application (user exit)" in Data Movement Utilities Guide and Reference

"Data row compression" in Data Servers, Databases, and Database Objects Guide

Related reference

"SQL and XML limits" in Data Servers, Databases, and Database Objects Guide

Information as a service

In Version 9.1, IBM introduced pureXML support, which transformed the DB2 database system into a hybrid relational and XML data server. Version 9.5 builds on these enhancements and extends the support to make your XML data processing even more flexible, faster, and even more reliable.

Version 9.5 includes the following key pureXML features:

High-speed XML data load
> The high-performance load utility enables you to insert large amounts of XML data into DB2 tables quickly and efficiently. For more information, see "Load utility support for pureXML has been added" on page 60.

pureXML performance improvements
> Enhancements to the pureXML feature reduce execution times and, in some cases, reduce resource usage for applications that process XML data. The performance enhancements include improvements to data manipulation in both SQL/XML and XQuery, indexes over XML data creation, the query compiler and optimization, and XML document navigation. For more information, see "pureXML application processing performance has been improved" on page 60.

Sub-document updates for improved query performance
> Enhancements to DB2 XQuery provide more support for structural modifications to XML documents. New XQuery updating expressions improve the efficiency of sub-document updates by enabling you to modify portions of an existing XML document instead of constructing a new one. XQuery updating expressions allow you to delete, insert, replace, or rename nodes in an XML document. For more information, see "XQuery updating expressions allow portions of XML documents to be modified" on page 59.

XML support for integrity features
> Version 9.5 introduces XML support for the following integrity features:
> - Check constraints functionality allows you to specify additional options with constraints on an XML column to ensure the consistency of information before it is processed. For more information, see "Check constraints functionality has been extended" on page 60.
> - Trigger processing supports automatic validation of XML documents against registered XML schemas based on the current validation state of the documents. For more information, see "Trigger processing supports automatic validation of XML documents" on page 61.

Easier conversion of XML into HTML, plain text, and other formats
> Extensible Stylesheet Language Transformation (XSLT) is the most popular way of transforming XML. Enhancements to pureXML enable the flexible transformation of XML through built-in XSLT support. The new XSLTRANSFORM function converts XML documents that reside in a database into HTML, plain text, or other forms of XML. For more information, see "XSLT support enables transforming XML data into other formats" on page 61.

Schema compatibility management and evolution support
> Schemas evolve over time. Enhancements available in Version 9.5 ensure that you are able to validate both previously inserted and new XML documents against an evolved version of a registered schema. The UPDATE XMLSCHEMA command and the XSR_UPDATE stored

procedure allow you to modify an XML schema that you already registered in the XML schema repository. For more information, see "XML schemas can be updated without requiring re-validation of XML documents" on page 63.

Non-Unicode database support

The features of pureXML are now available in non-Unicode databases. New functionality manages code page conversion so that you no longer need a Unicode database. A new **enable_xmlchar** configuration parameter prevents possible character substitution from occurring as SQL string data is converted from the client code page into the database code page and then into Unicode for internal storage. For more information, see "Non-Unicode databases can store XML data" on page 62.

More agile development through SQL/XML and XQuery

Improvements to SQL/XML and XQuery ensure that you are able to leverage the power of both languages to create robust and efficient queries over XML data. Version 9.5 introduces the following improvements:

- Parameter passing is simplified and extended for both SQL/XML and XQuery to become more flexible. For more information, see "SQL/XML and XQuery parameter passing is more flexible" on page 62.
- New publishing functions are available for mapping relational data to XML. These functions require you to specify fewer options than you need to for existing SQL/XML publishing functions. For more information, see "Publishing functions are easier to use" on page 64.
- The syntax of many existing SQL/XML publishing functions is simplified.
- The XQuery language feature now includes support for using type casting, specifying locales when using the upper-case and lower-case functions, extracting date and time components, and adjusting time zones. For more information, see "XQuery castable expression supports testing the casting of values" on page 64, "XQuery upper-case and lower-case functions support locales" on page 64, and "XQuery functions extract components from and adjust dates and times" on page 64.

Decomposition enhancements for pureXML

Decomposition is enhanced for pureXML to support order of insertion and the registration of recursive schemas:

- New XML schema annotations allow you to specify a decomposition hierarchy to ensure that the contents of an XML document are inserted into the rows of a target table in a determinable order. This enhancement ensures that referential integrity constraints are respected during XML document shredding.
- You can now register XML schemas that contain recursion in the XML schema repository (XSR) and enable them for decomposition.

For more information, see "Annotated XML schema decomposition supports order of insertion and registration of recursive schemas" on page 65.

DB2 Text Search support

Available in Fix Pack 1, DB2 Text Search delivers an integrated and scalable search technology for DB2 databases that you can use for performing relational data, XQuery, and SQL/XML text searches and searches of other document formats. DB2 Text Search makes it easier than

ever to search within DB2 databases. For more information, see DB2 Text Search supports searching SQL and XML data.

Related enhancements from Version 9.1

In Version 9.1, IBM delivered the most powerful XML data server in the industry. The pureXML support handles XML as a new data type that is stored in a natural hierarchy - different from relational data. The seamless integration of XML with relational data speeds application development, improves search performance with highly optimized XML indexes, and is flexible because both SQL and XQuery can be used to query XML data.

Version 9.1 includes the following pureXML features:
- Integration with the DB2 database system, which includes support for the following functionality:
  - A new XML data type, which supports storing well-formed XML documents in their hierarchical forms within columns of a table
  - XQuery, a functional programming language for querying XML data
  - The XML data type in SQL statements and SQL/XML functions
  - Indexing of XML data
- Enhanced and new tools to access and manage XML data, which include the following ones:
  - The Developer Workbench, which supports XML functions, the XML data type, and XML schema registration
  - The DB2 Command Line Processor (CLP), which supports the XML data type
  - The Explain facility and Visual Explain GUI tool, which support SQL/XML functions and XQuery statements
- Application development support, which includes the following items:
  - XML support for programming languages, which enables applications to access and store both XML and relational data
  - XML support in SQL and external procedures, which enables XML data to be passed to SQL and external procedures by including parameters of data type XML in CREATE PROCEDURE parameter signatures

  Related concepts

  "Explain facility" in Tuning Database Performance

  "Visual Explain" in Visual Explain Tutorial

  "XML data type" in pureXML Guide

  "Querying XML data" in pureXML Guide

  "Indexing XML data" in pureXML Guide

  Related reference

  "CREATE PROCEDURE statement" in SQL Reference, Volume 2

  "Command line processor features" in Command Reference

  Related information

  "Introduction to XQuery" in XQuery Reference

Improved security and resilience

Version 9.5 provides new features and enhancements that contribute to providing a secure and resilient environment for your data.

IT security is a major concern for organizations today. It has become crucial to ensure that sensitive data is protected. You need the ability to efficiently manage system security, quickly analyze your security environment, and monitor access to data. Building on the enhancements introduced in Version 9.1, new security enhancements in Version 9.5 ensure that your sensitive data is even better protected.

Version 9.5 includes the following key security features:

Simplified security management with database roles
A *role* is a database object that groups together one or more privileges. The security administrator (who holds SECADM authority) can assign a role to users, groups, PUBLIC, other roles, or a trusted context. When a user becomes a member of a role, the user automatically obtains all of the privileges that are assigned to the role. When the security administrator revokes a user's membership in a role, the user automatically loses all of the privileges that are assigned to the role. Roles simplify the administration and management of privileges by enabling security administrators to control access to their databases in a way that mirrors the structure of their organizations (they can create roles in the database that map directly to the job functions in their organizations). For more information, see "Roles simplify privilege administration and management" on page 50.

Trusted contexts support
Trusted contexts provide a way of building much faster and more secure three-tier applications. The user's identity is always preserved for auditing and security purposes. When you need secure connections, trusted contexts improve performance because you do not need to get new connections. For more information, see "Trusted contexts enhance security" on page 47.

Label-based access control (LBAC) enhancements
LBAC provides finer control over data access by enabling you to specify access to individual rows and individual columns. The management of security labels and exemptions is now simplified. You no longer have to manage these credentials at the individual user level; you can now manage them at the group or role level as well. For more information, see "Label-based access control (LBAC) enhancements provide better security" on page 51.

Audit facility enhancements
Major enhancements to the audit facility for Version 9.5 provide better control over auditing and significantly improve its performance and ease of use. These enhancements include fine-grained configuration, new audit categories, separate instance and database logs, and new ways to customize the audit configuration. Sole responsibility for managing audits at the database level now lies with the security administrator (who holds SECADM authority). For more information, see "Audit facility performance and administration have been enhanced" on page 49.

Security administrator authority level enhancements
As mentioned previously, the security administrator can now manage trusted context objects, database roles, and audit policies. The security administrator is now able to create, alter, drop, or comment on any of these objects. For more information, see "Trusted contexts enhance security" on page 47, "Roles simplify privilege administration and management" on page 50, and "Audit facility performance and administration have been enhanced" on page 49.

Related enhancements from Version 9.1

In Version 9.1, IBM introduced several features designed to protect your sensitive data. These features introduced in Version 9.1 include:

- Support for data access control using LBAC
- Security plug-in modules that support authentication and group lookup using the Lightweight Directory Access Protocol (LDAP)
- A new security administrator authority level (SECADM), which provides greater control over access to information assets and improved reporting capabilities for monitoring access to sensitive data
- A new **RESTRICTIVE** option for the CREATE DATABASE command, which provides greater control over database privileges

   Related concepts

   "Security administration authority (SECADM)" in Database Security Guide

   "Label-based access control (LBAC) overview" in Database Security Guide

   "LDAP-based authentication and group lookup support" in Database Security Guide

   Related reference

   "CREATE DATABASE command" in Command Reference

High availability and data recovery

Version 9.5 introduces several enhancements to keep your critical database applications online and available. Improved autonomic features, increased flexibility, and reduced outage windows ensure that your applications remain running, at a minimum cost.

Version 9.5 includes the following key high availability and data recovery features:

Enhanced autonomic features

- Automated maintenance configuration is simplified. You can use four new system stored procedures to collect automated maintenance policy information and to configure an automated maintenance policy. For more information, see "New system stored procedures simplify automated maintenance policy configuration" on page 93.
- Recovery object management is automated. You can now configure the DB2 database manager to automatically delete backup images, load copy images, and old log files that are no longer needed for recovery. For more information, see "Recovery object management is simplified through automated recovery object removal" on page 94.

Simpler and faster backup and restore operations

- The integration of storage management software such as IBM Tivoli® Storage Management (TSM) includes the following enhancements:
  - Storage management software integration is enabled. The new DB2 Advanced Copy Services (ACS) application programming interface (API) enables you to perform snapshot backup operations with your storage hardware. For more information, see "New DB2 Advanced Copy Services (ACS) API enables integration with storage hardware" on page 94.
  - Backup and restore operations are much faster with snapshot backups. When you run a snapshot backup or restore operation, your storage device performs the data copying part of the backup or

restore. Being able to use the storage device to perform the data copying makes the backup and restore operations much faster. For more information, see "Backing up and restoring data is faster with snapshot backups" on page 99.

- You can now back up and restore multiple database partitions at once using the new single system view (SSV) backup. For more information, see "Multiple database partitions can be backed up and restored at once with single system view backup" on page 97.
- Rollforward to minimum recovery time is simplified. You can use the **TO END OF BACKUP** clause with the ROLLFORWARD command or the DB2ROLLFORWARD_END_OF_BACKUP flag with the db2Rollforward API to roll forward all partitions in a partitioned database to the minimum recovery time. For more information, see "Rolling forward to minimum recovery time has been enabled" on page 98.

Improved failover and recovery capabilities

- Failover is more robust with the HADR peer window. You can use the new **hadr_peer_window** database configuration parameter to cause a DB2 High Availability Disaster Recovery (HADR) primary and standby databases pair to behave as though it is in the peer state if the primary database loses connection with the standby database. This feature can reduce the risk of data loss in the event of multiple or cascading failures. For more information, see "HADR peer window reduces the risk of data loss during cascading or multiple failures" on page 96.
- Recovery is more resilient with dual log control files. In Version 9.1, the database manager maintained one log control file: SQLOGCTL.LFH. In Version 9.5, the database manager maintains two copies of the log control file: SQLOGCTL.LFH.1 and SQLOGCTL.LFH.2. Having two copies of the log control file reduces the risk of losing data in the event of a failure. For more information, see "Dual log control files makes database recovery more resilient" on page 96.

Simplified clustered environment management

- The IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component, version 2.2, is now bundled with the IBM Data Server on Linux and AIX® operating systems. For more information, see "IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component is integrated into the DB2 installation (Linux and AIX)" on page 101.
- Cluster management software integration is enabled. The new DB2 cluster manager application programming interface (API) enables you to use IBM Data Server cluster configuration tools such as the DB2 high availability instance configuration utility (db2haicu) to configure your clustered environment. For more information, see "Cluster management software integration enabled" on page 99.

Faster offline redistribution of partitioned databases

Starting in Version 9.5 Fix Pack 1, the following improvements to the REDISTRIBUTE DATABASE PARTITION GROUP command make system capacity growth scenarios more manageable and efficient:

- New command options (**TABLE** and **STATISTICS USE PROFILE**) improve the usability and control of data redistribution processing.
- The REDISTRIBUTE DATABASE PARTITION GROUP command can now perform a variety of operations, including table reorganization, index maintenance, reclamation of free disk space, and database statistics

collection. The integration of these operations within the command reduces the number of table scans that the database manager performs, which improves performance. Internal architecture enhancements also contribute to the overall performance of the end-to-end capacity growing operation.

- The active log space requirement for data movement activities is minimized when using the NOT ROLLFORWARD RECOVERABLE option. This means that the redistribute utility can run with very little active log space, which eliminates the need to break a single redistribute operation into multiple smaller ones.

For more information, see "Data redistribution enhancements reduce capacity growth costs and load-balancing tasks" on page 33.

More agile development

Version 9.5 provides new features and enhancements that simplify database application development, improve application portability, and ease application deployment.

Version 9.5 includes the following key application development features and enhancements:

Decimal floating-point data type support
Version 9.5 introduces DECFLOAT, a decimal floating-point data type that is useful in business applications (for example, financial applications) that deal with exact decimal values..DECFLOAT combines the accuracy of the DECIMAL data type with some of the performance advantages of the FLOAT data type, which is beneficial in applications where monetary values are being manipulated. For more information, see "Decimal floating-point data type improves the accuracy and performance of decimal data" on page 73.

Application portability improvements
Improved application portability in Version 9.5 increases compatibility with existing code and eases migration of applications that use databases from other vendors. Version 9.5 includes the following enhancements:

- Support for the ARRAY data type within procedures and applications that call procedures. For more information, see "Array support enhances application portability" on page 72.
- Support for global variables. A global variable is a named memory variable that you can access and modify through SQL statements. Version 9.5 supports created session global variables, which are associated with a specific session and contain a value that is unique to that session. For more information, see "Global variables improve data sharing between SQL statements" on page 71.
- Support for larger identifiers. The maximum lengths of many identifiers have increased to 128 bytes. For more information, see "Identifier length limits have increased" on page 67.
- Support for SQL dialects from other vendors. Version 9.5 includes toleration of the DECODE, NVL, LEAST, and GREATEST functions. For more information, see "New scalar functions simplify application porting" on page 91.

JDBC and SQLJ enhancements
Version 9.5 includes support for the functions in JDBC 4.0, JDBC 3.0, and

earlier specifications. For more information, see "JDBC 2.0 and JDBC 3.0 support has been enhanced" on page 74 and "JDBC 4.0 support has been added" on page 81.

Perl, PHP, and Ruby application development enhancements

Improved support for Perl, Hypertext Preprocessor (PHP) extensions, and the Ruby on Rails framework simplifies application development and improves access to data that is stored in DB2 databases. Version 9.5 includes the following enhancements:

- The DB2 Perl driver now supports pureXML and multibyte locales. These enhancements simplify application development by removing some of the programming logic that was previously required to store and retrieve XML data and to convert between character sets. For more information, see "Perl driver supports pureXML and multibyte characters" on page 69.

- Ruby on Rails driver and binaries are now part of the DB2 installation for a subset of platforms to allow rapid deployment; you no longer need to download the driver and binaries separately. For more information, see "Ruby on Rails framework support has been integrated into DB2 installation (Linux, AIX, and Windows)" on page 68

- The following PHP extensions are available as part of the DB2 installation on a subset of platforms:
  - IBM_DB2, an existing extension that provides direct access to data stored in your DB2 database by using the DB2 Call Level Interface (CLI) libraries.
  - PDO_IBM, a new extension that provides access to a DB2 database through the PHP Data Objects (PDO) interface.

  For more information, see "PHP extensions have been integrated into DB2 installation (Linux, AIX, and Windows)" on page 68.

Tool set enhancements

IBM Data Studio is a rich and robust user interface that you can use to perform database design, development, deployment, and management tasks. It replaces the DB2 Developer Workbench, shipped in Version 9.1. You can use IBM Data Studio to develop and test routines, deploy data-centric Web services, create and run SQL and XQuery queries, and develop database applications. For more information, see "Data server tools have been added" on page 27.

Related enhancements from Version 9.1

In Version 9.1, IBM introduced several features and improvements that support more agile development. These features and improvements introduced in Version 9.1 include:

- pureXML support, which includes the following features:
  - Application development support for pureXML
  - Support for the XQuery language
  - Support for XML in SQL statements and SQL/XML functions
  - Support for the XML type in SQL and external procedures
  - Annotated XML schema decomposition
- An enhanced DB2 Driver for JDBC and SQLJ, which includes the following features:

- Support for SQLJ statements that perform functions equivalent to most JDBC methods
- Support for many new data types
- New DB2-only methods to support trusted connections to DB2 for z/OS® database servers
- Heterogeneous pooling and connection reuse
- The Developer Workbench, which includes the following features:
  - Integrated stored procedure debugging capabilities
  - Support for developing SQLJ applications
  - Support for XML functions
  - Support for change management systems that enable you to share projects

  Related concepts

  "Annotated XML schema decomposition" in pureXML Guide

  "XML data type" in pureXML Guide

  "Querying XML data" in pureXML Guide

  "Indexing XML data" in pureXML Guide

  "Supported drivers for JDBC and SQLJ" in Getting Started with Database Application Development

  Related reference

  "CREATE PROCEDURE statement" in SQL Reference, Volume 2

  Related information

  "Introduction to XQuery" in XQuery Reference

DB2 Version 9.5 for Linux, UNIX, and Windows fix pack summary

Version 9.5 fix packs contain important changes that might affect your product usage.

If you did not apply the Version 9.5 fix packs or did not update your local Information Center since Version 9.5 was available, you should review the following topics to understand the technical changes included in the DB2 Version 9.5 fix packs. Fix packs are cumulative, they contain all of the changes and functionality shipped in previous fix packs.

Fix Pack 1

Fix Pack 1 includes the following changes to existing functionality:
- JDBC 2.0 and JDBC 3.0 support changes. For more information, see "JDBC 2.0 and JDBC 3.0 support has been enhanced" on page 74.
- New SQLSTATEs returned by the IBM Data Server Driver for JDBC and SQLJ Version 4.0. For more information, see "JDBC 4.0 support has been added" on page 81.

Fix Pack 1 includes the following enhancements:
- DB2 Text Search component. For more information, see "FP1: DB2 Text Search supports searching SQL and XML data" on page 25.
- The load utility now supports the ALLOW READ ACCESS option for tables containing XML columns. For more information, see "Load utility support for pureXML has been added" on page 60.

- Locale-sensitive UCA-based collations. For more information, see "FP1: Locale-sensitive UCA-based collations provide more options for ordering data" on page 115.
- Faster data redistribution options. For more information, see "Data redistribution enhancements reduce capacity growth costs and load-balancing tasks" on page 33.
- Solaris Operating System x64 support. For more information, see "FP1: Solaris x64 support has been added" on page 104.
- The DB2_KEEP_AS_AND_DMS_CONTAINERS_OPEN registry variable, which can improve query performance in DMS environments. For more information, see "Performance variables" in the *Data Servers, Databases, and Database Objects Guide*.
- The DB2_LOGGER_NON_BUFFERED_IO registry variable, which enables direct I/O on the log file system. For more information, see "Performance variables" in the *Data Servers, Databases, and Database Objects Guide*.
- The DB2_HADR_PEER_WAIT_LIMIT registry variable, which can improve logging performance in HADR environments. For more information, see "Miscellaneous variables" in the *Data Servers, Databases, and Database Objects Guide*.
- db2pd command enhancements. For more information, see "FP1: db2pd command output provides more information" on page 38.
- IBM Data Server Provider for .NET support of trusted context. For more information, see "FP1: IBM Data Server Provider for .NET supports trusted context" on page 91.
- A new sqlj4.zip package that includes JDBC 4 functions. For more information, see "JDBC and SQLJ support has been enhanced" on page 74.

Chapter 2. DB2 Connect enhancements summary

DB2 Connect provides fast and robust connectivity to IBM mainframe databases for e-business and other applications running on Linux, UNIX, and Windows operating systems. Version 9.5 enhancements and changes affect the functionality and capabilities of DB2 Connect.

IBM's DB2 for i5/OS®, DB2 for z/OS, and DB2 Server for VSE & VM continue to be the systems of choice for managing most critical data for the largest organizations in the world. Although these host and i5/OS database products manage the data, there is a great demand to integrate this data with applications running on Linux, UNIX, and Windows operating systems.

DB2 Connect has several connection solutions, including DB2 Connect Personal Edition and a number of DB2 Connect server products. A DB2 Connect server is a connectivity server that concentrates and manages connections from multiple desktop clients and Web applications to DB2 database servers running on host or System i™ systems.

DB2 Connect servers enable local and remote client applications to create, update, and manage DB2 databases and host systems using:
- Structured Query Language (SQL)
- DB2 application programming interfaces (APIs)
- Open Database Connectivity (ODBC)
- Java™ Database Connectivity (JDBC)
- Structured Query Language for Java (SQLJ)
- DB2 Call Level Interface (CLI)
- Microsoft® ActiveX Data Objects .NET (ADO .NET)

The following Version 9.5 enhancements and changes affect the functionality and capabilities of DB2 Connect.

Product packaging enhancements
- "New DB2 client product simplifies deployment (Windows)" on page 25
- "Component names have changed" on page 26
- "FP1: DB2 Connect products added (Solaris x64)" on page 22

Application development enhancements
- "PHP extensions have been integrated into DB2 installation (Linux, AIX, and Windows)" on page 68
- "Ruby on Rails framework support has been integrated into DB2 installation (Linux, AIX, and Windows)" on page 68
- "Perl driver supports pureXML and multibyte characters" on page 69
- "IBM Database Add-Ins for Visual Studio 2005 has been enhanced" on page 70
- "Decimal floating-point data type improves the accuracy and performance of decimal data" on page 73
- "JDBC and SQLJ support has been enhanced" on page 74
- "JDBC 2.0 and JDBC 3.0 support has been enhanced" on page 74
- "JDBC 4.0 support has been added" on page 81

 19

- "IBM Data Server Provider for .NET supports IBM Informix Dynamic Server, IBM UniData, and IBM UniVerse" on page 86
- "CLIENT APPLNAME special register is automatically set by the CLP" on page 88
- "FP1: IBM Data Server Provider for .NET supports trusted context" on page 91

Federation enhancements

- "Application development has been enhanced for federation" on page 107
- "Security has been enhanced for federation" on page 107
- "Configuration has been enhanced for federation" on page 109

Replication enhancements

- "New CCD target type avoids join of UOW and CD tables" on page 111
- "DECFLOAT data type is supported for replication" on page 111

Installation, migration, and fix pack enhancements

- "Fix pack post-installation tasks are automated (Linux and UNIX)" on page 102
- "Non-root users can install and configure DB2 products (Linux and UNIX)" on page 102
- "Common fix pack simplifies server product updates" on page 101
- "New response file keywords have been added" on page 103
- "Non-DB2 instance merge modules have been added" on page 103
- "IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component is integrated into the DB2 installation (Linux and AIX)" on page 101
- "Windows Vista deployment and usage is easier" on page 104
- "Some database manager configuration parameters have changed" on page 123
- "Header files are no longer installed by default" on page 147

Troubleshooting and problem determination enhancements

- "Data collection tool tracks unexpected errors" on page 117
- "SQL administrative routine has been added for logging facilities" on page 117
- "Storage keys detect memory access problems" on page 118
- "Online data consistency checking is improved" on page 118
- "Index data consistency problem toleration is higher" on page 118
- "Database resilience is improved during unexpected errors" on page 119
- "FP1: db2pd command output provides more information" on page 38

Application development changes

- "ResultSetMetaData returns different values for the IBM Data Server Driver for JDBC and SQLJ Version 4.0" on page 148
- "Batch updates with automatically generated keys cause an SQLException" on page 149
- "Columns and application buffers require larger default values" on page 152
- "Some CLI/ODBC applications can consume more memory" on page 152
- "Command Line Processor (CLP) output has changed" on page 155
- "db2Load and db2Import parameters have been changed to support longer identifiers" on page 152

- "Identifiers that are too long result in errors and warnings being returned earlier" on page 153
- "Downlevel utilities and APIs might not handle longer identifiers correctly" on page 153
- "Unqualified SYSFUN functions might return SYSIBM error messages" on page 154
- "Special registers are longer" on page 154

CLP and system command changes

- "Customized callout scripts are searched for (Linux and UNIX)" on page 159
- "Listing of OS processes and threads has changed (Linux and UNIX)" on page 159
- Chapter 19, "DB2 Version 9.1 fix pack changes that affect DB2 Version 9.5 usage," on page 179

Deprecated functionality

- "Static data stream snapshot output is deprecated" on page 170

Discontinued functionality

- "CLISchema CLI keyword is discontinued" on page 177

Version 9.5 fix pack summary

DB2 Version 9.5 fix packs includes important changes to existing features and additional features that might affect DB2 Connect usage.

If you did not apply the Version 9.5 fix packs or did not update your local Information Center since Version 9.5 was available, you should review the following topics to understand the technical changes included in the DB2 Version 9.5 fix packs that might affect DB2 Connect. Fix packs are cumulative, they contain all of the changes and functionality shipped in previous fix packs.

Fix Pack 1

Fix Pack 1 includes the following changes to existing functionality:

- JDBC 2.0 and JDBC 3.0 support changes. For more information, see "JDBC 2.0 and JDBC 3.0 support has been enhanced" on page 74.
- New SQLSTATEs returned by the IBM Data Server Driver for JDBC and SQLJ Version 4.0. For more information, see "JDBC 4.0 support has been added" on page 81.

Fix Pack 1 includes the following enhancements:

- Solaris Operating System x64 support. For more information, see "FP1: DB2 Connect products added (Solaris x64)" on page 22.
- db2pd command enhancements. For more information, see "FP1: db2pd command output provides more information" on page 38.
- IBM Data Server Provider for .NET support of trusted context. For more information, see "FP1: IBM Data Server Provider for .NET supports trusted context" on page 91.
- A new sqlj4.zip package that includes JDBC 4 functions. For more information, see "JDBC and SQLJ support has been enhanced" on page 74.

FP1: DB2 Connect products added (Solaris x64)

You can install DB2 Connect Version 9.5 Fix Pack 1 (and higher) products on the Solaris Operating Environment (x64 architecture).

This includes support for DB2 Connect Personal Edition and all of the DB2 Connect server products.

Related reference

"DB2 Connect product offerings" in Quick Beginnings for DB2 Connect Servers

"Installation requirements for DB2 Connect products (Solaris Operating Environment)" in Quick Beginnings for DB2 Connect Servers

DB2 Version 9.1 fix pack changes that affect DB2 Connect Version 9.5 usage

Version 9.1 Fix Pack 3 (and earlier) include changes to features and functionality that might affect your usage of DB2 Connect Version 9.5.

Details

If you did not apply the Version 9.1 Fix Pack 3 or earlier fix packs, or did not update your local Information Center since Version 9.1 was available, you might not be aware of all the changes that could affect your usage of DB2 Version 9.5.

Resolution

Review the following topics if you are not familiar with the technical changes included in the DB2 Version 9.1 fix packs. Fix packs are cumulative, they contain all of the changes and functionality shipped in previous fix packs.

DB2 Version 9.1 Fix Pack 1

Fix Pack 1 includes the following changes to existing functionality:
- Availability of LOB or XML values is changed in JDBC applications with progressive streaming
- Modification level of product identifier can contain alphanumeric characters

Fix Pack 1 includes the following enhancement:
- DB2Binder utility includes two new options

DB2 Version 9.1 Fix Pack 2

Fix Pack 2 contains the Fix Pack 1 functionality and includes the following enhancements:
- BINARY, VARBINARY, and DECFLOAT data type support added for C and C++ embedded SQL applications
- DB2 .NET Data Provider enhancements and support for the .NET Framework 2.0
- IBM Database Add-Ins for Visual Studio 2005 enhancements
- IBM Software Development Kit (SDK) for Java 5.x support added for the Solaris Operating System
- Windows Vista support added (Windows)

DB2 Version 9.1 Fix Pack 3

Fix Pack 3 contains the Fix Pack 2 functionality and includes the following
enhancements:
* Change password support added (Linux)
* JDBC and SQLJ enhancements

Chapter 3. Product packaging enhancements

As IBM data server products have continued to evolve, the packaging of the DB2 components and component names have changed.

In Version 9.5, IBM has updated the list of DB2 database products available and added several new features to respond to market needs. To read about these products and to view the related licensing and marketing information, see the DB2 Database for Linux, UNIX, and Windows home page at http://www.ibm.com/db2/9.

New DB2 client product simplifies deployment (Windows)

The new IBM Data Server Driver for ODBC, CLI, and .NET client makes it easier to provide access to DB2 servers from Windows-based applications that use the ODBC driver, CLI driver, OLE DB driver, or IBM Data Server Provider for .NET.

The new IBM Data Server Driver for ODBC, CLI, and .NET client simplifies application deployment on Windows platforms. This client, which has a small footprint, is designed to be redistributed by independent software vendors (ISVs) and to be used for application distribution in mass deployment scenarios typical of large enterprises.

Driver registration and configuration during installation and driver unregistration during uninstallation are handled automatically by the DB2 installation program.

For Linux and UNIX operating systems, you can still get the IBM Data Server Driver for ODBC and CLI, in a tar format.

Related concepts

"IBM data server clients setup overview" in Quick Beginnings for IBM Data Server Clients

"IBM Data Server CLI and ODBC drivers" in Call Level Interface Guide and Reference, Volume 1

Related tasks

"Installing IBM data server clients (Windows)" in Quick Beginnings for IBM Data Server Clients

"Deploying .NET applications (Windows)" in Developing ADO.NET and OLE DB Applications

FP1: DB2 Text Search supports searching SQL and XML data

Available in Fix Pack 1, DB2 Text Search delivers an integrated and scalable search technology for DB2 databases that you can use for performing relational data, XQuery, and SQL/XML text searches and searches of other document formats. DB2 Text Search makes it easier than ever to search within DB2 databases.

DB2 Text Search is an alternative to DB2 Net Search Extender that provides new and extended search support for XML documents. DB2 Text Search includes the following features:

- Text search index support for all DB2 data servers on Linux, UNIX, and Windows. DB2 Text Search is integrated as an optionally installable feature of

DB2 data server products, with the install and configuration being performed by the DB2 installer. Tight administrative integration with DB2 data servers is provided through a DB2 command interface, a DB2 administrative routine interface, and DB2 error messages that reference an SQLCODE and an SQLSTATE.

- Text search and index support for multiple text formats. These include plain text, HTML, and XML.
- Text search capability within SQL, SQL/XML, and XQuery.
- Refined search analysis, such as the ability to process different forms of a word such as *go*, *going*, and *gone*.
- Synonym dictionary support, which allows a search to find synonyms of a piece of text, such as the synonym beautiful for the text pretty, and indexes both the text and its synonyms
- Support for the XPath expression search syntax and the use of the ftcontains function within XPATH expressions. DB2 Net Search Extender does not support the XPath expression search syntax.
- Advanced memory management during search.

Related information

"DB2 Text Search overview" in Text Search Guide

"Searching with text search indexes" in Text Search Guide

Component names have changed

As IBM data servers have continued to evolve, the related components and component names have changed.

The following table lists renamed product components in Version 9.5:

Table 1. New names for DB2 product components

Version 9.1 component name	Version 9.5 component name
DB2 Client	IBM Data Server Client
DB2 Developer Workbench	IBM Data Studio
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

Related concepts

"DB2 integration in Visual Studio" in Developing ADO.NET and OLE DB Applications

"IBM data server clients setup overview" in Quick Beginnings for IBM Data Server Clients

Chapter 4. Manageability enhancements

Managing common and critical tasks has become easier in Version 9.5. More tasks have been automated, some tasks have been consolidated, and many processes require fewer steps to complete.

Review this section to find out more details about enhancements included in Version 9.5 that make it easier to manage DB2 data servers.

Data server tools have been added

IBM Data Studio is a rich and robust user interface that you can use to perform database design, development, deployment, and management tasks. It replaces the DB2 Developer Workbench, shipped in Version 9.1.

You can use IBM Data Studio to develop and test routines, deploy data-centric Web services, create and run SQL and XQuery queries, and database applications. In addition, you can perform basic database administration tasks such as creating and altering database objects and managing privileges. You can install IBM Data Studio on the following platforms: Linux on x86; 32-bit and 64-bit Windows platforms. For more information, see "DB2 Developer Workbench has been renamed and enhanced" on page 89.

The Control Center and related tools (such as the Command Editor and the Task Center) are still available to provide a general-purpose solution for database design and operational database administration. They allow you to create and modify database objects, move data into and out of databases, perform routine maintenance, set up and manage database recovery strategies, manage database scripts, optimize queries, and optimize database design using advisor-based wizards.

IBM Data Studio maintenance is applied using the IBM Installation Manager updater facility. You can download the latest IBM Data Studio fix packs at http://www.ibm.com/software/data/studio/support.html and use the IBM Data Studio information center at http://publib.boulder.ibm.com/infocenter/dstudio/v1r1m0.

Related concepts

"DB2 Developer Workbench has been renamed and enhanced" on page 89

Real-time statistics gathering ensures that the latest statistics are used for optimization

Version 9.5 introduces real-time statistics gathering, whereby table statistics are automatically collected whenever they are needed to optimize and run a query. Real-time automated statistics are enabled by the new dynamic configuration parameter **auto_stmt_stats**.

Using accurate and up-to-date database statistics helps create good query execution plans and can shorten query processing times. Prior to Version 9.5, you could collect statistics either manually or through periodic automated statistics collection.

DB2 Universal Database™ Version 8.2 introduced automatic statistics collection, which involved monitoring tables and collecting statistics periodically on tables where high amounts of table activity led to changed statistics. This background process performed an evaluation of table activity at fixed time intervals. Therefore, there could be a gap between the time when data changed and the time when new statistics were collected.

Real-time statistics gathering closes the gap in automated statistics collection. When you submit a query to the compiler, the optimizer determines whether the statistics for the referenced tables are accurate. If there are no statistics or if the tables have changed significantly since statistics were last collected, the statistics are collected again, possibly during statement compilation. The time to collect statistics at statement-compilation time is limited to 5 seconds by default. If statistic collection takes longer than 5 seconds, a background request is generated instead. The limit is configurable via an optimization profile. In some cases, statistics can be fabricated at statement-compilation time, based on the meta data maintained by the data manager and the index manager. If changes to a table do not require immediate action to update the statistics yet there are significant changes to the table, a background request is generated to collect statistics as soon as possible.

Related concepts

"Automatic statistics collection" in Data Servers, Databases, and Database Objects Guide

"Optimizer profiles and guidelines overview" in Optimization Profiles

"RTS requests" in Tuning Database Performance

Related reference

"catalogcache_sz - Catalog cache size configuration parameter" in Data Servers, Databases, and Database Objects Guide

"auto_maint - Automatic maintenance configuration parameter" in Data Servers, Databases, and Database Objects Guide

Simplified multithreaded architecture reduces the Total Cost of Ownership (TCO)

DB2 data servers now have a multithreaded architecture on all platforms. Prior to Version 9.5, DB2 data servers running on UNIX and Linux operating systems used the process-based model, in which each agent ran in its own process.

The move to the multithreaded architecture provides the following benefits:

• Enhanced usability and lower Total Cost of Ownership (TCO), through the following features and enhancements:
  – Dynamic self tuning memory is now fully enabled on all platforms.
  – Memory configuration is simplified through more automatic and dynamic memory configuration parameters. For more information, see "Memory configuration has been simplified" on page 30.
  – Agent and process model configuration is simplified. These enhancements eliminate the need for regular DBA interventions to adjust the process model-related parameters and reduce the time and effort required to configure them. You do not need to shut down and restart DB2 instances to have the new values take effect.
  – New dynamic configuration parameters enable the database manager to automatically tune your system without manual intervention.

– Performance is improved because context switching between threads is usually faster than between processes (depending on the application and platform).
- Sharing of resources, such as file handles, is more efficient and provides system resources savings because all agents working on the same database file share the same file handle.
- Memory footprint is reduced.
- A consistent threading architecture provided across all operating systems reduces the complexity and maintenance of your data servers.

  Related concepts

  "The DB2 Process Model" in Tuning Database Performance

  "Memory Visualizer displays maximum memory consumption" on page 137

  "Database configuration across multiple partitions has been simplified"

  "Memory configuration has been simplified" on page 30

  "Some database configuration parameters have changed" on page 139

  "Some database manager configuration parameters have changed" on page 123

Database configuration across multiple partitions has been simplified

Version 9.5 provides a single view of all database configuration elements across multiple partitions. You can update or reset a database configuration across all database partitions without invoking either the UPDATE DATABASE CONFIGURATION or RESET DATABASE CONFIGURATION command against each database partition. You no longer need to use the db2_all command.

You can update a database configuration across partitions by issuing only one SQL statement or only one administration command from any partition on which the database resides. This means that by default, the behavior of updating or resetting a database configuration has been changed from on local database partition to on all database partitions.

Version 9.5 provides the following ways to achieve backward compatibility for command scripts and applications:
- A new registry variable, **DB2_UPDDBCFG_SINGLE_DBPARTITION**. Set this registry variable to TRUE. Note that this alternative method does not apply to UPDATE DATABASE CONFIGURATION or RESET DATABASE CONFIGURATION requests that you make through the ADMIN_CMD procedure.
- A new option, **dbpartitionnum**, for the UPDATE DATABASE CONFIGURATION and RESET DATABASE CONFIGURATION commands and for the ADMIN_CMD procedure.

To update or reset a database configuration on a specific partition, you specify the **dbpartitionnum** option with the UPDATE DATABASE CONFIGURATION or RESET DATABASE CONFIGURATION command.

  Related reference

  "RESET DATABASE CONFIGURATION command" in Command Reference

  "UPDATE DATABASE CONFIGURATION command" in Command Reference

  "System environment variables" in Data Servers, Databases, and Database Objects Guide

"ADMIN_CMD procedure – Run administrative commands" in Administrative Routines and Views

Memory configuration has been simplified

In previous releases, you could enable self-tuning memory for most performance-related memory parameters; however, you still needed to configure other memory heaps required by the DB2 data server. Now, this configuration task has been simplified by the default AUTOMATIC setting on most memory-related configuration parameters.

Version 9.5 enhancements to self-tuning memory provide the following benefits:
* You can use a single parameter, **instance_memory**, to specify all of the memory that the database manager is allowed to allocate from its private and shared memory heaps. You can use the new **appl_memory** configuration parameter to control the maximum amount of application memory that is allocated by DB2 database agents to service application requests. By default, its value is set to AUTOMATIC, meaning that application memory requests are allowed if the total amount of memory allocated by the database partition is within the **instance_memory** limits.
* You do not need to manually tune parameters used solely for functional memory.
* You can query how much total memory is currently being consumed by private and shared memory heaps of the database manager (by using the Memory Visualizer). You can also use the db2mtrk command to monitor heap usage and the ADMIN_GET_DBP_MEM_USAGE table function to query overall memory consumption.
* With the simplified application memory model, it is much easier to configure and tune application memory when required.
* The default DB2 configuration requires much less tuning, an immediate benefit for new instances.

    Related concepts

    "Self-tuning memory" in Data Servers, Databases, and Database Objects Guide

    "Self tuning memory operational details and limitations" in Data Servers, Databases, and Database Objects Guide

    "Self-tuning memory overview" in Data Servers, Databases, and Database Objects Guide

    "Memory Visualizer displays maximum memory consumption" on page 137

    "Some configuration parameters are impacted by simplified memory configuration" on page 145

    "Some database configuration parameters have changed" on page 139

    "Some database manager configuration parameters have changed" on page 123

    Related reference

    "instance_memory - Instance memory configuration parameter" in Data Servers, Databases, and Database Objects Guide

    "db2mtrk - Memory tracker command" in Command Reference

    "appl_memory - Application Memory configuration parameter" in Data Servers, Databases, and Database Objects Guide

    "ADMIN_GET_DBP_MEM_USAGE table function - Get total memory consumption for instance" in Administrative Routines and Views

Data compression has been enhanced

Automatic dictionary creation (ADC) is now available. ADC reduces and sometimes eliminates the need for doing offline table reorganizations. When you add data to a table enabled for data compression, the first time that a threshold (default of approximately 1 to 2 MB) is passed, the data compression dictionary is automatically created.

In this release, data compression dictionaries can be automatically created during data population operations on tables where you have defined the COMPRESS attribute. If no data compression dictionary yet exists in the table, ADC can create a dictionary and insert it into the table. Data moved into the table following the creation of the dictionary is compressed.

Data table population operations that can cause the creation of a data compression dictionary are INSERT, IMPORT INSERT, LOAD INSERT, and REDISTRIBUTE DATABASE PARTITION GROUP. The LOAD REPLACE utility now has the ability to perform explicit data compression dictionary management by using the KEEPDICTIONARY and RESETDICTIONARY keywords. You can keep an existing compression dictionary, or you can now create a new compression dictionary even if one already exists.

Related concepts

"Space compression for tables" in Data Servers, Databases, and Database Objects Guide

"Automatic (compression) dictionary creation (ADC)" in Data Servers, Databases, and Database Objects Guide

More configuration parameters can be set to AUTOMATIC and configured dynamically

In Version 9.5, you can set more configuration parameters to the AUTOMATIC setting, and you can configure more configuration parameters dynamically, without having to stop and restart your instance or database. These configuration parameter settings enable the database manager to handle more of the database tuning and react automatically to changes in the system workload.

Table 2 on page 32 lists the configuration parameters that allow the AUTOMATIC setting and that you can configure dynamically. The meaning of the AUTOMATIC setting differs with each parameter. In general, the AUTOMATIC setting means that the database configuration parameters are automatically tuned for you according to your system resources. These parameters are enabled by default during the creation of new databases or instances and in some database and instance migration scenarios.

Table 2. Additional configuration parameters that can be configured dynamically and set to AUTOMATIC

Configuration parameter name	Description	Meaning of the AUTOMATIC setting
applheapsz	In previous releases, this sets the amount of application memory that each individual database agent working for an application could consume. With Version 9.5, this sets the total amount of application memory that can be consumed by an entire application. In the case that there are multiple agents working for the same application, then **applheapsz** refers to the sum of application memory that are consumed by all these agents. For DPF, Concentrator, or SMP configurations, the **applheapsz** value used in previous releases might need to be increased for workloads similar to those in previous releases unless the AUTOMATIC setting is used.	The AUTOMATIC setting allows the application heap size to increase as needed until either the **appl_memory** limit is reached or the **instance_memory** limit is reached.
database_memory	This specifies the amount of memory that is reserved for the database shared memory region.	The AUTOMATIC setting enables self-tuning. When enabled, the memory tuner determines the overall memory requirements for the database and increases or decreases the amount of memory allocated for database shared memory depending on the current database requirements.
dbheap	Determines the maximum memory used by the database heap.	The AUTOMATIC setting allows the database heap to increase as needed until either the **database_memory** limit is reached, or the **instance_memory** limit is reached.
instance_memory	This specifies the maximum amount of memory that can be allocated for a database partition.	The AUTOMATIC setting allows the actual value to be computed at database partition activation time (db2start).

Configuration parameter name	Description	Meaning of the AUTOMATIC setting
mon_heap_sz	This determines the amount of the memory to allocate for database system monitor data.	The AUTOMATIC setting allows the monitor heap to increase as needed until the **instance_memory** limit is reached.
stat_heap_sz	Indicates the maximum size of the heap used in collecting statistics using the RUNSTATS command.	The AUTOMATIC setting allows the statistics heap size to increase as needed until either the **appl_memory** limit is reached, or the **instance_memory** limit is reached.
stmtheap	This specifies the size of the statement heap, which is used as a work space for the SQL or XQuery compiler during compilation of an SQL or XQuery statement.	The AUTOMATIC setting allows the statement heap to increase as needed until either the **appl_memory** limit is reached, or the **instance_memory** limit is reached.

Example of using the AUTOMATIC setting for the database_memory configuration parameter

If you set the **database_memory** configuration parameter to AUTOMATIC, if the current database requirements are high and there is sufficient free memory on the system, more memory is consumed by database shared memory. If the database memory requirements drop or the amount of free memory on the system drops too low, some database shared memory is released.

Related reference

"Configuration parameters summary" in Data Servers, Databases, and Database Objects Guide

Data redistribution enhancements reduce capacity growth costs and load-balancing tasks

Starting with Version 9.5 Fix Pack 1, performance and usability of data redistribution are dramatically improved through the provision of an enhanced data redistribution utility that can perform all of the tasks related to data redistribution and do so more efficiently.

Data redistribution in capacity growth scenarios, during load balancing, or during performance tuning can require precious maintenance window time, a considerable amount of planning time, as well as log space and extra container space that can be expensive.

Prior to Version 9.5 Fix Pack 1, you had to perform some data redistribution tasks, such as reorganizing tables and collecting statistics, separately. Now, you can do them together by issuing the REDISTRIBUTE DATABASE PARTITION GROUP command and by specifying the NOT ROLLFORWARD RECOVERABLE option which enables new features and optional keywords for controlling how and when

you perform these tasks. The combination and automation of these otherwise manual tasks makes them less error prone, faster, and more efficient, while providing you with more control over the operations. Examples of the tasks that you can perform by issuing the REDISTRIBUTE DATABASE PARTITION GROUP command include:

- Adding partitions to a database partition group
- Dropping partitions from a database partition group
- Moving data to a target partition based on a new partitioning map
- Reorganizing table data so that the effects of fragmentation are greatly reduced, which can improve scan performance for all tables and reduce storage requirements for tables that are not multi-dimensionally clustered
- Rebuilding indexes
- Collecting statistics

When the REDISTRIBUTE DATABASE PARTITION GROUP command is used with the NOT ROLLFORWARD RECOVERABLE option:

- Data is moved in bulk instead of by internal insert and delete operations. This reduces the number of times that a table must be scanned and accessed, which results in better performance.
- Log records are no longer required for each of the insert and delete operations. This means that you no longer need to manage large amounts of active log space and log archiving space in your system when performing data redistribution. This is particularly beneficial if, in the past, large active log space and storage requirements forced you to break a single data redistribution operation into multiple smaller redistribution tasks, which might have resulted in even more time required to complete the end-to-end data redistribution operation.
- More granular control of data redistribution is possible through the use of additional REDISTRIBUTE DATABASE PARTITION GROUP command options:
  - **Addition or removal of database partitions**

    You can add or remove database partitions during data redistribution using the ADD DBPARTITIONNUM option. Previously, you had to issue a separate ALTER NODEGROUP command to add or remove database partitions.
  - **Control of data redistribution processing**

    You can use the new **TABLE** option to specify the order in which tables are processed as part of a data redistribution operation. For example, you can ensure that critical tables are processed first and brought back online (read-only) as quickly as possible but that less critical tables are processed later. You can now halt data redistribution for a table using the **STOP** option to pause future processing of additional tables after the specified time and use the **CONTINUE** option to complete the data redistribution at a later time.

    Note: If the stop event occurs while the command is in the middle of redistributing a table, the data redistribution utility does not stop at the specified time. The stop time is checked only between table redistributions.
  - **Maintenance of indexes**

    The INDEXING MODE option allows users to choose between two different types of index maintenance during data redistribution. The two possible values of this option are:
    - **INDEXING MODE REBUILD**
      - This option specifies that indexes are to be completely rebuilt.

- **INDEXING MODE DEFERRED**
  - This option specifies that index maintenance should be deferred. Indexes are marked as invalid. This option is useful if there are other utility operations to be performed on the tables and all of the indexes are to be recreated after these operations are complete.
- **Freeing of space during redistribution**

  When you perform data redistribution, the table is automatically reorganized, which can free up disk space. This table reorganization comes at no additional performance cost to the redistribute operation.

  For tables with clustering indexes, the reorganization does not attempt to maintain clustering. If perfect clustering is desired, it will be necessary to perform a REORG on tables with a clustering index after data redistribution completes.

  For MDC tables, the reorganization maintains the clustering of the table and frees unused blocks for reuse; however the total size of the table after redistribution appears unchanged.
- **Maintenance of statistics**

  When you perform data redistribution on a table that has a statistics profile, you can simultaneously gather table statistics and then use them to update the table statistics in the DB2 catalogs after the processing of the table is complete. All statistics specified in the statistics profile are gathered.

  To specify that statistics should be gathered and updated, specify the new default option **STATISTICS USE PROFILE**.

  Index statistics are collected only if you specify the **INDEXING MODE REBUILD** option.

  If no statistics profile exists and you specify **INDEXING MODE REBUILD**, no statistics are gathered.
- **Memory consumption**

  When you perform data redistribution, utility heap memory is used. To specify the number of 4 KB pages of utility heap memory to be used to redistribute data in each table, specify the new option **DATA BUFFER** *data_buffer_sz*. Use this option to tune the performance of data redistribution operations.

  If you do not specify this option, the default behavior is to use 50% of the utility heap memory that is available for each table at the time when processing of the table begins.
- **Crash recovery or rollforward and data redistribution**

  The REDISTRIBUTE DATABASE PARTITION GROUP command puts table spaces associated with tables being redistributed into the backup pending state to ensure that no modifications to data in that table space can be performed until the table space is backed up and to ensure that all affected table spaces have been backed up, either through table space–level backups or a database-level backup. It is critical that you back up each affected table space or the entire database when the redistribute operation is complete because rolling forward through a redistribute operation results in all tables that were redistributed being marked invalid. The REDISTRIBUTE DATABASE PARTITION GROUP command is not rollforward recoverable. For more information, refer to the REDISTRIBUTE DATABASE PARTITION GROUP topic.

The streamlining of data redistribution tasks into one utility, the new REDISTRIBUTE DATABASE PARTITION GROUP command options, and the new

redistribute utility behaviors will help data redistribution take less time, be less error prone, and consume fewer system resources. As a result, the total cost of performing data redistribution is reduced, which means frees up time and resources for other business operations.

Related concepts

"Collecting statistics using a statistics profile" in Tuning Database Performance

Related reference

"REDISTRIBUTE DATABASE PARTITION GROUP command" in Partitioning and Clustering Guide

db2look command generates DDL for more database objects

The db2look command, which you can use to help you understand, move, or reproduce database objects generates DDL for more database objects.

In addition to generating DDL for the new Version 9.5 objects, such as roles, workload management objects, audit objects, trusted contexts, array types, and global variables, the db2look command includes the following enhancements:

- You can generate the table space DDL for partitioned tables using db2look **-d** *dbname* **-t** *tabname* **-l**
- You can generate UPDATE statistics for statistical views using db2look **-d** *dbname* **-t** *tabname* **-m**
- Errors are generated if you specify incorrect command line parameters or if you specify table names that do not exist when you use the **-t** option.
- The order of the DDL generation for database objects is improved when you use the **-e** option.

Related reference

"db2look - DB2 statistics and DDL extraction tool command" in Data Movement Utilities Guide and Reference

Enhanced access to DB2 administration commands through SQL

SQL administrative routines introduced in Version 8 are expanded in Version 9.5 to include more administrative tasks. New administrative views are also added in Version 9.5.

The SQL administrative routines and views provide a primary, easy-to-use programmatic interface to use DB2 functionality through SQL. They encompass a collection of built-in views, table functions, procedures, and scalar functions for performing a variety of administrative tasks such as: reorganizing a table, capturing and retrieving monitor data, or retrieving the application ID of the current connection.

These routines and views can be invoked from an SQL-based application, a command line or a command script.

In addition to the new administrative views, routines, and procedures, Version 9.5 includes:

- Extended workload management capabilities
- Expanded support for monitoring your database

In order to provide expanded support for the existing administrative routines, some of the Version 9 routines have been replaced with new, more comprehensive routines or views in Version 9.5.

For a list of all supported administrative SQL routines and view, including the new ones, see "Supported administrative SQL routines and views" in *Administrative Routines and Views*.

Related concepts

"Some system catalog views and built-in routines have been added and changed" on page 133

Monitoring capabilities have been extended

Version 9.5 includes new monitor elements to help you check the status of various processes, watch for potential problems, gather information when you are experiencing a problem, or adjust configuration settings to enhance performance.

More than 80 monitor elements have been added to track of system activities. You can use some of these new elements to monitor new Version 9.5 functionality such as real-time statistics and the DB2 workload management feature. Other elements extend the existing monitoring functionality such as by monitoring additional table space activities.

If you are migrating from a previous release and you created write-to-table event monitors, you must re-create them to use the new elements. For more information, see "Re-creating write-to-table event monitors" in *Migration Guide*.

Related reference

"Database system monitor elements" in System Monitor Guide and Reference

"Table space activity monitor elements" in System Monitor Guide and Reference

"Workload management monitor elements" in Workload Manager Guide and Reference

"Real-time statistics monitor elements" in System Monitor Guide and Reference

"On-Line Analytical Processing (OLAP) monitor elements" in System Monitor Guide and Reference

Licensing monitoring is more flexible and efficient

A new table function and administrative view make it easier to gather licensing information. You can use them to create reports and manage the licensing of the DB2 products and features that are installed.

The ENV_FEATURE_INFO administrative view and the ENV_GET_FEATURE_INFO table function return information about all available features for which a license is required. For each feature, there is information about whether a valid license for the feature is installed.

Related reference

"db2licm - License management tool command" in Command Reference

"DB2 license files" in Getting Started with DB2 installation and administration on Linux and Windows

"ENV_FEATURE_INFO administrative view - Return license information for DB2 features" in Administrative Routines and Views

Table spaces use space more efficiently

You can use a new option for the ALTER TABLESPACE statement to reclaim unused table space. With less storage space to back up and restore, this, in turn, saves you time.

The following table space handling enhancements ensure that storage space usage is optimal for the amount of information contained in a database:

- The ALTER TABLESPACE statement supports the REDUCE option for table spaces managed by automatic storage
- Version 9.5 determines a more accurate value for the High Water Mark (HWM). This would then allow the REDUCE, RESIZE, and DROP options to change the table space size to something that more accurately represents the amount of space used. In turn, the space that is freed can then be used by other table spaces.

These statements can reduce the size of a table space only to a value above the HWM, not below it.

> **Related reference**
>
> "ALTER TABLESPACE statement" in SQL Reference, Volume 2

FP1: db2pd command output provides more information

Version 9.5 Fix Pack 1 contains new db2pd features that make it easier to monitor system activities. A summary option is available for the **-pages** command parameter, and new command parameters (**-wlocks** and **-apinfo**) are also available.

You can use the summary option for the **-pages** parameter to generate a more compact report that contains only the buffer pool summary information section. Additional columns that include information about table space IDs, dirty pages, permanent pages, and temporary pages are displayed in the summary section.

You can use the**-wlocks** parameter to monitor dynamically the applications with locks that are in lock wait mode. You can use the **-apinfo** parameter to capture detailed runtime information about a specific application or for all applications. Both parameters have options to save the information into separate files.

> **Related reference**
>
> "db2pd - Monitor and troubleshoot DB2 database command" in Command Reference

Lock timeout diagnosis has been enhanced

Version 9.5 includes a lock timeout reporting function that simplifies the diagnosis of lock timeout situations in complex DB2 environments and reduces the time required to correct them.

You can use the lock timeout reporting function to log information about lock timeouts at the time that they occur. Information is captured for both the lock requester (the application that received the lock timeout error) and the current lock owner. The logged information identifies: the key applications involved in the lock contention that resulted in the lock timeout, the details about what these applications were running at the time of the lock timeout, and the details about the lock causing the contention. A text report is written and stored in a file for each lock timeout encountered.

You can turn on and off the lock timeout reporting function dynamically by setting the **DB2_CAPTURE_LOCKTIMEOUT** registry variable.

Related concepts

″Lock timeout reporting″ in Tuning Database Performance

Related reference

″General registry variables″ in Data Servers, Databases, and Database Objects Guide

″Lock timeout report files″ in Tuning Database Performance

RUNSTATS utility option has been added to reset statistics profiles

The new UNSET PROFILE option for the RUNSTATS utility allows you to reset your statistics profile.

The RUNSTATS utility provides an option to register and use a statistics profile, which is a set of options that specify which statistics are to be collected on a particular table.

To revert to the default settings for the statistics profile, issue the RUNSTATS command with the UNSET PROFILE option. Alternately, you can call the db2Runstats API and set the **iRunstatsFlags** parameter to DB2RUNSTATS_UNSET_PROFILE.

Related concepts

″Collecting statistics using a statistics profile″ in Tuning Database Performance

Related reference

″RUNSTATS command″ in Command Reference

″db2Runstats API - Update statistics for tables and indexes″ in Administrative API Reference

Chapter 5. Workload management enhancements

Version 9.5 features extends the workload management capabilities provided in previous releases.

Review this section to find out more details about Version 9.5 workload management enhancements.

Workload management improvements provide better control

Comprehensive workload management has been integrated into Version 9.5, giving you deeper insight into how your system is running and finer control over resources and performance.

Version 9.5 workload management lets you divide your work into classes and tailor your data server to support a variety of users and applications on the same system. You do not have to create separate databases for transaction processing and data mining.

You can perform the following tasks using DB2 workload management features:

- You can divide your work into manageable and logical groups by automatically identifying work using workload definitions, assigning workloads to services classes and allocating resources to each service class. You can capture detailed workload profiles and performance information to help refine your workload and service class definitions.
- You can control execution through cost, time, and concurrency thresholds, which enable you to take control of rogue queries and help you meet your service level agreement (SLA) objectives. Using thresholds, the system can automatically react to a bad situation or predict it before it happens. You can keep transactions running smoothly while you control the impact of long running and complex queries.
- You can follow work through every stage of processing so that you can provide your users with up-to-date status information.

You can leverage the power of the AIX operating system by tying DB2 service classes to AIX Workload Manager (WLM) service classes. For example, the AIX WLM can dynamically adjust CPU shares or use spare CPU shares from other service classes to get the most out of your data server.

Overview of DB2 workload management

The following figure shows how multiple requests sent to the data server are evaluated against and assigned to specific workloads and then executed in the applicable service class. Requests that cannot be matched to workloads that you defined are assigned to the default workload, which is executed in the default service class.
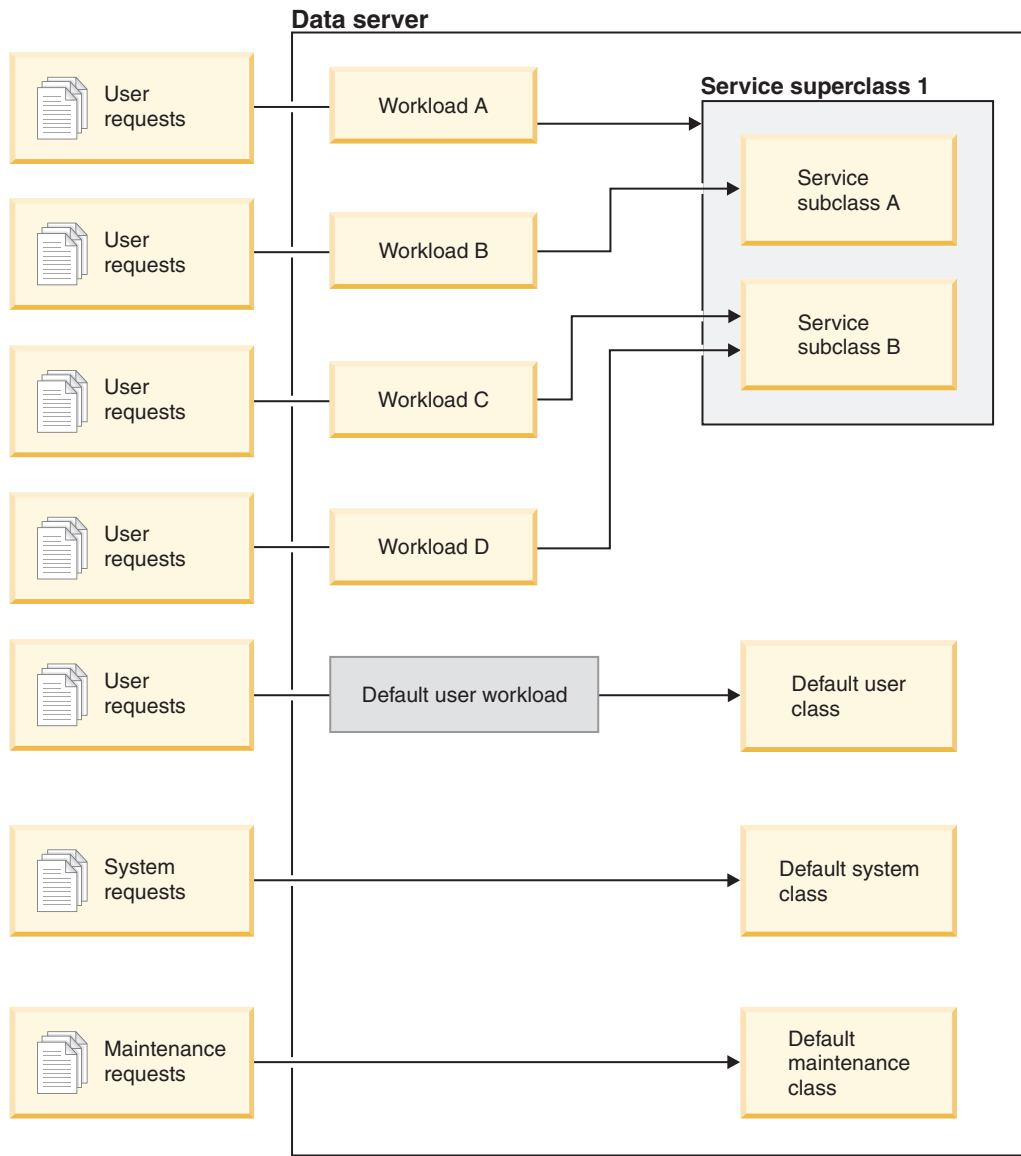
Figure 1. Service classes and workloads

Advantages of using DB2 workload management

Finer control over system resources

> With the ever-increasing volume of database activity today, contention for system resources such as CPU, I/O, and memory is becoming more and more of an obstacle to achieving business objectives. The enhanced DB2 workload management features help you to identify and isolate a defined set of database activities into their own execution environments to which you can assign appropriate resources needed to meet your objectives. Within the environment or service classes, you can explicitly manage system resources so that more critical resources are available for consumption by higher-priority work and contention with lower-priority work can be controlled or eliminated.

> Although resource control is available on all platforms supported in Version 9.5, data servers on the AIX operating system can directly exploit the resource management capabilities provided by the WLM. You can link

DB2 service classes to an AIX WLM resource group, thereby enabling an even more dynamic level of resource management that combines the workload management capabilities of the data server and those of the operating system itself.

Increased predictability and stability of your system

Most businesses experience peak periods, with increased volumes of activities and demands that affect the workloads of data servers. A peak period could be in the middle of the day, when the most users are on the system, or it could be at the end of the month, when all of the detailed monthly reports are due. During these peak times, the response time for a workload that normally runs in a predictable amount of time can become unpredictable. Unintentional peak periods can also be created by users who impose additional work on data servers because there is no control in place to restrict how much resources they can use. For example, users can inadvertently submit SQL statements that require a lot of processing by the server or SELECT statements that include complicated joins, such as Cartesian joins.

DB2 workload management can smooth out peak workloads by allowing you to predetermine the proper resource allocation, prioritization of activities, and queueing options to process the work in an efficient manner. After you define these directives, the data server uses them to allocate its resources and prioritize its work. For example, you can isolate work from the impact of rogue queries, which use an excessive amount of database resources, negatively impacting other queries running on the system and, potentially, impacting the database system as a whole. Using thresholds, you can define acceptable query behaviors within your system using a number of different characteristics such as execution time or system temporary table space usage and define what actions to take for any query that does not perform as expected. These actions include the ability to collect detailed information about the query and the ability to automatically cancel that query.

Tailored performance requirements in environments with mixed workloads

Mixed workloads that coexist on the same data server share the same resources but might have different performance requirements. For example, batch workloads are often run overnight when the data server is relatively quiet and they do not negatively affect the daily report jobs that run in the middle of the day.

DB2 workload management helps you focus on the performance of mixed workloads by allowing you to prioritize workloads effectively and by directing resources to where they are most needed. You can maximize the overall throughput of the data server activities on the system by using the customized control and resource allocation features.

You can also gauge the performance of a data server by using tangible and intangible measures. Examples of tangible measures are the database statistics showing the amount of time required to complete a specific set of activities and the individual amount of time required to complete a simple query or a more complex job, such as a batch job that loads data into a warehouse. An intangible method might be your users' perception of and satisfaction with data server response time.

To optimize performance, you can use the workload management monitoring features to obtain both aggregate and point-in-time information about work that is running on the data server. If some types of work are

not completing in the required time frame, you can use the monitoring data to help you understand what is going on and modify your configuration. For example, you might decide to assign additional resources to a service class or assign resource-usage controls for certain types of work. After making these changes, you might then monitor system behavior to validate that the changes that you made produce the response times that you require and do not introduce other, unexpected behavior. Workload management is an iterative process; you can refine your configuration until you obtain results that meet your business requirements.

Easier management and monitoring of SLA objectives

An SLA is a formal agreement between groups that defines the expectations between them and includes objectives for items such as services, priorities, and responsibilities. SLA objectives are often formulated using response time goals. For example, a specific Human Resources report might need to run, on average, within 5 minutes. Other examples are a requirement that updates from a cash register to the inventory system must always be completed in less than 2 seconds or that data must be loaded using a batch job that runs before 8 a.m. to have the daily sales reports available by 9 a.m.

In the past, tracking actual performance against SLA objectives required custom coding to record statistics and manual steps to extract data to compute the performance measures. However, DB2 workload management provides a robust set of monitoring functions that simplify the tasks of measuring results against goals defined in an SLA. You can use aggregate measures to capture average response time without collecting data about each and every activity. For example, you do not need to collect a response time value every time that a cash register sale is recorded in a table; there could be hundreds of thousands of these entries in a single day. Instead, you can now collect these statistics when you need them to determine whether your objectives are being met.

DB2 workload management retains information about the distribution of activities that you are measuring through histograms. You can analyze histogram data to determine particularly high (or low) standard deviations that could indicate that response times are inconsistent (or very consistent) and to determine whether response times are meeting the SLA objectives.

Customized SLA objectives for multiple customer groups on the same system

Multiple customers sharing the resources of a data server do not necessarily have the same SLA objectives. For example, one application might be used by three different departments. One of the departments might want the response time to be less than 2 seconds on average, while the other two departments might be satisfied with a response time of 5 seconds.

DB2 workload management enables you to provide distinct environments to enable independent support for customized SLAs for customers who share the same database. You can use the execution environments to isolate the activities on the data server using service classes for each different types of customers. For example, you can set up a workload by group, then assign it to a different service class with less resources. After you set up the service classes, you can easily collect and monitor aggregate activity statistics to ensure that the SLA objectives are met for each customer. You can charge each customer based on the level of service received.

Simplified consolidation of applications and business units on the same data server

With hardware and operating systems being able to handle more transactions, many opportunities exist to reduce operating costs by consolidating business units and applications on fewer data servers. You can use DB2 workload management to manage environments where applications and business units use the same server but do not have the same needs and have separate funding.

For example, assume that the data server for a Payroll department is merged with the data server for a Human Resources department. The Payroll group has a standard set of tasks to provide weekly paychecks, track employee expenses, and send out year-end tax receipts. The Human Resources department does some regular trend analysis but has a lot more ad hoc activities because the department reacts to issues that requires ad hoc access to employee data. Each group has its own set of goals and priorities, and each department has separate funding. To ensure that each group on the same data server uses only the resources to which it is entitled, you can create separate execution environments for each group. You can define each environment to include the work specific to that particular group, and you can allocate resources to and set priorities for that group that correspond to its level of funding.

Isolation of database activities ported from other data servers

Consolidation also commonly occurs when applications and data are ported to a DB2 data server from a different data server system. You can use DB2 workload management to establish an execution environment to help ensure that the ported applications get their required resources. Also, when you have all of this work isolated in a service class, it is straightforward to monitor these activities, which makes it easy to compare performance between the old data server and the new DB2 data server and to help validate the decision to move to the new server.

Dynamic monitoring of database activity

All database activities are assigned to DB2 workloads, which run in a DB2 service class. In Version 9.5, you can use a number of table functions to query information about the state and content of the activities found in a workload or service class. This information can give you a clear picture of what work is currently running on the system, how it is distributed across partitions, and whether there are particular activities that might cause resource contention on the data server.

Enhanced chargeback capability

With the mapping of DB2 workload management service classes to AIX WLM service classes, you can not only control CPU resources at the operating system level but also monitor CPU usage at the service class level. This gives you more options to charge business units appropriately based on the amount of CPU resources that they consume.

Using service classes, moving resource management from Query Patroller and the Governor to the new workload management facilities is easy. As you identify types of work that you would like to start managing through DB2 workload management, you can define service classes where each type of work is to be run and map each type of work to its designated service class. You no longer need Query Patroller to manage this work. DB2 workload management has the added benefit of being able to monitor and control database activities through the life

cycle of the activities across all database partitions.

Related concepts

″Workloads″ in Workload Manager Guide and Reference

″Service classes″ in Workload Manager Guide and Reference

″Integration of DB2 workload management and the AIX Workload Manager″ in Workload Manager Guide and Reference

″Thresholds″ in Workload Manager Guide and Reference

″Introduction to workload management concepts″ in Workload Manager Guide and Reference

″Work action sets, work actions, work class sets, and work classes″ in Workload Manager Guide and Reference

Chapter 6. Security enhancements

Security enhancements for Version 9.5 include support for trusted contexts and roles and improved auditing and label-based access control.

Review this section to find out more details about Version 9.5 security enhancements.

Trusted contexts enhance security

Using trusted contexts provides greater control when you use restricted, sensitive privileges, and allows middle-tier servers or applications to assert the identity of the end-user to the database server.

A database connection is referred to as a trusted connection when the connection attributes match the attributes of a trusted context defined at the DB2 server. The trust relationship is based upon the following set of attributes:

- System authorization ID: Represents the user who establishes a database connection
- IP address (or domain name): Represents the host from which a database connection was established
- Data stream encryption: Represents the encryption setting (if any) for the data communication between the database server and the database client

How trusted contexts improve security

The three-tier application model extends the standard two-tier client/server model by placing a middle tier between the client application and the database server. It has gained great popularity in recent years, particularly with the emergence of Web-based technologies and the Java 2 Enterprise Edition (J2EE) platform. An example of a software product that supports the three-tier application model is IBM WebSphere® Application Server (WAS).

In a three-tier application model, the middle tier is responsible for authenticating the users running the client applications and for managing the interactions with the database server. Traditionally, all of the interactions with the database server occurred through a database connection established by the middle tier using a combination of a user ID and a credential that identified that middle tier to the database server. That is, the database server used the database privileges associated with the user ID of the middle tier for all authorization checking and auditing that had to occur for any database access, including access provided by the middle tier on behalf of a user.

Although the three-tier application model has many benefits, having all interactions with the database server (for example, a user request) occur using the authorization ID of the middle tier raises several security concerns:

- **Loss of user identity**

  For access control purposes, some enterprises prefer to know the identity of the actual user accessing a database.

- **Diminished user accountability**

Accountability through auditing is a basic principle in database security. Not knowing the user's identity makes it difficult to distinguish the transactions performed by the middle tier for its own purpose from those performed by the middle tier on behalf of a user.

- **Overgranting of privileges to authorization ID of the middle tier**

  The authorization ID of the middle tier must have all the privileges necessary to run all of the requests from all the users. This results in the security issue of giving users unnecessary access to certain information.

- **Weakened security**

  In addition to the privilege issue raised in the previous point, current industry practice requires that the authorization ID used by the middle tier to connect must be granted privileges on all resources that might be accessed by user requests. If that middle-tier authorization ID is ever compromised, all those resources are exposed.

These security concerns show that there is a need for a mechanism whereby the actual user's identity and database privileges are used for database requests performed by the middle tier on behalf of that user. The most straightforward approach for achieving this goal would be for the middle tier to establish a new connection using the user's ID and password and then direct the user's requests through that connection. Although simple, this approach suffers from several drawbacks:

- Inapplicability to certain middle tiers. Many middle-tier servers do not have the user authentication credentials needed to establish a connection.

- Performance overhead. There is an obvious performance overhead associated with creating a new physical connection and re-authenticating the user at the database server.

- Maintenance overhead. There is maintenance overhead in having two user definitions (one on the middle tier and one at the server). This requires changing passwords at different places.

The trusted contexts capability addresses this problem. The security administrator (who holds SECADM authority) can create a trusted context object in the database that defines a trust relationship between the database and the middle tier. The middle tier can then establish an explicit trusted connection to the database, which gives the middle tier the ability to switch the current user ID on the connection to a different user ID, with or without authentication. In addition to solving the user identity assertion problem, trusted contexts offer another advantage: the ability to control when a privilege is made available to a database user. The lack of control over when privileges are available to a user can weaken overall security: for example, privileges might be used for purposes other than those originally intended. Now, the security administrator can assign one or more privileges to a database role and assign that role to a trusted context object. Only trusted database connections (explicit or implicit) that match the definition of that trusted context can take advantage of the privileges associated with that role.

> **Related concepts**
>
> "Trusted contexts and trusted connections" in Database Security Guide
>
> **Related tasks**
>
> "Establishing an explicit trusted connection and switching the user ID Using trusted contexts and trusted connections" in Database Security Guide
>
> **Related reference**
>
> "CREATE TRUSTED CONTEXT statement" in SQL Reference, Volume 2

Audit facility performance and administration have been enhanced

The audit utility generates a trail of audit records for a series of predefined and monitored database events. Version 9.5 offers major enhancements to the audit facility.

The enhancements to the DB2 audit facility for Version 9.5 include fine grained configuration, new audit categories, separate instance and database logs, and new ways to customize the audit configuration. Because you now have control over exactly which database objects are audited, you no longer need to audit events that occur for database objects that you are not interested in. Consequently, the performance of auditing (and its performance impact on other database operations) has been greatly improved.

Sole responsibility for managing audits at the database level now lies with the security administrator.

The following audit facility enhancements are included in Version 9.5:
- You can use new database objects called audit policies to control audit configuration within a database.

  Individual databases can have their own audit configurations, as can particular objects within a database, such as tables, or even users, groups, and roles. In addition to providing easier access to the information that you need, this enhancement also improves performance, because less data needs to be written to disk.
- Auditing SQL statements is easier and produces less output.

  The new audit category, EXECUTE, allows you to audit just the SQL statement that is being run. Previously, you needed to audit the CONTEXT event to capture this detail.
- Audit logs exist for each database.

  There is now one audit log for the instance and one audit log for each database. This feature simplifies audit reviews.
- The audit log now has a customizable path.

  Control over the audit log path allows you to place audit logs on a large, high-speed disk, with the option of having separate disks for each node in a database partitioning (DPF) installation. This feature also allows you to archive the audit log offline, without having to extract data from it until necessary.
- You can archive audit logs.

  Archiving the audit log moves the current audit log to an archive directory, while the server begins writing to a new, active audit log. When you extract data from an audit log to a database table, it is from an archived log, not the active audit log. This prevents performance degradation caused by locking of the active audit log.
- The security administrator (who holds SECADM authority) now manages the audit for each database.

  The security administrator is solely in control of configuring an audit for a database; the system administrator (holding SYSADM authority) no longer has this authority. The security administrator also has sufficient access to manipulate the audit log, issue the ARCHIVE command, and extract a log file into a table.
- You can audit new information in each category.

The CURRENT CLIENT special registers allow values for a client user ID, accounting string, workstation name, and application name to be set within applications so that these values will be recorded in the audit data.

The local and global transaction IDs can be recorded in audit data. This facilitates correlation between the audit log and the transaction log.

Related concepts

"Audit policies" in Database Security Guide

"The EXECUTE category for auditing SQL statements" in Database Security Guide

"Storage and analysis of audit logs" in Database Security Guide

Related reference

"CREATE AUDIT POLICY statement" in SQL Reference, Volume 2

"AUDIT statement" in SQL Reference, Volume 2

Roles simplify privilege administration and management

Roles simplify the administration and management of privileges by offering a capability equivalent to groups but without the same restrictions.

A role is a database object that groups together one or more privileges. You can assign a role to users, groups, PUBLIC, or other roles by using a GRANT statement or assign a role to a trusted context by using a CREATE TRUSTED CONTEXT or ALTER TRUSTED CONTEXT statement. You can specify a role for a SESSION_USER ROLE connection attribute in a workload definition.

Roles provide several advantages:
- You can control access to your databases in a way that mirrors the structure of your organization (you can create roles in the databases that map directly to the job functions in your organizations).
- You can grant users membership in the roles that reflect their job responsibilities. As their job responsibilities change, you can easily grant users access to a new role and revoke access the old roles.
- The assignment of privileges is simplified. Instead of granting the same set of privileges to each individual user in a particular job function, you can grant this set of privileges to a role representing that job function and then grant that role to each user in that job function.
- If you updated the privileges of a role, all users to whom you granted that role receive the update; you do not need to update the privileges for every user on an individual basis.
- The privileges and authorities that you granted to roles are always used when you create views, triggers, materialized query tables (MQTs), static SQL, and SQL routines, whereas privileges and authorities that you granted to groups (directly or indirectly) are not used.

  This is because the DB2 database system cannot determine when membership in a group changes because the group is managed by third-party software (for example, the operating system). Because roles are managed inside the database, the DB2 database system can determine when authorization changes and act accordingly. Roles granted to groups are not used because they are managed externally.

- All of the roles that you assigned to a user are enabled when that user establishes a connection, so all privileges and authorities granted to roles are taken into account when a user connects. You cannot explicitly enable or disable roles.
- The security administrator can delegate management of a role to others.

Related concepts

"Roles" in Database Security Guide

Related reference

"CREATE ROLE statement" in SQL Reference, Volume 2

Label-based access control (LBAC) enhancements provide better security

Label-based access control (LBAC) enhancements have been introduced to allow you to grant security labels and exemptions to roles as well as groups. Also, you can now add new elements to security label components, and you can modify security policies to change behaviors or add components to their security labels.

LBAC enhancements are as follows:
- The new ALTER SECURITY LABEL COMPONENT statement allows you to add a new element to a security label component.
- The new ALTER SECURITY POLICY statement allows you to modify a security policy. You can add a component to a defined security policy, which cannot be in use by a table while you update the policy. In addition, you can use the statement to enable or disable the granting of security labels and exemptions and to alter the write authorization error behavior for a security policy.
- The GRANT SECURITY LABEL statement now allows you to grant security labels to roles and groups; the REVOKE SECURITY LABEL statement now allows you to revoke security labels from roles and groups.
- The GRANT EXEMPTION statement now allows you to grant exemptions to roles and groups; the REVOKE EXEMPTION statement now allows you to revoke exemptions from roles and groups.

Related concepts

"Label-based access control (LBAC) overview" in Database Security Guide

"LBAC security policies" in Database Security Guide

"LBAC security label components overview" in Database Security Guide

"LBAC security labels" in Database Security Guide

"LBAC rule exemptions" in Database Security Guide

Chapter 7. Performance enhancements

DB2 Version 9.5 contains numerous performance enhancements, providing radical improvements in query time for complex queries involving time series, spatial data, and sliding query windows.

Review this section to find out more details about performance enhancements included in Version 9.5.

Queries that reference LOB columns run faster

Performance is improved in Version 9.5 for queries that return a result set containing many rows of data that include LOB columns.

These performance improvements are due to changes that allow multiple rows of data to be blocked together and returned as a result set to the client for a single cursor request if the result set contains LOB data. Blocking of row data that contains references to LOB data types is supported in all environments. Previously, blocking of row data was restricted to result sets without LOB columns.

Additionally, the Version 9.5 server supports Dynamic Data Format, also known as progressive streaming, which allows the server to return LOB values in the most optimal manner. This should reduce the length of time taken to retrieve rows from a cursor with LOB columns. Most IBM data server clients have supported the Dynamic Data Format feature since Version 9.1 Fix Pack 1 and automatically take advantage of it to retrieve LOB values using progressive streaming.

Improved row blocking for LOB values is limited as follows:
- Queries in embedded SQL applications that reference LOB columns might show little or no improvement.
- The presence of any user-defined function that outputs a LOB value within a query disables cursor blocking.

This performance feature is also available on DB2 for z/OS and DB2 for i5/OS.

> **Related concepts**
>
> "Progressive streaming with the IBM Data Server Driver for JDBC and SQLJ" in Developing Java Applications

Optimistic concurrency control and update detection enhancements provide a scalable locking alternative

Enhancements to optimistic concurrency control provide a more scalable locking alternative to database locking for concurrent data access. This replaces and performs better than values-based optimistic locking. A related enhancement provides a mechanism to detect recent (daily, weekly, or monthly) database updates.

Optimistic locking and optimistic concurrency control

Optimistic locking minimizes the time for which a given resource is unavailable for use by other transactions. Because the database manager can determine when a row is changed, it can ensure data integrity while limiting the time that locks are

held. With optimistic concurrency control, the database manager releases the row or page locks immediately after a read operation.

Optimistic concurrency control is supported by the use of RID_BIT() and ROW CHANGE TOKEN expressions on the initial select list, allowing the original values to be supplied as predicates for the searched update.

Daily, weekly, and monthly update detection

As a database administrator, you benefit from knowing the volume of updates within specific time ranges so that you can plan for data replication, create auditing scenarios, and so on.

A new expression, ROW CHANGE, that you use with SELECT, INSERT, and UPDATE statements, returns a token or a timestamp that represents the last change to a row. Now, an application has the following options:

- Determine when a row was last changed (or changed within a range of dates or number of days) by using the ROW CHANGE TIMESTAMP expression
- Return a token as a BIGINT (big integer) value that represents a relative point in the modification sequence of a row by using the ROW CHANGE TOKEN expression

  Related concepts

  "Optimistic locking" in Data Servers, Databases, and Database Objects Guide

  "Time-based update detection" in Data Servers, Databases, and Database Objects Guide

  Related reference

  "Optimistic locking overview" in Data Servers, Databases, and Database Objects Guide

MDC rollout deletions are faster with the deferred index cleanup option

You can now specify that the cleanup of record ID (RID) indexes is to take place after a rollout delete on a multidimensional clustering (MDC) table is complete. This deferred index cleanup option significantly improves the speed of certain rollout delete operations.

Since Version 8.2.2, you have been able to take advantage of rollout deletion, which provides the dual benefits of faster, block-based deletes and reduced logging. The rollout efficiency is based on knowing which blocks of an MDC table contain rows that are to be deleted and that all rows in a particular block will be deleted. However, the performance benefit of a rollout prior to Version 9.5 depended on the amount of RID indexes that you had on your table because rollouts prior to Version 9.5 required row-level processing and logging for each index.

Version 9.5 further enhances the performance of rollout deletion by allowing you the option of deferring RID index cleanup. This makes delete operations that occur on dimensional boundaries much faster because delete processing no longer removes index keys that reference the table rows being deleted. DB2 marks blocks as rolled out without updating the RID indexes until after the transaction commits and then cleans up these indexes asynchronously.

The improved speed of deferred cleanup for your rollout deletions depends on the kinds of deletes that you are performing. If there are a number of RID indexes on your table or if the size of your delete is quite large, deferred cleanup significantly

reduces deletion times. As well, if you have several rollout deletes planned for a particular table, it is much faster to have a single cleanup of RID indexes performed on all of the blocks that have been rolled out.

There are two ways to enable the new feature. You can set the existing **DB2_MDC_ROLLOUT** registry variable, which is now dynamic, to its new value, DEFER. Alternatively, you can set the CURRENT MDC ROLLOUT MODE special register to DEFERRED by using the SET CURRENT MDC ROLLOUT MODE statement. A new database monitor element, **BLOCKS_PENDING_CLEANUP**, allows you to determine the number of MDC table blocks that are pending cleanup.

The default behavior for deletes that qualify for rollout continues to be rollout with an immediate index cleanup. With the added feature, you can decide when a deferred index cleanup is needed. Because **DB2_MDC_ROLLOUT** is dynamic, any new compilations of your DELETE statement use the new setting. However, you will probably find using the CURRENT MDC ROLLOUT MODE special register a way to more finely control rollout behavior.

Related concepts

″Optimization strategies for MDC tables″ in Partitioning and Clustering Guide

″Asynchronous index cleanup for MDC tables″ in Data Servers, Databases, and Database Objects Guide

Related reference

″CURRENT MDC ROLLOUT MODE special register″ in SQL Reference, Volume 1

″SET CURRENT MDC ROLLOUT MODE statement″ in SQL Reference, Volume 2

″blocks_pending_cleanup - Pending cleanup rolled-out blocks monitor element″ in System Monitor Guide and Reference

Index build parallelism is enabled by default

In Version 9.5, index build parallelism is enabled regardless of the setting of the **intra_parallel** configuration parameter.

The performance of the CREATE INDEX statement can be improved by using multiple processors to scan and sort the index data in parallel. The index manager component decides whether to parallelize the index creation and bases its decision on many considerations, including table size and whether multiple processors are present. The decision is no longer controlled by the **intra_parallel** configuration parameter.

Related concepts

″Parallelism and loading″ in Data Movement Utilities Guide and Reference

″Optimization strategies for intra-partition parallelism″ in Partitioning and Clustering Guide

Related reference

″max_querydegree - Maximum query degree of parallelism configuration parameter″ in Data Servers, Databases, and Database Objects Guide

″intra_parallel - Enable intra-partition parallelism configuration parameter″ in Data Servers, Databases, and Database Objects Guide

OLAP functions have been enhanced

New and improved OLAP functions permit finer control over queries, particularly aggregations performed on large *sliding windows* of data.

The built-in OLAP functions have been improved in three ways:
- You can use four new functions. New LEAD, LAG, FIRST_VALUE, and LAST_VALUE OLAP functions enable finer control over aggregations performed on windows of rows that move as table rows are read (*sliding windows*). These new functions provide access to specific values from rows other than the current row. For example, if an OLAP function window represents the trades for a stock during a specific period of time, these new OLAP functions can return the first and last trades during the period (perhaps the open and closing prices). Also, if calculating a 150-day moving average of a close price for a stock that was not traded for a particular day, financial institutions might prefer to include in the moving average calculation the close price from some other day, either before or after the day where there was no trade. This has the effect of *smoothing* the calculations by compensating for missing (NULL) values used in the calculation.
- Performance of existing OLAP functions when you apply them to sliding windows of data has improved. These performance increases can be dramatic, resulting in some queries running hundreds of times faster than in previous versions.
- OLAP memory use has been optimized. The memory used by OLAP functions can be dynamically tuned by the self tuning memory, and OLAP functions are now less memory constrained. These optimizations allow queries against much larger windows of data. Available temporary space, rather than system memory, is now the limiting factor for many queries.

  Related reference

  "OLAP specifications" in SQL Reference, Volume 1

Query optimizer has been enhanced

Complex queries have been optimized, particularly query types of interest to the financial sector.

You can gain substantial performance improvements for the following types of queries:
- Queries that involve multiple distinct aggregations in a single SELECT statement
- Queries that involve *ORDER BY* and *FETCH FIRST n ROWS ONLY* semantics
- Queries with MIN and MAX functions with GROUP BY clauses
- Queries with large IN list predicates (searching through the list has been improved, and more strategies for evaluating such predicates are considered at query optimization time)

  Related concepts

  Chapter 7, "Performance enhancements," on page 53

  "OLAP functions have been enhanced"

NO FILE SYSTEM CACHING default reduces file system cache memory

For table space containers that you create in Version 9.5, by default the database manager attempts to use Concurrent I/O (CIO) wherever possible. On the system configurations where CIO is not supported, Direct I/O (DIO) or buffered I/O is used instead.

CIO and DIO improve memory performance because these settings allow the database manager to bypass caching at the file system level. This process reduces CPU overhead and makes more memory available to the database instance. For the system configurations that support CIO, DIO, or file system caching, see "File system caching configurations".

The FILE SYSTEM CACHING and NO FILE SYSTEM CACHING attributes specify whether I/O operations are to be cached at the file system level. Because the database manager manages its own data caching using buffer pools, caching at the file system level is not needed if the size of the buffer pool is tuned appropriately.

The new default does not apply to table spaces that you created prior to Version 9.5.

Combined with the enablement of the self-tuning memory manager and the AUTOMATIC buffer pool size setting on the ALTER BUFFERPOOL statement, the new default provides the following benefits:

- You do not need to explicitly specify NO FILE SYSTEM CACHING on the CREATE TABLESPACE statement to take advantage of the benefits of non-buffered I/O.
- It reduces the consumption of the file system cache because data is automatically cached at the buffer pool level. This, in turn, reduces the amount of memory dedicated to the file system cache.

Related concepts

"Management of multiple database buffer pools" in Tuning Database Performance

"Self-tuning memory" in Data Servers, Databases, and Database Objects Guide

"File system caching configurations" in Data Servers, Databases, and Database Objects Guide

Related tasks

"Creating table spaces" in Data Servers, Databases, and Database Objects Guide

Related reference

"ALTER BUFFERPOOL statement" in SQL Reference, Volume 2

"CREATE TABLESPACE statement" in SQL Reference, Volume 2

DB2 Spatial Extender query performance has improved

DB2 Spatial Extender generates and analyzes spatial information about geographic features, and stores and manages the data on which this information is based. DB2 Spatial Extender query performance has drastically improved.

You can now more efficiently query spatial data about customer locations, cell-phone towers, roads, towns, or other objects that is generally stored as LOB data. This is due to improvements in data blocking between the underlying DB2 database client and the server for queries referencing LOB columns. Faster retrieval of this data contributes to faster drawing and displaying of spatial data maps by third-party software. For example, using ESRI ArcGIS prior to Version 9.5 to draw a map over a moderate latency network link took over 2 minutes. Using ArcGIS in Version 9.5, with the new LOB blocking support enabled, the same map is displayed in 4 seconds.

Related concepts

"The purpose of DB2 Spatial Extender" in Spatial Extender and Geodetic Data Management Feature User's Guide and Reference

Additional statements can be explained

In Version 9.5, the REFRESH TABLE and SET INTEGRITY statements have been added to the list of explainable statements that you use to collect and analyze explain information to diagnose performance problems. This enhancement makes it easier for you to maintain your materialized query tables (MQTs).

You can now obtain the access plan to self-diagnose performance problems with the SET INTEGRITY and REFRESH TABLE statements.

Related reference

"EXPLAIN statement" in SQL Reference, Volume 2

"REFRESH TABLE statement" in SQL Reference, Volume 2

"SET INTEGRITY statement" in Data Movement Utilities Guide and Reference

"CURRENT EXPLAIN MODE special register" in SQL Reference, Volume 1

"CURRENT EXPLAIN SNAPSHOT special register" in SQL Reference, Volume 1

"EXPLAIN_STATEMENT table" in SQL Reference, Volume 1

"Explain tables" in SQL Reference, Volume 1

Chapter 8. pureXML enhancements

Version 9.5 builds on the pureXML support first introduced in Version 9.1 and enhances the hybrid relational and XML data server to make your XML data processing even more flexible, faster, and even more reliable.

Review this section to find out more details about pureXML enhancements included in Version 9.5.

Related concepts

″pureXML overview″ in pureXML Guide

″Tutorial for pureXML″ in pureXML Guide

″XML input and output overview″ in pureXML Guide

Related tasks

″Converting non-Unicode databases to Unicode″ in Internationalization Guide

XQuery updating expressions allow portions of XML documents to be modified

New XQuery updating expressions enable you to modify portions of an existing XML document instead of having to construct a new one.

For example, if you have purchase orders that are stored as XML documents in an XML column in a DB2 database, you can use DB2 XQuery updating expressions to add new line item elements to the purchase orders. You do not have to re-create the purchase orders with the new line items. The updating expressions can make the changes to the XML documents without converting the XML data to another format.

DB2 XQuery updating expressions always work on a copy of the XML data created by the XQuery transform expression. The transform expression consists of three clauses: the copy clause, the modify clause, and the return clause. The copy clause creates a copy of the XML data, the modify clause contains the XQuery updating expressions that change the copy, and the return clause can contain other non-updating expressions that can further process the changed copy of the XML data.

DB2 XQuery expressions, including the updating expressions, are based on the XQuery language described in W3C documents. The expressions operate on instances of the XQuery and XPath data model (XDM). The XDM allows XQuery to operate on the abstract, logical structure of an XML document or fragment, rather than its syntax visible in a text file. The inputs (if any) of an XQuery expression are instances of the XDM, and the result of an expression is also an instance of the XDM. XML documents are converted into the XDM when they are stored in an XML column.

Related information

″Use of updating expressions in a transform expression″ in XQuery Reference

Load utility support for pureXML has been added

In addition to the import and export utilities that became available for XML data in Version 9.1, you can now use the high performance load utility to insert XML documents into DB2 tables.

The load utility is especially useful when large amounts of data need to be inserted into a table within a narrow time frame. Because the load utility writes formatted pages directly into the database, loading data can be much faster than importing. Also, minimal logging is performed during load operations, allowing for more efficient data movement. The ability to load XML data also allows you to use load-specific options on tables that contain XML columns, such as loading from the CURSOR file type, and a variety of data manipulation features.

Related concepts

"Loading XML data" in Data Movement Utilities Guide and Reference

Related reference

"Differences between the import and load utility" in Data Movement Utilities Guide and Reference

pureXML application processing performance has been improved

The performance of the pureXML feature has improved. Execution times and, in some cases, resource usage for applications processing XML data have been reduced.

The broad scope of the improvements covers data manipulation in both SQL/XML and XQuery, index over XML data creation, the query compiler and optimization, XML document navigation, and other areas. Following are more specific examples of improvements:

- Insert, update, and delete operations can run more quickly.
- Index ANDing during query evaluation can include both indexes over XML data and relational indexes.
- Queries over UNION ALL views can use indexes over XML data.
- Complex access plans require fewer NLJOIN operators during query evaluation.

New performance guidelines are included with the pureXML topics.

Related concepts

"Explain facility" in Tuning Database Performance

Check constraints functionality has been extended

You can now specify additional options with check constraints on XML columns to ensure the consistency of information before it is processed.

A check constraint allows you to place certain restrictions on an XML column. The constraint is enforced whenever an attempt is made to insert or update data in the XML column; only when the criteria specified by the constraint evaluate as true is the operation performed.

You can now require that an XML value either is or is not validated when using the VALIDATED predicate, optionally including one or several registered XML schemas by specifying an ACCORDING TO XMLSCHEMA clause.

You can now also specify column constraints with a correlation name of type XML as part of a BEFORE trigger.

Related concepts

"Check constraints on XML columns" in pureXML Guide

Trigger processing supports automatic validation of XML documents

Trigger processing now supports automatic validation of XML documents against registered XML schemas based on the current validation state of the documents.

Validation of XML documents against registered XML schemas before the documents are stored in an XML column is optional, but you should do it when data integrity is uncertain because it ensures that only valid XML documents are inserted or updated.

For automatic validation of XML documents against registered XML schemas, BEFORE triggers can specify NEW AS correlation names of type XML to invoke the XMLVALIDATE function from a SET statement, to set values to NULL, or to leave values of type XML unchanged.

To determine whether validation of an XML document against an XML schema must be triggered, the WHEN clause of the BEFORE trigger can test for the validation state of the document by including the search condition IS VALIDATED or IS NOT VALIDATED, optionally including one or several XML schemas by specifying the ACCORDING TO XMLSCHEMA clause.

Related concepts

"Trigger processing of XML data" in pureXML Guide

XSLT support enables transforming XML data into other formats

You can use the new XSLTRANSFORM function to convert XML documents residing in the database into HTML, plain text, or other forms of XML.

XSLT transformation is the standard method used to transform XML data into other formats and it allows you to generate multiple output formats from a single data source. This functionality is similar to the XSLT transforms provided by XML Extender.

XSLTRANSFORM uses style sheets to convert XML into other data formats. You can convert part or all of an XML document and select or rearrange the data using the XPath query language and the built-in functions of XSLT.

An important feature of the XSLTRANSFORM function is its ability to accept XSLT parameters at run time. In the absence of this ability, you would need to maintain a large library of XSLT style sheets, one for each variant of a query against the XML data, or you would need to manually edit your style sheets for each new kind of query. Parameter passing allows you to separate the standardized behaviors of your style sheets from the customized or unique requirements of a particular query. After carefully designing your core style sheets, you can leave them alone and offload any custom behavior into parameter files.

Related concepts

"Transforming with XSLT stylesheets" in pureXML Guide

SQL/XML and XQuery parameter passing is more flexible

Parameter passing is simplified and extended for both SQL/XML and XQuery to make it more flexible.

When issuing combined SQL statements and XQuery expressions, you can now pass data between the SQL statements and XQuery expressions as follows:

Passing parameters from SQL
- By default, you do not need to explicitly specify parameters for the XMLQuery scalar function, the XMLExists predicate, and the XMLTable table function.
- By default, you do not need to specify columns for the XMLTable table function.

Passing parameters from XQuery to SQL
- You can pass parameters to an SQL fullselect within an XQuery expression using the db2-fn:sqlquery function. You can use the parameters to modify the SQL fullselect executed by the XQuery expression and to modify the data returned to the XQuery expression.

Related concepts

"Constant and parameter marker passing to XMLEXISTS and XMLQUERY" in pureXML Guide

"Simple column name passing with XMLEXISTS, XMLQUERY, or XMLTABLE" in pureXML Guide

"Passing parameters from XQuery to SQL" in pureXML Guide

Related information

"sqlquery function" in XQuery Reference

Non-Unicode databases can store XML data

You can now store XML data in a non-Unicode database.

Because the DB2 database manager stores XML data internally as Unicode, prior to Version 9.5, the features of pureXML were available only in a UTF-8 encoded database. New functionality manages code page conversion so that you no longer need a Unicode database. However, insert XML data into a non-Unicode database only in a form that does not undergo code page conversion (for example, BIT DATA, BLOB, or XML). To avoid introducing substitution characters into the data, insert only XML data that contains code points that are included in the code page of the database.

Use the new **enable_xmlchar** configuration parameter to prevent possible character substitution from occurring as SQL string data is converted from the client code page into the database code page, and then into Unicode for internal storage. Setting **enable_xmlchar** to NO blocks the use of character data types during XML parsing, preventing possible character substitution and ensuring the integrity of stored XML data. By default, **enable_xmlchar** is set to YES so that parsing of character data types is allowed.

Non-Unicode XML databases can be managed using the Control Center, like any other databases.

Related concepts

"Using XML in a non-Unicode database" in pureXML Guide

Small XML documents can be stored in base table row for improved performance

An additional storage choice is available for XML documents with a size of 32 KB or less. When you add XML columns to a table or change existing XML columns, you can store these documents in a row of the base table instead of in the default XML storage object.

Row storage of XML documents is similar to how a structured type instance can be stored inline in the row of a table, and is under your control. Which storage choice you make depends on your storage and performance requirements; larger documents must always be stored in the default XML storage object, but if you work mostly with small documents, base table row storage can provide you with the following benefits:

- Increased performance for any operation that queries, inserts, updates, or deletes XML documents, because fewer I/O operations are required for documents stored in base table rows.
- Reduced storage space requirements and improved I/O efficiency for XML documents if you also use data row compression.

You use the option by including the INLINE LENGTH keywords with CREATE TABLE and ALTER TABLE statements, followed by the maximum size of XML documents to store in the base table row. When you store XML documents larger than 32 KB in an XML table column that is enabled for base table row storage, the oversize documents are stored transparently in the default XML storage object.

> **Related concepts**
>
> "XML storage" in pureXML Guide
>
> "XML base table row storage" in pureXML Guide

XML schemas can be updated without requiring re-validation of XML documents

You can now update an XML schema registered in the XML schema repository so that both previously inserted and new XML documents validate against the evolved version of the schema.

An UPDATE XMLSCHEMA command and an XSR_UPDATE stored procedure have been added, allowing you to modify an XML schema that has already been registered. This is useful in cases where, for example, you want to add optional elements or attributes to an existing schema, and you also want the updated schema to be available to validate both new and previously stored XML instance documents.

The key requirement for evolving an XML schema is that the original and new schemas are compatible, so that previously validated XML documents will continue to be valid and the type annotations in the original schema do not change. During the update process, a check is performed to confirm the compatibility of the original and new schemas. The update operation will fail if the schemas are not compatible.

> **Related concepts**
>
> "Scenario: Evolving an XML schema" in pureXML Guide
>
> **Related tasks**
>
> "Evolving an XML schema" in pureXML Guide

Related reference

"Compatibility requirements for evolving an XML schema" in pureXML Guide

XQuery upper-case and lower-case functions support locales

The XQuery functions fn:upper-case and fn:lower-case can now change the case of text using a locale-aware conversion. By default, fn:upper-case and fn:lower-case convert the case of text based on the Unicode standard. For some characters, there is a different mapping between uppercase and lowercase characters when using a locale-based conversion than when using the Unicode standard.

For example, in Turkish, when changing the case of the character i or I (uppercase or lowercase Latin I), the dot remains with the character. When you specify the Turkish locale, tr_TR, fn:upper-case converts the character i to a Latin uppercase I with a dot above. The numeric character reference is İ. When you specify the Turkish locale, fn:lower-case converts the character I to a Latin small letter dotless i. The numeric character reference is ı. Without the locale specified, fn:upper-case converts the character i to I, and fn:lower-case converts the character I to i.

Related information

"lower-case function" in XQuery Reference

"upper-case function" in XQuery Reference

XQuery functions extract components from and adjust dates and times

You can now use XQuery date and time component extraction functions to extract portions of date, time, dateTime, and duration data types. You can now use XQuery time zone adjustment functions to adjust dates and times to a specified time zone or remove the time zone component from a date, time, or dateTime data type.

For example, using date extraction functions, you can easily extract the year and month component from a date data type. Using the time zone adjustment functions, you can easily convert a date or time from one time zone to another.

Related information

"Functions by category" in XQuery Reference

XQuery castable expression supports testing the casting of values

The XQuery language castable expression allows you to test the casting of values to XQuery data types.

You can use the castable expression as an XPath predicate to avoid errors at evaluation time. You can also use the castable expression to select an appropriate data type when processing a given value.

Related information

"Castable expressions" in XQuery Reference

Publishing functions are easier to use

New publishing scalar functions are available for mapping relational data to XML. These functions require fewer options than previous SQL/XML publishing functions required and provide default behaviors to leverage many of the rules defined by ISO SQL/XML 2006 or to support the most commonly needed options.

64 What's New

The new publishing functions are as follows:

XMLGROUP

This function returns a single top-level element to represent a table or the result of a query. By default, each row in the result set is mapped to a row subelement, and each input expression is mapped to a subelement of the row subelement. Optionally, each input expression can be mapped to an attribute of the row subelement.

XMLROW

This function returns a sequence of row elements to represent a table or the result of a query. By default, each input expression is transformed into a subelement of a row element. Optionally, each input expression can be transformed into an attribute of a row element.

XSLTRANSFORM

This function transforms XML documents into HTML, plain text, or other forms of XML, based on style sheets that you supply. XSLT transformation is the standard method used to transform XML data into other formats and it allows you to generate multiple output formats from a single data source.

Related reference

"XMLROW scalar function" in SQL Reference, Volume 1

"XMLGROUP aggregate function" in SQL Reference, Volume 1

"XSLTRANSFORM scalar function" in SQL Reference, Volume 1

Annotated XML schema decomposition supports order of insertion and registration of recursive schemas

Two new features have been added to the decomposition function for pureXML, allowing control over the order of insertion and the registration of recursive XML schemas.

Support for order of insertion

New XML schema annotations allow you to specify a decomposition hierarchy, such that the contents of an XML document are inserted into the rows of a target table in a determinable order. Prior to Version 9.5, there was no facility to control the order in which data from the decomposition process was inserted as rows into target tables, so inserting data in a way that met any consistency requirements of the target tables might have been difficult. The new feature provides a way to ensure that the referential integrity constraints defined in a relational schema are respected during XML document shredding.

Registration of recursive schemas

Prior to Version 9.5, it was impossible to register an XML schema containing recursion, even if the recursive section of the XML instance document did not need to be decomposed. XML schemas containing recursion can now be registered in the XML schema repository (XSR) and enabled for decomposition. The recursive sections of an associated XML instance document cannot be decomposed as scalar values into a target table. However, by using appropriate schema annotations, the recursive sections can be stored and later retrieved as serialized markup.

Related concepts

"Annotated XML schema decomposition and recursive XML documents" in pureXML Guide

Related reference

"db2-xdb:rowSetOperationOrder decomposition annotation" in pureXML Guide

"db2-xdb:order decomposition annotation" in pureXML Guide

Chapter 9. Application development enhancements

Application development improvements include new features and enhancements that simplify database application development, improve application portability, and ease application deployment.

Review this section to find out more details about application development enhancements included in Version 9.5.

Identifier length limits have increased

Large identifier support lets you more easily port applications from other DBMS vendors. You will also find it easier to migrate data definition language (DDL) because you no longer need to shorten identifiers.

Identifiers with longer maximum length are listed in the following table:

Table 3. Identifier length limits in Version 9.1 and 9.5

Identifier Name	Length in Version 9.1 (bytes)	Length in Version 9.5 (bytes)
Attribute	18	128
Authorization ID (Authid)	30	128
Column	30	128
Constraint	18	128
Cursor	18	128
Database partition group	18	128
Event monitor	18	128
Group	30	128
Package	8	128
Schema	30	128
Specific name	18	128
SQL path (specified by the FUNCPATH BIND option and CURRENT PATH special register)	254	2048
Statement	18	128
Trigger	18	128
User-defined type	18	128

Note that the 128-byte limit applies only to non-embedded SQL only because the SQLDA is still limited to 8-byte schema names for user-defined types (UDTs), 18-byte names for UDTs, and 30-byte names for columns.

The 128-byte limit refers to the limit that is stored in the system catalog by the database manager. Because the code page that is used to represent an identifier in an application can vary, the limit on the application side is undefined. DB2 utilities on both the application and the server side use a limit of 128 bytes regardless of the application code page.

You can find a useful sample file, checkv9limits in samples/admin_scripts. You can use this file to find identifiers in a database that might be using the larger Version 9.5 limits.

Related reference

″SQL and XML limits″ in Data Servers, Databases, and Database Objects Guide

PHP extensions have been integrated into DB2 installation (Linux, AIX, and Windows)

In Version 9.5, the IBM DB2 Data Server Client (formerly, the DB2 Client) comes with Hypertext Preprocessor (PHP) extensions; you no longer need to download them. Also, Version 9.5 builds on the Version 9.1 PHP support by providing a new extension called PDO_IBM.

The new PDO_IBM and the existing IBM_DB2 extensions are small footprint extensions that enable robust access through your PHP applications to data stored in your DB2 databases and allow you to quickly and easily deploy your PHP applications. The differences between the extensions are as follows:

PDO_IBM

This new extension provides access to a DB2 database through the PHP Data Objects (PDO) interface. PDO provides a generalized, object-oriented interface to access your data. With this extension, you no longer need to create an ODBC data source to be able to use PDO. You can use PDO_IBM to connect to your database using either a cataloged connection or a direct TCP/IP connection to the DB2 database manager.

IBM_DB2

This extension provides direct access to data stored in your DB2 database by using the DB2 Call Level Interface (CLI) libraries. The interface of this extension is specific to the DB2 product and uses some of the more advanced DB2 features that are not available with other extensions. The IBM_DB2 extension provides an application programming interface (API) with extensive access to database metadata.

The DB2 installation programs contain the PHP extensions on the following operating systems:
- AIX
- Linux on x86
- Linux on AMD64 and Linux on EM64T
- Linux on POWER™ (PowerPC® and pSeries®)
- Windows on x86

Related concepts

″Introduction to PHP application development for″ in Getting Started with Database Application Development

Ruby on Rails framework support has been integrated into DB2 installation (Linux, AIX, and Windows)

Rapid development and deployment of DB2 Web applications have been enabled and enhanced for Ruby and Ruby on Rails applications.

The Rails adapter (IBM_DB) and the Ruby driver are developed by IBM and have been optimized for all DB2 data servers, including DB2 Database for Linux, UNIX,

and Windows, DB2 for i5/OS using DB2 Connect, and DB2 for z/OS using DB2 Connect. IBM is the only vendor that provides enablement and support for Ruby on Rails.

Although you can configure a variety of data stores in the Rails framework, the IBM DB2 data server has unique advantages. Using the IBM_DB Rails adapter and Ruby driver with the DB2 data server allows you to manipulate pureXML data. Also, the Rails framework provides features such as model relationships, Rake tasks, migration, scaffolding, and an integrated testing environment, which make iterative, agile application development a reality.

To enable rapid application development and deployment, the new DB2 Ruby driver and Rails adapter have been provided in the DB2 installation directory on the following operating systems:

- AIX (APAR IZ01456 must be applied)
- Linux on x86
- Linux on AMD64 and Linux on EM64T
- Linux on POWER (PowerPC and pSeries)
- Windows on x86

The DB2 Ruby driver and Rails adapter are also available on the RubyForge Rails Adapter/Driver for IBM Databases Web site.

Related concepts

"The IBM_DB Ruby driver and Rails adapter" in Getting Started with Database Application Development

Perl driver supports pureXML and multibyte characters

The DB2 Perl driver has been enhanced by adding two new significant features: pureXML support and multibyte locale support. These new features allow more direct access to your data through the DB2 Perl driver.

The new features help to decrease application logic by providing more transparent communication between your application and database. Details of these new features are as follows:

pureXML support

With pureXML support, you can directly insert XML documents into your DB2 database. Your application no longer needs to parse XML documents because the pureXML parser is automatically run when you insert XML data into the database. Having document parsing handled outside your application improves application performance and reduces maintenance efforts. Retrieval of XML stored data with the DB2 Perl driver is easy as well; you can access the data using a BLOB or record.

Support for multibyte character sets

This feature provides a more transparent interface between your application and your DB2 database. Your Perl application no longer needs to convert between character sets before interacting with your DB2 database. Removing the need to convert results in an application that has a smaller footprint, requires less maintenance and is less error prone.

For information on how to download the latest DB2 Perl driver, go to the http://www.ibm.com/software/data/db2/perl/ Web site.

Related concepts

"Programming Considerations for Perl" in Developing Perl and PHP Applications

IBM Database Add-Ins for Visual Studio 2005 has been enhanced

IBM Database Add-Ins for Visual Studio 2005, which provides tools for rapid application development, database schema development, and debugging, offers even better support in Version 9.5.

Enhancements are as follows:

- IBM Database Add-Ins for Visual Studio 2005 now supports IBM Informix®️ Dynamic Server, IBM UniData®️, and IBM UniVerse®️. This support enables you to use the same add-ins tool when working with any combination of DB2, Informix, UniData, or UniVerse products. Specifically, your .NET applications can now access the following IBM data servers:

  - DB2 Universal Database for Linux, UNIX, and Windows, Version 8.1, or later
  - DB2 Universal Database for z/OS or OS/390®️, Versions 6 and 7
  - DB2 Universal Database for z/OS, Version 8, or later
  - DB2 Universal Database for iSeries™️ (using OS/400®️ Version 5 Release 1, or later)
  - DB2 for i5/OS (using i5/OS Version 5 Release 3, or later)
  - IDS Version 11.1
  - UniData 7.1, or later
  - UniVerse 10.2, or later

  Availability of features and data type compatibility varies depending on the data server used. For example, Informix Dynamic Server, UniData, and UniVerse data servers do not support the XML data type.

- The new IBM Function Designer provides an easier way for you to work with functions. With the designer, you can perform the following actions:

  - Create and alter functions
  - Create and alter roles and define access privileges for functions
  - Clone functions
  - View or create scripts for all functions

- When you define a data connection to a DB2 server, you can filter database tables based on table types. An option lets you select table types such as P (physical) and L (logical) and other platform- or database-specific table types.

- You can now use the add-ins to create Web services that expose database operations (SQL SELECT and DML statements, XQuery expressions, or calls to stored procedures) to client applications. You can also use the add-ins to deploy Web services to or undeploy Web services from a Web server.

- XML tooling has been enhanced to support the following functionality:

  - Annotating XML schemas in the IBM XML Schema Mapping Designer:

    - Selecting the results of running a Web service method as an XML schema source for mapping
    - Dragging tables for mapping from the Server Explorer to the mapping editor in the designer
    - Using the results of running a Web service method to test your map links after annotating an XML schema

  - Generating Web service methods for annotated XML schemas in the XML Schema Repository

- Generating client-side and server-side XML schema validation code
- Comparing two versions of an XML schema and reviewing their differences
- Generating client-side and server-side XSL transformation code

Related concepts

"DB2 integration in Visual Studio" in Developing ADO.NET and OLE DB Applications

Global variables improve data sharing between SQL statements

Version 9.5 introduces the concept of global variables, which are named memory variables that you can access and modify through SQL statements. Global variables enable you to share data between different SQL statements running in the same session (or connection) without the need for application logic to support this data transfer.

It is no longer necessary for applications that issue such statements to copy values from the output arguments (for example, host variables) of one statement to the input arguments of another statement. Moreover, SQL statements that are contained within the database system itself, such as the statements that define triggers and views, can now access this shared information.

Global variables help you to implement more complex, interactive models for data transfer within the database itself so that you don't have to put supporting logic inside your applications or SQL procedures. The defined privileges that are associated with global variables ensure that the security of the transferred data is not left up to you to enforce through application logic. If security is an issue, you can control access to global variables through GRANT and REVOKE statements.

Global variables are particularly useful for storing data that is static, that changes rarely during a session, or that exerts administrative control. Examples of such data are the pager number that is used to communicate alerts to a DBA and indicators of whether certain triggers are to be enabled or disabled.

Version 9.5 supports created session global variables. A session global variable is associated with a specific session and contains a value that is unique to that session. A created session global variable is available to any active SQL statement running against the database on which you defined the variable. The system catalog contains the definitions of created session global variables and the privileges that are associated with them.

Related reference

"Identifiers" in SQL Reference, Volume 1

"CREATE VARIABLE statement" in SQL Reference, Volume 2

"GRANT (Global Variable Privileges) statement" in SQL Reference, Volume 2

"REVOKE (Global Variable Privileges) statement" in SQL Reference, Volume 2

"Global variables" in SQL Reference, Volume 1

SET variable is now an executable statement that can be dynamically prepared

Version 9.5 introduces a SET statement which provides consistent methods for setting host variables, bind variables, global variables and local variables in triggers and functions, procedures and as a stand alone statement. The SQL statement can be dynamically prepared and executed, virtually eliminating the need for single row dynamic cursors.

The SET variable statement assigns values to variables. Prior to Version 9.5, this statement was allowed only in dynamic compound statements, triggers, SQL functions, SQL methods, SQL procedures, and UPDATE statements.

Because the SET variable statement can now be embedded in an application program or issued interactively, and because it is an executable statement that can be dynamically prepared, the entire SET variable syntax is supported by the command line processor (CLP) in applications and in SQL stored procedures. For example, you can now use multi-target SET variable statements, such as SET (a, b) = (1, 2), within SQL stored procedures.

You no longer need to use cursors to retrieve data that is unavailable at precompile (prep) time. Prior to Version 9.5, you needed to use the SELECT INTO statement if you wanted your SQL procedure to copy table data into host variables, as shown in the following example:

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

However, the SELECT INTO statement is valid only for static SQL; it cannot be dynamically prepared. So, if the SELECT statement depends on data that is unavailable at prep time, in the past, you had to do something like this:

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

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

Now, you can do something like this:

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

Related reference

"SET variable statement" in SQL Reference, Volume 2

Array support enhances application portability

Version 9.5 supports the ARRAY collection data type. Arrays are transient values that you can manipulate in stored procedures and applications but cannot store in tables. This feature makes it easier to port applications and stored procedures from other database vendors that already support arrays.

You can use arrays to efficiently pass collections of data between applications and stored procedures and to store and manipulate transient collections of data within

SQL procedures without having to use relational tables. Operators on arrays available within SQL procedures allow for the efficient storage and retrieval of data.

Support for array data types in Version 9.5 allows you to:
- Create user-defined types based on arrays; for example, `CREATE TYPE INT10 AS INTEGER ARRAY[10]` defines a type for arrays of up to 10 integer values;
- Declare variables and parameters of array types in stored procedures and applications;
- Create and manipulate array values; primitives for array manipulation include array constructors, subindexing, element count, and trim;
- Pass arrays back and forth between JDBC and CLI applications and SQL and Java stored procedures;
- Convert arrays to tables (one array element per table row) and aggregate columns into arrays for easy interfacing between arrays and SQL;
- Call procedures with input and output array parameters from the command line processor.

   Related reference

   "User-defined types" in SQL Reference, Volume 1

   "CREATE TYPE (Array) statement" in SQL Reference, Volume 2

Decimal floating-point data type improves the accuracy and performance of decimal data

Version 9.5 introduces DECFLOAT, a decimal floating-point data type that is useful in business applications (for example, financial applications) that deal with exact decimal values.

Binary floating-point data types (REAL and DOUBLE), which provide binary approximations for decimal data, are not appropriate in such applications. DECFLOAT combines the accuracy of DECIMAL with some of the performance advantages of FLOAT, which is beneficial in applications where monetary values are being manipulated. For example, computing a 5% tax on a $0.70 telephone call results in a calculated value of 0.734999987483025 if you define the column containing the price as REAL and a value of 0.7350 if you define the column as DECFLOAT(16).

Version 9.5 supports DECFLOAT on the new IBM POWER6™ hardware. On IBM Power 6 DB2 Version 9.5 exploits decfloat hardware acceleration previously known only for regular floating point arithmetic. Version 9.5 implements the 16-digit and the 34-digit decimal floating-point encodings, as specified by the IEEE standard. The following table shows the precision, length, and range of each encoding:

Table 4. Precision, length, and range of the supported decimal floating-point encodings

Precision in digits	Length in bits	Range
16	64	$-9.999999999999999 \times 10^{384}$ to -1×10^{-383} : 1×10^{-383} to $9.999999999999999 \times 10^{384}$
34	128	$-9.999999999999999999999999999999999 \times 10^{6144}$ to -1×10^{-6143} : 1×10^{-6143} to $9.999999999999999999999999999999999 \times 10^{6144}$

This data type supports the five IEEE-specified rounding modes: ROUND_HALF_EVEN, ROUND_HALF_UP, ROUND_DOWN, ROUND_CEILING,

and ROUND_FLOOR. You can specify the rounding mode through a new database configuration parameter, **decflt_rounding**.

> **Related reference**
>
> "Numbers" in SQL Reference, Volume 1
>
> "COMPARE_DECFLOAT scalar function" in SQL Reference, Volume 1
>
> "DECFLOAT scalar function" in SQL Reference, Volume 1
>
> "NORMALIZE_ DECFLOAT scalar function" in SQL Reference, Volume 1
>
> "QUANTIZE scalar function" in SQL Reference, Volume 1
>
> "TOTALORDER scalar function" in SQL Reference, Volume 1
>
> "decflt_rounding - Decimal floating point rounding configuration parameter" in Data Servers, Databases, and Database Objects Guide

JDBC and SQLJ support has been enhanced

The IBM Data Server Driver for JDBC and SQLJ contains a number of major enhancements for Version 9.5.

In Version 9.5, two versions of the IBM Data Server Driver for JDBC and SQLJ are available: one that supports functions in the JDBC 3.0 and earlier specifications and one that supports functions in the JDBC 4.0 and earlier specifications. The following table lists the files in which the driver versions are packaged and the corresponding levels of support:

Table 5. IBM Data Server Driver for JDBC and SQLJ packages

Driver package name	Level of JDBC support	Minimum level of SDK for Java required
db2jcc.jar and sqlj.zip	JDBC 3.0 and earlier	1.4.2
db2jcc4.jar and sqlj4.zip[1]	JDBC 4.0 and earlier	6

Note:

1. sqlj4.zip is added with DB2 Version 9.5 Fix Pack 1.

Review this section to find out more details about JDBC and SQLJ support enhancements included in Version 9.5.

JDBC 2.0 and JDBC 3.0 support has been enhanced

IBM Data Server Driver for JDBC and SQLJ Version 3.50, which provides support for JDBC 3.0 and earlier specifications, contains a number of enhancements.

Driver name is changed

The new driver name is the IBM Data Server Driver for JDBC and SQLJ. However, in Version 3.50 of the driver, the java.sql.DatabaseMetaData.getDriverName method returns `IBM DB2 JDBC Universal Driver Architecture`.

License files are no longer required

You do not need license files to access DB2 Database for Linux, UNIX, and Windows, Cloudscape™, or Informix Dynamic Server (IDS) data sources. Therefore, you no longer need to include the db2jcc_license_*.jar files in your **CLASSPATH** when you connect to these database servers.

This does not apply to DB2 Connect users.

runJDBCBinder method is added as an alternative to the DB2Binder utility

In previous versions of the IBM Data Server Driver for JDBC and SQLJ, only the DB2Binder command line interface was available for binding the DB2 packages that are used at the database server by the IBM Data Server Driver for JDBC and SQLJ. The runJDBCBinder method is an application programming interface for performing the same task.

The runJDBCBinder method supports the following options, which are equivalent to BIND options:
- **action** (add│replace│drop); drop is supported for DB2 for z/OS only
- **blocking** (all│no│unambig)
- **dbprotocol** (drda│private); dbprotocol is supported for DB2 for z/OS only
- **keepdynamic** (no│yes)
- **owner**
- **reopt** (none│always│once│auto)
- **size**
- **optprofile**; optprofile is supported for DB2 Database for Linux, UNIX, and Windows only

In addition, runJDBCBinder supports the size option. The size option specifies the number of internal JDBC driver packages to bind or drop for each isolation level and cursor holdability.

DB2Binder utility is enhanced

New options: The DB2Binder utility supports the following new options:

-action (drop)
      Indicates that existing IBM Data Server Driver for JDBC and SQLJ packages have been dropped

-size (*n*)
      Specifies the number of internal IBM Data Server Driver for JDBC and SQLJ packages to bind or drop for each isolation level and cursor holdability

-optprofile
      Specifies the optimization profile that is used for all static statements in the IBM Data Server Driver for JDBC and SQLJ packages when a value is not set in the CURRENT OPTIMIZATION PROFILE special register

Better diagnostics: In previous releases, the DB2Binder utility always returned return code 0. The DB2Binder utility now returns an integer value that indicates whether DB2Binder processing was successful. If the processing was unsuccessful, the returned value indicates the nature of the error.

XML data type encryption is supported

Data encryption is now supported for XML data under IBM Data Server Driver for JDBC and SQLJ type 4 connectivity when applications set the securityMechanism property to ENCRYPTED_USER_AND_DATA_SECURITY or ENCRYPTED_USER_PASSWORD_AND_DATA_SECURITY.

Progressive streaming is supported

The IBM Data Server Driver for JDBC and SQLJ supports progressive streaming, also known as dynamic data format, for LOBs for connections to DB2 Database for Linux, UNIX, and Windows and connections to DB2 for z/OS. Progressive streaming is enabled by default.

DatabaseMetaData methods long identifiers are supported

The following DatabaseMetaData methods now return the correct lengths for 128-byte identifiers:
* getMaxColumnNameLength
* getMaxCursorNameLength
* getMaxSchemaNameLength

New trace level is added

The new value TRACE_TRACEPOINTS for the Connection or DataSource property traceLevel determines whether internal trace points for the driver are traced. If you set the TRACE_TRACEPOINTS value, driver internal trace points are printed to the LogWriter that is enabled on a connection. As with any other traceLevel value, you can combine TRACE_TRACEPOINTS with any other values through OR logic.

ResultSet and DatabaseMetaData methods for dynamic scrollable cursors are added

The following JDBC 2.0 java.sql.ResultSet methods are now supported:

ResultSet.insertRow
> Inserts the contents of the insert row into a ResultSet object and into a table

ResultSet.moveToInsertRow
> Moves the cursor to the insert row for the ResultSet object

ResultSet.moveToCurrentRow
> Moves a cursor that is on an insert row to its previous cursor position in a ResultSet object

ResultSet.rowInserted
> Determines whether the current row in a ResultSet object was inserted.

The following JDBC 2.0 java.sql.DatabaseMetaData methods are now supported:

DatabaseMetaData.ownInsertsAreVisible
> Determines whether rows that are inserted into the underlying table by the ResultSet object are visible to the ResultSet

DatabaseMetaData.othersInsertsAreVisible
> Determines whether rows that are inserted into the underlying table by other applications or other ResultSet objects are visible to the specified ResultSet

DatabaseMetaData.insertsAreDetected
> Determines whether the ResultSet object can detect inserted rows

Arrays are supported

Version 9.5 supports arrays as input or output parameters for stored procedures. JDBC has corresponding support for retrieving or updating array parameters in client programs that call those stored procedures. Therefore, the IBM Data Server Driver for JDBC and SQLJ introduces the IBM Data Server Driver for JDBC and SQLJ–only interface com.ibm.db2.jcc.DB2Array.

You can retrieve the contents of an output array parameter in the following ways:
* As a java.lang.Object, using the DB2Array.getArray method
* As a java.sql.ResultSet, using the DB2Array.getResultSet method

You can update an input array parameter in the following ways:
* Using the PreparedStatement.setArray method
* Using the PreparedStatement.setObject method

Decimal floating-point data type is supported

Version 9.5 supports the DECFLOAT SQL type for storing decimal floating-point data. Applications that use the IBM Data Server Driver for JDBC and SQLJ can now store and retrieve decimal floating-point data in Version 9.5 databases.

DB2 Database for Linux, UNIX, and Windows uses the CURRENT DECFLOAT ROUNDING MODE special register to specify the default rounding mode that is used for decimal or decimal floating-point values. You can use the decimalRoundingMode Connection or DataSource property to set the special register if it is not already set.

Client reroute feature is enhanced
* Domain Name System (DNS) is supported as a repository of alternate server information. For client reroute during connections to DB2 Database for Linux, UNIX, and Windows servers, you can use DNS instead of the JNDI directory as a repository of alternate server information.

  You can specify multiple IP addresses in a DNS entry. For client reroute, you can specify two: one for the primary server and one for the secondary server. If JNDI is not configured, the IBM Data Server Driver for JDBC and SQLJ uses the DNS addresses to identify the servers for client reroute.
* Client reroute feature support is enhanced for connections to DB2 for z/OS.

Connection concentrator and Sysplex workload balancing support for DriverManager connections is added

In previous releases of the IBM Data Server Driver for JDBC and SQLJ, you could use the connection concentrator and Sysplex workload balancing feature only for connections that you established with the DataSource interface. You can now use this feature for connections that you establish with the DriverManager interface.

setXXXStream methods without an explicit length are supported

You can now specify -1 for the **length** parameter when you call the setAsciiStream, setBinaryStream, or setCharacterStream method. Doing this directs the IBM Data Server Driver for JDBC and SQLJ to input data until the input stream is exhausted.

Java application support is added for XML schema updates

The DB2Connection.updateDB2XmlSchema method updates one XML schema with the content of another XML schema. DB2Connection.updateDB2XmlSchema performs the same function as the new SYSPROC.XSR_UPDATE stored procedure.

PreparedStatement.setObject calls with Reader and InputStream objects are supported

In PreparedStatement.setObject, the data types of the input parameters for input to CLOB or XML columns can now be Reader. The data types of the input parameters for input to BLOB or XML columns can now be InputStream. The driver uses streaming to send the data to the database server if the database server supports streaming.

Properties are added

IBM Data Server Driver for JDBC and SQLJ includes the following new `Connection` and `DataSource` properties:

- Two new `Connection` and `DataSource` properties let you control optimization profile use at the connection level in JDBC or SQLJ programs:

  optimizationProfile
  Specifies an optimization profile to use for an SQLJ or a JDBC program.

  optimizationProfileToFlush
  Specifies an optimization profile to remove from the optimization profile cache.

- currentDegree property

  Sets the CURRENT DEGREE special register which specifies the degree of intrapartition parallelism for the execution of dynamic SQL statements.

- queryBlockSize property

  Specifies the size of query blocks that the database server uses to return data.

- retryWithAlternativeSecurityMechanism property

  Specifies whether the IBM DB2 Driver for JDBC and SQLJ retries a connection with an alternative security mechanism when the client-specified security mechanism is not supported by the data source. This property applies to type 4 connectivity to DB2 for Linux, UNIX, and Windows Version 8 and later only.

- reportLongTypes property

  Specifies whether DatabaseMetaData methods report LONG VARCHAR and LONG VARGRAPHIC data types on DB2 for z/OS as java.sql.Types.LONGVARCHAR or java.sql.Types.VARCHAR.

- timestampFormat property

  Specifies the format in which the result of the `ResultSet.getString` or `CallableStatement.getString` method against a TIMESTAMP column is returned.

javax_jcc.jar file is removed

The db2jcc_javax.jar file is no longer part of the IBM Data Server Driver for JDBC and SQLJ. Therefore, you no longer need to include the db2jcc_javax.jar file in the **CLASSPATH** environment variable for the IBM Data Server Driver for JDBC and SQLJ.

Optimistic locking is supported

Version 9.5 supports optimistic locking, which is a technique that an SQL database application can use to release a lock on a row after the application selects that row and before the application updates or deletes that row. The IBM Data Server Driver for JDBC and SQLJ now provides the following methods to support optimistic locking:

DB2Connection.prepareDB2OptimisticLockingQuery
> Creates a `PreparedStatement` object that can request optimistic locking information

DB2Statement.executeDB2OptimisticLockingQuery
> Executes a SELECT statement and optionally requests that optimistic locking columns are returned

DB2ResultSetMetaData.getDB2OptimisticLockingColumns
> Returns information about whether optimistic locking columns are available in a `ResultSet`

DB2ResultSet.getDB2RowChangeToken
> Returns the row change token for the current row if optimistic locking was requested

DB2ResultSet.getDB2RID
> Returns the RID column value for the current row if optimistic locking was requested

DB2ResultSet.getDB2RIDType
> Returns the underlying data type of the RID column

Timeout methods are supported

Support is added for the following methods for IBM Data Server Driver for JDBC and SQLJ type 4 connectivity against DB2 Database for Linux, UNIX, and Windows Version 9.1 or later databases:

javax.transaction.xa.XAResource.setTransactionTimeout
> Sets the current transaction timeout value for an `XAResource` instance

javax.transaction.xa.XAResource.getTransactionTimeout
> Obtains the current transaction timeout value for an `XAResource` instance

JDBC 3.0 methods are added

The following JDBC 3.0 methods are supported:

ResultSet.updateBlob
> Updates a value with an SQL BLOB data type in an updatable result set

ResultSet.updateClob
> Updates a value with an SQL CLOB data type in an updatable result set

Internal statement caching is supported

IBM Data Server Driver for JDBC and SQLJ type 4 connectivity supports an internal statement cache for `PooledConnection` objects. Multiple logical connections that are associated with the same physical `PooledConnection` can reuse statements from the cache, which can improve performance.

Enable and configure internal statement caching by using the maxStatements property on the `ConnectionPoolDataSource`. maxStatements specifies the maximum number of statements that can be kept open by the driver in an internal statement cache that is associated with a `PooledConnection`.

Error processing usability is improved

The following enhancements provide more useful diagnostic information:

- More meaningful error message text. All calls to java.sql.SQLException.getMessage and java.sql.SQLWarning.getMessage now return an SQLCODE and SQLSTATE. For errors that originate from the IBM Data Server Driver for JDBC and SQLJ, the message text also includes the version of the driver.
- Generation of warnings when the SQLSTATE is not null. In previous versions of the IBM Data Server Driver for JDBC and SQLJ, if the driver or the database server returned an SQLCODE of 0 and a non-null SQLSTATE, the driver did not accumulate a warning. Now, the driver accumulates warnings under those conditions to make the SQLSTATE information available to you.

Informix Dynamic Server database server is supported

You can now use the IBM Data Server Driver for JDBC and SQLJ to access Informix Dynamic Server (IDS) database servers.

Connection reuse support is enhanced

Support of connection reuse by a connection pooling module that is written by a user or software vendor is enhanced for connections to DB2 for Linux, UNIX, and Windows.

Enhancements for Version 9.5 Fix Pack 1

Starting in Version 9.5 Fix Pack 1, the following enhancements are included:

- Seamless failover is added to client reroute operation.

  During client reroute, if a connection is in a clean state, you can use the enableSeamlessFailover property to suppress the SQLException with error code -4498 that the IBM Data Server Driver for JDBC and SQLJ issues to indicate that a failed connection was re-established.

- Client affinities are added to cascaded failover support.

  For cascaded failover, you can use the enableClientAffinitiesList property to control the order in which primary and alternate server reconnections are attempted after a connection failure.

- Statement.setMaxRows performance is enhanced.

  For connections to DB2 for z/OS servers, the Statement.setMaxRows method has been modified to provide better performance.

- IDS database names can be longer than 18 bytes.

  For connections to IDS V11.11 and later, database names can be up to 128 bytes.

- IDS ISAM error reporting is enabled.

  For connections to IDS V11.10 and later, ISAM errors are reported as SQLException objects, so SQLException methods can be used to obtain the error code and the message description. In addition, SQLException.printStackTrace calls display information about the cause of the ISAM errors.

- More functions are supported for connections to IDS.

For connections to IDS 11.50 and later, the following functions are supported:
- Progressive streaming
- Multi-row insert operations
- SSL support
- Setting and Retrieving of client info properties

- Progressive streaming behavior can be changed after a connection is established.

  For connections to DB2 for z/OS or DB2 Database for Linux, UNIX, and Windows servers, the DB2Connection.setDBProgressiveStreaming method can be used to change progressive streaming behavior after a connection to a data source is established. The DB2Connection.getDBProgressiveStreaming method can be used to determine the current progressive streaming behavior.

- Global trace settings can be changed without shutting down the driver.

  You can set the db2.jcc.tracePolling global configuration property to indicate that when the following trace settings in the IBM Data Server Driver for JDBC and SQLJ global configuration file are changed while a driver instance is up, the driver modifies the trace behavior:
  - db2.jcc.override.traceLevel
  - db2.jcc.override.traceFile
  - db2.jcc.override.traceDirectory
  - db2.jcc.override.traceFileAppend

- ResultSet.next behavior for DB2 connections can be more compatible with ResultSet.next behavior for connections with other database managers.

  The allowNextOnExhaustedResultSet property can be set so that ResultSet.next behavior for a connection to DB2 for z/OS or DB2 Database for Linux, UNIX, and Windows is the same as ResultSet.next behavior for applications that are connected to an Oracle or MySQL data source. When the allowNextOnExhaustedResultSet property is set to DB2BaseDataSource.YES (1), and a forward-only cursor is positioned after the last row of a result set, a call to ResultSet.next returns `false`, instead of throwing an SQLException.

JDBC 4.0 support has been added

IBM Data Server Driver for JDBC and SQLJ Version 4.0 includes a number of JDBC 4.0 capabilities. To use these capabilities, you need an SDK for Java, Version 6.

Driver name has been changed

The new JDBC and SQLJ driver name is the IBM Data Server Driver for JDBC and SQLJ. The java.sql.DatabaseMetaData.getDriverName method returns this name instead of `IBM DB2 JDBC Universal Driver Architecture`.

JDBC 4.0 data type support is added

JDBC and SQLJ support is provided for the following JDBC 4.0 interfaces for updating and retrieving data in ROWID or XML columns:
- RowId. IBM Data Server Driver for JDBC and SQLJ–only type `com.ibm.db2.jcc.DB2Types.ROWID` and class com.ibm.db2.jcc.DB2RowId are deprecated.
- SQLXML. IBM Data Server Driver for JDBC and SQLJ–only type `com.ibm.db2.jcc.DB2Types.DB2Xml` and class com.ibm.db2.jcc.DB2Xml are deprecated.

JDBC 4.0 methods are supported

The following JDBC 4.0 methods are supported:

java.sql.Array.free
> Closes an Array object and releases any resources that it holds.

java.sql.Blob.free
> Closes a Blob object and releases any resources that it holds.

java.sql.Blob.getBinaryStream
> Retrieves a value from a Blob object as a binary stream.

java.sql.CallableStatement.getRowId
> Retrieves the value of an SQL ROWID parameter as a RowId object.

java.sql.CallableStatement.getSQLXML
> Retrieves the value of an SQL XML parameter as an SQLXML object.

java.sql.Clob.free
> Closes a Clob object and releases any resources that it holds.

java.sql.Clob.getCharacterStream
> Retrieves a value from a Clob object as a binary stream. The new form of this method supports a specified length of up to 2 GB.

java.sql.Connection.createBlob
> Creates a Blob object.

java.sql.Connection.createClob
> Creates a Clob object.

java.sql.Connection.createSQLXML
> Creates an SQLXML object.

java.sql.Connection.getClientInfo
> Returns information about the client info properties that the IBM Data Server Driver for JDBC and SQLJ supports. Those properties are as follows:
> - ApplicationName
> - ClientAccountingInformation
> - ClientHostname
> - ClientUser
>
> Connection.getClientInfo performs the same function as the following IBM Data Server Driver for JDBC and SQLJ–only methods, which are deprecated:
> - DB2Connection.getDB2ClientUser
> - DB2Connection.getDB2ClientWorkstation
> - DB2Connection.getDB2ClientApplicationInformation
> - DB2Connection.getDB2ClientAccountingInformation

java.sql.Connection.isValid
> Determines whether a connection is open. This method performs the same function as the IBM Data Server Driver for JDBC and SQLJ–only method DB2Connection.isDB2Alive, which is deprecated.

java.sql.Connection.setClientInfo
> Sets values for client info properties that the IBM Data Server Driver for JDBC and SQLJ supports. This method performs the same function as the following IBM Data Server Driver for JDBC and SQLJ-only methods, which are deprecated:
> - DB2Connection.setDB2ClientUser
> - DB2Connection.setDB2ClientWorkstation

- DB2Connection.setDB2ClientApplicationInformation
- DB2Connection.setDB2ClientAccountingInformation

java.sql.DatabaseMetaData.getClientInfoProperties
Retrieves a list of client info properties that the IBM Data Server Driver for JDBC and SQLJ supports.

java.sql.DatabaseMetaData.getColumns
Returns the additional column IS_AUTOINCREMENT in the result set.

java.sql.DatabaseMetaData.getFunctions
Retrieves a description of built-in functions and user-defined functions that are defined at the database server.

java.sql.DatabaseMetaData.getFunctionColumns
Retrieves information about parameters of specified built-in functions or user-defined functions that are defined at the database server.

java.sql.DatabaseMetaData.getProcedureColumns
Returns the additional column IS_AUTOINCREMENT in the result set.

java.sql.DatabaseMetaData.getProcedures
Returns the additional column SPECIFIC_NAME in the result set.

java.sql.DatabaseMetaData.getRowIdLifetime
Returns the length of time that a ROWID value is valid.

java.sql.DatabaseMetaData.getSchemas
Has a new form that allows specification of a catalog and schema pattern.

java.sql.PreparedStatement.setBlob
Sets an input parameter to a value and informs the driver that the value should be sent to the data source as a BLOB type.

java.sql.PreparedStatement.setAsciiStream
Sets an input parameter to a value and informs the driver that the value should be sent to the database server as a VARCHAR type. The new forms of this method support a specified length of up to 2 GB or no specified length.

java.sql.PreparedStatement.setBinaryStream
Sets an input parameter to a value and informs the driver that the value should be sent to the database server as a VARCHAR type. The new forms of this method support a specified length of up to 2 GB or no specified length.

java.sql.PreparedStatement.setCharacterStream
Sets an input parameter to a value and informs the driver that the value should be sent to the database server as a BLOB type. The new forms of this method support a specified length of up to 2 GB or no specified length.

java.sql.PreparedStatement.setClob
Sets an input parameter to a value and informs the driver that the value should be sent to the database server as a CLOB type.

java.sql.PreparedStatement.setObject
Sets an input parameter to a value, using the specified object. This existing method now supports RowId and SQLXML objects.

java.sql.PreparedStatement.setRowId
Sets an input parameter to a value and informs the driver that the value should be sent to the database server as a ROWID type.

java.sql.PreparedStatement.setSQLXML
>   Sets an input parameter to a value and informs the driver that the value should be sent to the database server as an XML type.

java.sql.ResultSet.getRowId
>   Retrieves a value into a RowId object from a ROWID column of a result set.

java.sql.ResultSet.getSQLXML
>   Retrieves a value into an SQLXML object from an XML column of a result set.

java.sql.ResultSet.updateAsciiStream
>   Updates a character column of an updatable result set. The new forms of this method support a specified length of up to 2 GB or no specified length.

java.sql.ResultSet.updateBinaryStream
>   Updates a binary column of an updatable result set. The new forms of this method support a specified length of up to 2 GB or no specified length.

java.sql.ResultSet.updateCharacterStream
>   Updates a character column of an updatable result set. The new forms of this method support a specified length of up to 2 GB or no specified length.

java.sql.ResultSet.updateBlob
>   Updates a value with an SQL BLOB data type in an updatable result set. The new forms of this method support a specified length of up to 2 GB or no specified length.

java.sql.ResultSet.updateClob
>   Updates a value with an SQL CLOB data type in an updatable result set. The new forms of this method support a specified length of up to 2 GB or no specified length.

java.sql.ResultSet.updateRowId
>   Updates a value with an SQL ROWID data type in an updatable result set.

java.sql.ResultSet.updateSQLXML
>   Updates a value with an SQL XML data type in an updatable result set.

java.sql.Statement.isClosed
>   Determines whether a `Statement` object is closed.

java.sql.Statement.setPoolable
>   Specifies whether a `Statement` object is pooled.

java.sql.Statement.isPoolable
>   Determines whether a `Statement` object can be pooled.

java.sql.SQLXML.free
>   Closes an SQLXML object and releases any resources that it holds.

java.sql.SQLXML.getBinaryStream
>   Retrieves a value from an SQLXML object as a binary stream.

java.sql.SQLXML.getCharacterStream
>   Retrieves a value from an SQLXML object as a character stream.

java.sql.SQLXML.getString
>   Retrieves a value from an SQLXML object as a string.

java.sql.SQLXML.getSource
> Returns a `javax.xml.transform.Source` object for reading the XML value in an SQLXML object.

java.sql.SQLXML.setBinaryStream
> Initializes an SQLXML object with a binary stream value.

java.sql.SQLXML.setCharacterStream
> Initializes an SQLXML object with a character stream value.

java.sql.SQLXML.setResult
> Returns a `javax.xml.transform.Result` object that initializes an SQLXML object.

java.sql.SQLXML.setString
> Initializes an SQLXML object with a string value.

javax.sql.PooledConnection.addStatementEventListener
> Registers a `StatementEventListener` object with a `PooledConnection` object.

javax.sql.PooledConnection.removeStatementEventListener
> Removes a `StatementEventListener` object from a `PooledConnection` object.

JDBC 4.0 exception classes are supported

The following JDBC 4.0 exception classes are supported:
- SQLNonTransientException and its subclasses:
    - SQLDataException
    - SQLFeatureNotSupportedException
    - SQLIntegrityConstraintViolationException
    - SQLInvalidAuthorizationException
    - SQLNonTransientConnectionException
    - SQLSyntaxErrorException
- SQLTransientException and its subclasses:
    - SQLTimeoutException
    - SQLTransactionRollbackException
    - SQLTransientConnectionException
- SQLRecoverableException
- SQLClientInfoException

Wrapper interface support is added

The `Wrapper` interface lets you access an instance of a resource that is wrapped. The following IBM Data Server Driver for JDBC and SQLJ–only classes implement the `Wrapper` interface:
- DB2Connection
- DB2BaseDataSource
- DB2SimpleDataSource
- DB2Statement
- DB2ResultSet
- DB2DatabaseMetaData

DB2Driver class is automatically loaded

Previously, when you used the `DriverManager` interface to establish a connection to a data source, you needed to call the Class.forName method to explicitly load the DB2Driver class. With JDBC 4.0 support, this step is no longer necessary.

Remote trace controller has been added

The remote trace controller lets you perform operations such as these for multiple driver instances:

* Start, stop, or resume a trace
* Change the output trace file or directory location
* Change the trace level

The remote trace controller uses the Java Management Extensions (JMX) architecture, which is part of the SDK for Java, Version 6, or later.

Enhancements for Version 9.5 Fix Pack 1

Starting in Version 9.5 Fix Pack 1, the following enhancements are included:

* SQLSTATEs are added for unsupported features and for timeouts.
    - For a feature that is not supported by a client, a java.sql.SQLFeatureNotSupportedException is thrown, with error code -4450 and SQLSTATE 0A504.
    - For a timeout, a java.sql.SQLTimeoutException is thrown, with error code -4210, -4211, or -4213, and SQLSTATE 57033.

IBM Data Server Provider for .NET supports IBM Informix Dynamic Server, IBM UniData, and IBM UniVerse

The IBM Data Server Provider for .NET now supports IBM Informix Dynamic Server, IBM UniData, and IBM UniVerse data servers along with DB2 data servers. This enables you to use the same data server provider for any combination of DB2, Informix, UniData, or UniVerse products in the same .NET application.

The Data Server Provider for .NET allows your .NET applications to access the following database management systems:

* DB2 Database for Linux, UNIX, and Windows, Version 9 (or later)
* DB2 Universal Database Version 8 for Windows, UNIX, and Linux-based computers
* DB2 Universal Database Version 7.3 (or later) for VSE & VM, through DB2 Connect
* DB2 Universal Database Version 6 (or later) for OS/390 and z/OS, through DB2 Connect
* DB2 Universal Database for iSeries (using OS/400 V5R1, or later), through DB2 Connect
* IBM Informix Dynamic Server, Version 11.10 or later
* IBM UniData 7.1.11 or later
* IBM UniVerse 10.2 or later

Data type compatibility varies depending on the data server used. For example, Informix Dynamic Server, UniData, and UniVerse data servers do not support the XML data type.

Related concepts

"IBM Data Server Provider for .NET" in Getting Started with Database Application Development

"New DB2 client product simplifies deployment (Windows)" on page 25

New DB2 sample programs have been added

You can use DB2 sample programs as templates to create your own application programs and to learn about DB2 product functionality.

The samples are bundled with all of the server editions of the DB2 Database for Linux, UNIX, and Windows and with the IBM Data Server Client. You can find the samples in the following locations:

- On Windows operating systems: *%DB2PATH%*\sqllib\samples (where *%DB2PATH%* is the directory where the DB2 product is installed, as set by the **DB2PATH** environment variable)
- On UNIX operating systems: *$HOME*/sqllib/samples (where *$HOME* is the home directory of the instance owner, as set by the **$HOME** environment variable)

For Version 9.5, there are many new sample programs available that demonstrate key features and enhancements:

Table 6. New Version 9.5 sample programs

Category	Feature or enhancement	Sample description
Performance	Faster data redistribution (available with Fix Pack 1)	Using the various options in the REDISTRIBUTE utility available in Fix Pack 1
	Reduced storage for automatic storage table spaces	Freeing up and reusing unused storage at the end of a table space
	Row compression enhancements	Using the Automatic Dictionary Creation enhancement
	Deferred index cleanup for rollout deletions	Changing the MDC rollout type from immediate index cleanup to deferred index cleanup
	Optimistic locking support	Using optimistic locking in a database
Manageability	Single-system view for database configuration	Updating database configuration parameters across multiple partitions in a partitioned database environment
	BACKUP DATABASE enhancements for partitioned database environments	Running the BACKUP DATABASE command to back up all partitions of a multi-partition database at once
Security	Audit facility enhancements	Creating, altering, and removing audit policies and archiving and viewing audit data
	Database role support	Creating roles, transferring object ownership using roles, using roles instead of groups, granting and revoking privileges through roles using role hierarchy
	Trusted context support	Creating a trusted context, identifying and using a trusted connection for switching user IDs, and acquiring trusted context–specific privileges

Table 6. New Version 9.5 sample programs (continued)

Category	Feature or enhancement	Sample description
Application development	Global variable support	Creating and deleting global variables and using global variables in triggers and stored procedures
	Array support	Declaring an array data type, passing an array to a stored procedure, using different functions to manipulate the array, and creating a table from the array and vice versa
	PHP support	Using PHP with various DB2 features such as database level authorities, DDL and DML statements, data types, and XML. Samples include support for the IBM_DB2 driver and the PDO driver.
	.NET support	Using DB2 XML features with .NET
XML	Trigger processing improvements	Using the trigger processing capability for enforced automatic validation of incoming XML documents
	Check constraints on XML columns	Creating tables with check constraints on an XML column using IS VALIDATED and IS NOT VALIDATED predicates and specifying one or more schemas using the ACCORDING TO XMLSCHEMA clause
	User-friendly publishing functions	Using the XMLROW and XMLGROUP functions for mapping relational data to XML
	XSLT support	Using the XSLTRANSFORM function to convert XML documents residing in the database into HTML, plain text, or other forms of XML using style sheets
	Support for document updates using XQuery	Using the XQuery transform expression to insert, delete, update, replace, and rename one or more XML documents or fragments
	Compatible XML schema evolution	Updating a registered XML schema, ensuring that the original and the new schemas are compatible
	Annotated XML schema decomposition enhancements	Specifying the order of insertion to be used during XML document shredding and registering recursive schemas and enabling them for decomposition
	Parameter passing to SQLQuery	Passing parameters from XQuery to the SQL fullselect specified in the db2-fn:sqlquery function
	Load utility support for XML	Loading XML documents into DB2 tables using the various LOAD command options

CLIENT APPLNAME special register is automatically set by the CLP

In Version 9.5, when you run a CLP script file using the db2 **-tvf** *filename* command, it sets the special register CLIENT APPLNAME to CLP *filename*. This is useful for monitoring which batch job is currently running and differentiating the CLP workloads.

You can obtain the value of the running application by extracting the value in the CLIENT_APPLNAME special register or by using the GET SNAPSHOT FOR ALL APPLICATIONS command.

Related reference

″sqleseti API - Set client information″ in Administrative API Reference

DB2 Developer Workbench has been renamed and enhanced

DB2 Developer Workbench has been renamed in Version 9.5 and is referred to now as IBM Data Studio.

IBM Data Studio maintenance is applied using the IBM Installation Manager updater facility. You can download the latest IBM Data Studio fix packs at http://www.ibm.com/software/data/studio/support.html and use the IBM Data Studio information center at http://publib.boulder.ibm.com/infocenter/dstudio/v1r1m0.

IBM Data Studio includes the following key features:

Database connection

- The IBM Data Server Driver for JDBC and SQLJ is included with the product. This driver can be used to connect to DB2 or Informix Dynamic Server 11 database servers.
- You can use the new connection profile feature to more easily share database connection information between IBM Data Studio users.
- You can connect to DB2 Database for Linux, UNIX, and Windows or DB2 for z/OS databases using Kerberos authentication.
- You can set database connection preferences on a new page in the Preferences window. Connection preferences include connection timeout, reconnect options, and user ID and password persistence.
- You can connect to DB2 servers using an LDAP (Lightweight Directory Access Protocol) infrastructure.
- You can generate trace files for JDBC connections. This feature is supported only for connections that use the IBM Data Server Driver for JDBC and SQLJ.

Data application development

- The SQL editor is designed to replace the XQuery builder, which is no longer part of the product. With the SQL editor, you can develop queries that work with both relational and XML data. You can use it to create and test SQL, SQL/XML, and XQuery expressions; editor features such as content assist, syntax highlighting, and query parsing and validation are provided. In addition to the editor enhancements, there is a new preferences page that allows you to create and use query templates.
- Creating and testing Java stored procedures has been improved. You can click a link to the Java source from the routine editor so that you can find and edit the Java source code more easily. Also, when you are deploying nested stored procedures, you can now see and select stored procedures with nested dependencies for deployment.
- Developing XML applications has been improved. XML data results are now easier to manipulate and the SQL editors for XML, XML schemas, and XSLT stylesheets improve developer productivity.

Database administration

There are several new tasks that are available from the Database Explorer that are typically performed by database administrators:

- Data object management. You can use the Data Object editor to create and alter many of the DB2 and Informix Dynamic Server database objects.
- Privilege management. You can use the Data Object editor to grant, revoke, and modify privileges related to data objects or authorization IDs.
- Visualization of data values and relationships. To visualize data object relationships, you can create an overview diagram from the Database Explorer and save these diagrams or print them as image files. Also, to visualize data value distributions, you can create a graphical value distribution view from the Database Explorer. These features can help you with query management and analysis (Visual Explain) tasks.
- Statistics support. You can view and update statistics for data objects to help you improve application performance. You can also use the DDL generation feature in the product to clone or migrate statistics from one database to another.

Informix Dynamic Server

There is new support for Informix Dynamic Server (IDS) in this release. After you connect to an IDS database in the Database Explorer, you can perform most database administration tasks, and you can use the connection information to create a data development project that targets IDS. You can develop and store SQL statements in the data development project; however, the wizards and editors that are available for DB2 routines are not yet available for IDS. To create and deploy routines for IDS, you can type the CREATE syntax and run it in the SQL editor. You can also run routines from the Database Explorer.

You can use the Informix JDBC driver or the IBM Data Server Driver for JDBC and SQLJ to connect to Informix databases.

Web services

You can now use the workbench to create Web services that expose database operations (SQL SELECT and DML statements, XQuery expressions, or calls to stored procedures) to client applications. You can create Web services in a data development project, and you can easily add stored procedures and SQL statements by dragging and dropping them to a Web service or by using a wizard. You can also use the workbench to deploy Web services to or undeploy them from a Web server.

You can also use the workbench to migrate existing Web services Object Runtime Framework (WORF) applications.

Installation

The IBM Installation Manager is now used to install, update, and manage IBM Data Studio from any DB2 server products on the following platforms: Linux on x86; 32-bit and 64-bit Windows platforms. You can also use Installation Manager for silent installation of the product. In addition, you can extend Eclipse 3.2 environments with IBM Data Studio functionality.

Related concepts

"Data server tools have been added" on page 27

New scalar functions simplify application porting

Version 9.5 includes new scalar functions with the same names as scalar functions used by other database vendors. When you port existing applications to Version 9.5, you can continue to use the names of the functions used by other vendors without changing your code.

The following scalar functions are provided:
- NVL (synonym for the existing COALESCE and VALUE functions)
- LEAST or MIN (synonyms for each other)
- GREATEST or MAX (synonyms for each other)
- DECODE (similar to the existing CASE expression)

  Related reference

  "DECODE scalar function" in SQL Reference, Volume 1

  "GREATEST scalar function" in SQL Reference, Volume 1

  "LEAST scalar function" in SQL Reference, Volume 1

  "MAX scalar function" in SQL Reference, Volume 1

  "MIN scalar function" in SQL Reference, Volume 1

  "NVL scalar function" in SQL Reference, Volume 1

New bitwise scalar functions added

You can simplify your application code by using new functions and operators to perform bitwise manipulation of DB2 data.

The following new bit manipulation scalar functions are available:
- BITAND
- BITOR
- BITXOR
- BITNOT
- BITANDNOT

These bitwise functions operate on the "two's complement" representation of the integer value of the input arguments and return the result as a corresponding base 10 integer value in a data type based on the data type of the input arguments. The largest supported type supports 113 bits.

  Related reference

  "BITAND, BITANDNOT, BITOR, BITXOR, and BITNOT scalar functions" in SQL Reference, Volume 1

FP1: IBM Data Server Provider for .NET supports trusted context

Starting in Version 9.5 Fix Pack 1, .NET applications support trusted context using connection string keywords.

Trusted contexts provide a way of building much faster and more secure three-tier applications. The user's identity is always preserved for auditing and security purposes. When you need secure connections, trusted contexts improve performance because you do not need to get new connections. For more information, see "Trusted contexts enhance security" on page 47.

The following properties in the DB2ConnectionStringBuilder class aid in the construction of trusted context connection strings:

- TrustedContextSystemUserID
- TrustedContextSystemPassword

Related concepts

″Creating a trusted connection through IBM Data Server Provider for .NET″ in Developing ADO.NET and OLE DB Applications

Chapter 10. High availability, backup, logging, and recovery enhancements

Version 9.5 includes enhancements that ensure that your data remains available for your users.

Review this section to find out more details about high availability, backup, logging, and recovery improvements in Version 9.5.

New system stored procedures simplify automated maintenance policy configuration

You can use four new system stored procedures to collect automated maintenance policy information and to configure an automated maintenance policy.

You can use the new SYSPROC.AUTOMAINT_SET_POLICY and SYSPROC.AUTOMAINT_SET_POLICYFILE system stored procedures to create a policy for maintenance windows, automatic backups, automatic table and index reorganizations, and automatic table RUNSTATS operations. AUTOMAINT_SET_POLICY takes an XML LOB as input, and AUTOMAINT_SET_POLICYFILE takes an XML file as input. There are sample XML input files located in the SQLLIB/samples/automaintcfg directory that you can modify to suit your requirements.

You can also use two new system stored procedures called SYSPROC.AUTOMAINT_GET_POLICY and SYSPROC.AUTOMAINT_GET_POLICYFILE to collect automated maintenance policy information about maintenance windows, automatic backups, automatic table and index reorganizations, and automatic table RUNSTATS operations. AUTOMAINT_GET_POLICY returns the policy information in a BLOB in XML format. AUTOMAINT_GET_POLICYFILE returns the policy information in an XML file.

You can pass the output from AUTOMAINT_GET_POLICY as input to AUTOMAINT_SET_POLICY, and you can pass the output from AUTOMAINT_GET_POLICYFILE as input to AUTOMAINT_SET_POLICYFILE.

Related tasks

"Configuring an automated maintenance policy using SYSPROC.AUTOMAINT_SET_POLICY or SYSPROC.AUTOMAINT_SET_POLICYFILE" in Data Recovery and High Availability Guide and Reference

"Collecting automated maintenance policy information using SYSPROC.AUTOMAINT_GET_POLICY or SYSPROC.AUTOMAINT_GET_POLICYFILE" in Data Recovery and High Availability Guide and Reference

Related reference

"AUTOMAINT_GET_POLICY procedure - retrieve automatic maintenance policy" in Administrative Routines and Views

"AUTOMAINT_GET_POLICYFILE procedure - retrieve automatic maintenance policy" in Administrative Routines and Views

"AUTOMAINT_SET_POLICYFILE procedure - configure automatic maintenance policy" in Administrative Routines and Views

"AUTOMAINT_SET_POLICY procedure - configure automatic maintenance policy" in Administrative Routines and Views

New DB2 Advanced Copy Services (ACS) API enables integration with storage hardware

The new DB2 Advanced Copy Services (ACS) application programming interface (API) enables you to perform snapshot backup operations with your storage hardware.

In a traditional backup or restore operation, the database manager copies data to or from disk or a storage device using operating system calls. Being able to use the storage device to perform the data copying makes the backup and restore operations much faster. A backup operation that uses DB2 ACS is called a snapshot backup.

The DB2 ACS application programming interface (API) defines a set of functions that the database manager uses to communicate with storage hardware to perform snapshot backup operations.

Integrated into IBM Data Server is a DB2 ACS API driver for the following storage hardware:
* IBM TotalStorage SAN Volume Controller
* IBM Enterprise Storage Server Model 800
* IBM System Storage DS6000
* IBM System Storage DS8000
* IBM N Series
* NetApp V-series

To perform snapshot backup operations with any other storage hardware, all you need is a DB2 ACS API driver for that storage hardware.

Related concepts

"DB2 Advanced Copy Services (ACS) API" in Data Recovery and High Availability Guide and Reference

Related tasks

"Enabling DB2 Advanced Copy Services (ACS)" in Data Recovery and High Availability Guide and Reference

Recovery object management is simplified through automated recovery object removal

You can now configure the DB2 database manager to automatically delete backup images, load copy images, and old log files that are no longer needed for recovery.

Backup images, load copy images, and old log files consume a large amount of storage space. It is necessary to regularly delete these recovery objects to save storage space. The DB2 database manager automatically prunes the database history file of entries that exceed the number specified in the **num_db_backups** configuration parameter and are older than the date specified by the **rec_his_retentn** configuration parameter. If you set the new configuration

parameter **auto_del_rec_obj** to ON, the database manager will also delete backup images, load copy images, and log files associated with any history file entries that it automatically prunes. Once **auto_del_rec_obj** is enabled, the system will only perform this maintenance when both the **num_db_backups** and **rec_his_retentn** values are exceeded.

You can also use the PRUNE HISTORY command to prune the history file manually. If you use the AND DELETE clause with the PRUNE HISTORY command or if you set the **iOption** parameter of the db2Prune API function to DB2PRUNE_OPTION_DELETE, the database manager will delete log files associated with any history file entries that you prune. If you set **auto_del_rec_obj** to ON, the database manager will also delete backup images, load copy images, and log files associated with any history file entries that you prune.

Related tasks

"Automating database recovery object management" in Data Recovery and High Availability Guide and Reference

"Protecting recovery objects from being deleted" in Data Recovery and High Availability Guide and Reference

"Managing recovery objects" in Data Recovery and High Availability Guide and Reference

Related reference

"db2Prune API - Delete the history file entries or log files from the active log path" in Administrative API Reference

"PRUNE HISTORY/LOGFILE command" in Command Reference

"PRUNE HISTORY/LOGFILE command using the ADMIN_CMD procedure" in Administrative Routines and Views

Cluster configuration and administration is simplified with new DB2 high availability instance configuration utility

You can use the new DB2 high availability instance configuration utility (db2haicu) to configure and administer your database solutions in clustered environments. db2haicu simplifies cluster configuration and administration because you can use db2haicu to perform database-related cluster configuration and administration instead of interacting directly with the cluster manager.

db2haicu has an interactive command-line interface. db2haicu collects configuration information about your cluster, database instance, and machines by asking a series of questions and by directly investigating your system. Once the configuration information collected, db2haicu then creates an abstract model of your clustered environment called the cluster domain.

Once db2haicu collects this information and creates a cluster domain, you can use db2haicu to perform cluster administration tasks such as:

- adding new databases to the cluster domain
- identifying DB2 High Availability Disaster Recovery (HADR) primary and standby database pairs
- adding new machines to the cluster domain
- moving database instances off a machine in the cluster to perform maintenance on the machine
- specifying failover policies

db2haicu acts as an interface between you and your cluster manager. For example, to add a database to the cluster domain means to make the cluster manager aware of a new database you have created on a machine in the cluster.

Your cluster manager must support the DB2 cluster manager API, providing a DB2 cluster manager API driver in order for db2haicu to successfully work with your cluster manager. IBM Tivoli System Automation for Multiplatforms (SA MP) supports the DB2 cluster manager API, and the Tivoli SA MP Base Component is integrated into IBM Data Server install on Linux and AIX as part of the DB2 High Availability Feature. You can use db2haicu to configure your clustered environment if you are using Tivoli SA MP as your cluster manager.

Related tasks

"Configuring a Clustered environment for high availability" in Data Recovery and High Availability Guide and Reference

"Configuring a clustered environment using DB2 High Availability Instance Configuration Utility (db2haicu)" in Data Recovery and High Availability Guide and Reference

Dual log control files makes database recovery more resilient

In Version 9.1, the database manager maintained one log control file: SQLOGCTL.LFH. In Version 9.5, the database manager maintains two copies of the log control file: SQLOGCTL.LFH.1 and SQLOGCTL.LFH.2. Having two copies of the log control file reduces the risk of losing data in the event of a failure.

When a database restarts after a failure, the database manager applies transaction information stored in log files to return the database to a consistent state. The database manager uses a log control file to determine which entries in the log files need to be applied.

If the log file control is damaged, it might not be possible for the database manager to return the database to a consistent state. However, having two copies of the log control file can make database recovery more resilient because if one copy of the log control file is damaged, the database manager can use the other copy during a restart.

Related concepts

"Database logging" in Data Recovery and High Availability Guide and Reference

"Log control files" in Data Recovery and High Availability Guide and Reference

HADR peer window reduces the risk of data loss during cascading or multiple failures

You can use the new **hadr_peer_window** database configuration parameter to cause a DB2 High Availability Disaster Recovery (HADR) primary and standby databases pair to behave as though it is in the peer state if the primary database loses connection with the standby database.

When a HADR primary and standby database pair is in peer state, transactions are not considered committed until the primary database receives confirmation from the standby database that database logs have been written to memory or the local log path (depending on the synchronization mode) for the standby database. This helps ensure data consistency: if there is a failure on the primary database, then all

of the transaction information that was in the database logs on the primary database is in the database logs on the standby database.

When the primary and standby databases are in peer state, then if the primary database loses the connection to the standby database, transactions cannot be committed because the primary database cannot receive confirmation from the standby database for any transactions. In previous versions of IBM Data Server, when the primary database lost connection with the standby database, the primary database moved into disconnected state by itself and continued to be available to process database application requests independent of the standby database. If the primary database failed while it was processing transactions independent of the standby database, transaction information on the primary database could be lost.

In Version 9.5, if you set the **hadr_peer_window** database configuration parameter to a non-zero value, then the primary database will move from peer state to a new disconnected peer state if it loses connection with the standby database. When the primary database is in disconnected peer state, it continues to behave as if it was in peer state: waiting for confirmation from the standby database before committing transactions. The period of time when the primary database remains in disconnected peer state is called the peer window. Although the availability of the primary database is reduced during the peer window, no committed transaction would be lost if the primary database failed during the peer window as in the case of multiple or cascading failures.

> **Related concepts**
>
> "DB2 High Availability Disaster Recovery (HADR) standby database states" in Data Recovery and High Availability Guide and Reference
>
> **Related reference**
>
> "db2HADRTakeover API - Instruct a database to take over as the high availability disaster recovery (HADR) primary database" in Administrative API Reference
>
> "TAKEOVER HADR command" in Command Reference
>
> "hadr_peer_window - HADR peer window configuration parameter" in Data Servers, Databases, and Database Objects Guide

Multiple database partitions can be backed up and restored at once with single system view backup

You can now back up and restore multiple database partitions at once using the new single system view (SSV) backup.

Prior to Version 9.5, you had to back up partitioned databases one database partition at a time. Backing up multiple database partitions one database partition at a time is prone to error and time consuming. If you back up a partitioned database one database partition at a time, you cannot include the log files required to restore and recover in the backup images.

Restoring multiple database partitions that were backed up individually is complicated because the backup timestamp for each database partition is slightly different. Because the backup timestamp is different for each of the database partitions, identifying all database partitions that belong to the same backup is difficult, and determining the minimum recovery time for the backup that contains all those database partitions is difficult.

Using the db2_all command simplified backing up partitioned databases somewhat, but there were still restrictions with the backup and restore operations that made the tasks difficult.

In Version 9.5, when you perform a backup operation from the catalog node of a partitioned database, you can specify which partitions to include in the backup, or specify that all the database partitions should be included. The specified partitions will be backed up simultaneously, and the backup timestamp associated with all specified database partitions will be the same. Also, you can include database logs with a SSV backup; including logs in backup images is the default behavior for snapshot backup operations. Finally, when you restore from a SSV backup image, you can specify to roll forward to *end of logs*, which is the minimum recovery time calculated by the database manager.

Related concepts

"Backup overview" in Data Recovery and High Availability Guide and Reference

Related tasks

"Using backup" in Data Recovery and High Availability Guide and Reference

"Backing up partitioned databases" in Data Recovery and High Availability Guide and Reference

Related reference

"BACKUP DATABASE command" in Command Reference

"db2Backup API - Back up a database or table space" in Administrative API Reference

"BACKUP DATABASE command using the ADMIN_CMD procedure" in Administrative Routines and Views

Rolling forward to minimum recovery time has been enabled

You can use the **TO END OF BACKUP** clause with the ROLLFORWARD command or the DB2ROLLFORWARD_END_OF_BACKUP flag with the db2Rollforward API to roll forward all partitions in a partitioned database to the minimum recovery time.

The minimum recovery time is the earliest point in time during a rollforward when a database is consistent (when the objects listed in the database catalogs match the objects that physically exist on disk). Manually determining the correct point in time to which to roll forward a database is difficult, particularly for a partitioned database. In Version 9.5, you can roll a database forward to the minimum recovery time, determined by the database manager, by using the **TO END OF BACKUP** parameter with the ROLLFORWARD DATABASE command, or the DB2ROLLFORWARD_END_OF_BACKUP option with the db2Rollforward API.

Related tasks

"Using rollforward" in Data Recovery and High Availability Guide and Reference

Related reference

"db2Rollforward API - Roll forward a database" in Administrative API Reference

"ROLLFORWARD DATABASE command" in Command Reference

Backing up and restoring data is faster with snapshot backups

When you run a snapshot backup or restore operation, your storage device performs the data copying part of the backup or restore. Being able to use the storage device to perform the data copying makes the backup and restore operations much faster.

In a traditional backup or restore operation, the database manager copies data to or from disk or a storage device using operating system calls. Being able to use the storage device to perform the data copying makes the backup and restore operations much faster. A backup operation that uses DB2 ACS is called a snapshot backup.

To perform a snapshot backup, you must have DB2 Advanced Copy Services (ACS) enabled, and you must have a DB2 ACS API driver for your storage hardware.

Integrated into IBM Data Server is a DB2 ACS API driver for the following storage hardware:
- IBM TotalStorage SAN Volume Controller
- IBM Enterprise Storage Server Model 800
- IBM System Storage DS6000
- IBM System Storage DS8000
- IBM N Series
- NetApp V-series

  Related tasks

  "Restoring from a snapshot backup image" in Data Recovery and High Availability Guide and Reference

  "Performing a snapshot backup" in Data Recovery and High Availability Guide and Reference

  Related reference

  "BACKUP DATABASE command" in Command Reference

  "RESTORE DATABASE command" in Data Movement Utilities Guide and Reference

  "db2acsutil - Manage DB2 snapshot backup objects command" in Command Reference

Cluster management software integration enabled

The new DB2 cluster manager application programming interface (API) enables you to use IBM Data Server cluster configuration tools such as the DB2 high availability instance configuration utility (db2haicu) to configure your clustered environment.

The DB2 cluster manager API defines a set of functions that the database manager uses to interface with your cluster manager to configure your clustered environment. Your cluster manager must support the DB2 cluster manager API, providing a DB2 cluster manager API driver in order for the database manager to successfully work with your cluster manager.

IBM Tivoli System Automation for Multiplatforms (SA MP) supports the DB2 cluster manager API, and the Tivoli SA MP Base Component is integrated with IBM Data Server install on Linux and AIX as part of the DB2 High Availability

Feature. You can use db2haicu to configure your clustered environment if you are using Tivoli SA MP as your cluster manager.

Related concepts

"DB2 cluster manager API" in Data Recovery and High Availability Guide and Reference

Related tasks

"Configuring a clustered environment using DB2 High Availability Instance Configuration Utility (db2haicu)" in Data Recovery and High Availability Guide and Reference

Related reference

"Supported cluster management software" in Data Recovery and High Availability Guide and Reference

Chapter 11. Installation, migration, and fix pack enhancements

Version 9.5 includes enhancements that make it faster to deploy products and easier to maintain them.

If you have a Version 9.1 or Version 8 copy installed and you want upgrade them to Version 9.5, you need to migrate to Version 9.5. DB2 Version 9.5 is a new release. You cannot apply a fix pack to upgrade from Version 9.1 to Version 9.5.

Review this section for more details about installation, migration, and fix pack enhancements for Version 9.5.

To learn about migration limitations, possible issues, and other details that you need to be aware of, see "Migration essentials for DB2 servers" in *Migration Guide* and "Migration essentials for clients" in *Migration Guide* .

Migrating your DB2 servers and DB2 clients to Version 9.5 might require that you migrate your database applications and routines. Review the "Migration essentials for database applications" in *Migration Guide* and "Migration essentials for routines" in *Migration Guide* topics to help you determine whether there is any migration impact.

IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component is integrated into the DB2 installation (Linux and AIX)

The IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component, version 2.2, is now bundled with the IBM Data Server on Linux and AIX operating systems.

You can now install, update, and uninstall the SA MP Base Component using the DB2 installer or Tivoli scripts that are also bundled with the IBM Data Server products.

IBM Tivoli System Automation for Multiplatforms provides cluster management such as system monitoring, failover, and autonomic repair and maintenance. For more information, see: Tivoli software information center.

Related concepts

"IBM Tivoli System Automation for Multiplatforms (Linux and AIX)" in Data Recovery and High Availability Guide and Reference

"Installing and upgrading the SA MP Base Component with the DB2 installer" in Data Recovery and High Availability Guide and Reference

Common fix pack simplifies server product updates

To update any or all of the DB2 server products in the same installation location, you no longer need to individually apply product-specific DB2 server fix packs. You can use a single DB2 server fix pack image to update them.

You can also use the server fix pack images to update the IBM Data Server Client on all platforms and the IBM Data Server Runtime Client on Linux and UNIX platforms.

Related tasks

Fix pack post-installation tasks are automated (Linux and UNIX)

Two previously manual steps during fix pack installation, the running of the db2iupdt and dasupdt commands, are now automated, which simplifies fix pack installation. In addition, binding occurs automatically at the first connection. As a result, once you start the database manager, the DB2 product is ready to use immediately after installation.

In previous releases, you had to update instances and bind manually.

Related tasks

"Applying fix packs" in Troubleshooting Guide

Non-root users can install and configure DB2 products (Linux and UNIX)

Before Version 9.5, you could install products, apply and roll back fix packs, configure instances, add features, or uninstall products only if you had root privileges. Now, if you are a non-root user, you can perform these tasks on Linux and UNIX platforms.

The DB2 installer automatically creates and configures a non-root instance during a non-root installation. As a non-root user, you can customize the configuration of the non-root instance during the installation. You can also use and maintain the installed DB2 product without root privileges.

The non-root installation of a DB2 product has one DB2 instance with most features enabled by default.

A non-root installation can be attractive for many groups, such as the following ones:

- Enterprises that have thousands of workstations and users who want to install a DB2 product without consuming a system administrator's time
- Application developers who are not typically system administrators but use DB2 products to develop applications
- Independent Software Vendors (ISVs) who develop software that does not require root authority yet embeds a DB2 product

Although non-root installations have most of the functionality of root installations, there are some differences and limitations. You can remove some of the limitations by having a root user run the db2rfe command.

Related concepts

"Differences between root installations and non-root installations" in Quick Beginnings for IBM Data Server Clients

"Limitations of non-root installations" in Quick Beginnings for IBM Data Server Clients

Related tasks

"Enabling root-based features in non-root installations with db2rfe" in Quick Beginnings for IBM Data Server Clients

New response file keywords have been added

A response file is an ASCII text file that contains setup and configuration information. Unlike installing with the DB2 Setup wizard, installing with a DB2 response file lets you install DB2 products or features without user interaction.

New response file keywords make it easier to install and deploy DB2 products.

You can use this keyword on AIX and Linux platforms to setup the IBM Tivoli System Automation for Multiplatforms (SA MP) Base Component:

- INSTALL_TSAMP

You can use these keywords to enable extended security on Windows platforms:

- DB2_ADMINGROUP_DOMAIN
- DB2_USERSGROUP_DOMAIN

Ready-to-use sample response files with default entries are included with the product. On the DB2 CD, the sample response files are located in db2/platform/samples (where platform refers to the hardware platform).

Related reference

"Response file keywords" in Quick Beginnings for DB2 Servers

Non-DB2 instance merge modules have been added

Before Version 9.5, you could add DB2 Runtime Client functionality to any product using the Windows Installer, using DB2 Runtime Client merge modules. In Version 9.5, the IBM Data Server Runtime Client merge modules are referred to as DB2 instance merge modules, and non-DB2 instance merge modules have been added.

Use the new non-DB2 instance merge modules to easily add IBM Data Server Driver for ODBC, CLI, and .NET functionality to any product that uses the Windows Installer. The IBM Data Server Driver for ODBC, CLI, and .NET is a new installable component that makes it much easier to provide access to DB2 servers from Windows-based applications. The IBM Data Server Driver for ODBC, CLI, and .NET is designed to be redistributed by independent software vendors (ISVs) and to be used for application distribution in mass deployment scenarios typical of large enterprises.

The main distinguishing features of the IBM Data Server Driver for ODBC, CLI, and .NET are as follows:

- It ships as a single executable, making it easy to redistribute and deploy.
- The Windows Installer Merge Module (.msm files) are available, which simplifies the integration of the IBM Data Server Driver for ODBC, CLI, and .NET code within a larger application.

Related concepts

"IBM data server client types" in Quick Beginnings for IBM Data Server Clients

Chapter 11, "Installation, migration, and fix pack enhancements," on page 101

Related reference

"Non-DB2 instance merge modules (Windows)" in Quick Beginnings for IBM Data Server Clients

Standalone instance type is more consistent across supported platforms (Linux and UNIX)

The instance type standalone has been added to Linux and UNIX operating systems to make instance types more consistent across supported DB2 platforms.

A standalone instance is an instance for a database server and local clients that allow only local connections. To create a standalone type, specify standalone for the **-s** parameter of the db2icrt command.

Related reference

"db2icrt - Create instance command" in Command Reference

Windows Vista deployment and usage is easier

Support for the Windows Vista operating system was introduced in Version 9.1 Fix Pack 2. It is now even easier to deploy and use DB2 and DB2 Connect products on the Windows Vista operating system.

Version 9.5 includes the following enhancements:

- Support has been added for IBM Data Studio and Query Patroller.
- A DB2 shortcut has been added to launch the DB2 command window with full administrator privileges. If you are a member of the local administrators' group, use this shortcut to launch DB2 commands and tools that require operating system local administrator privileges.

Related concepts

"Extended security requires users to belong to the DB2ADMNS or DB2USERS group (Windows Vista)" on page 144

Related reference

"Installation requirements for DB2 servers and IBM data server clients (Windows)" in Getting Started with Database Application Development

"Installation requirements for DB2 Connect Personal Edition (Windows)" in Quick Beginnings for DB2 Connect Personal Edition

"Installation requirements for DB2 Connect server products (Windows)" in Quick Beginnings for DB2 Connect Servers

FP1: Solaris x64 support has been added

You can install DB2 Version 9.5 Fix Pack 1 (and higher) products on the Solaris Operating Environment (x64).

The following products and features, however, are not supported on Solaris x64:

- IBM DB2 Personal Edition for Linux, UNIX, and Windows
- IBM DB2 Net Search Extender
- DB2 Query Patroller (QP)
- IBM DB2 Embedded Application Server
- The IBM Tivoli System Automation for Multiplatforms components of the High Availability Feature

The following functionality are also unavailable in DB2 database products on Solaris x64:

- Kerberos security plug-ins

- Application Response Measurement (ARM support)

  Related reference

  "Installation requirements for DB2 servers and IBM data server clients (Solaris Operating System)" in Getting Started with Database Application Development

  "DB2 Version 9 product and packaging information" in Quick Beginnings for DB2 Servers

Chapter 12. Federation enhancements

Application development, configuration and security have all been improved in IBM WebSphere Federation Server, Version 9.5.

The following federation enhancements are available in IBM WebSphere Federation Server, Version 9.5.

Application development has been enhanced for federation

The application development enhancements available in Version 9.5 include support for the XML data type, application savepoints, and WITH HOLD cursors.

XML data type support

With support for the remote XML data type, you can access and manipulate remote DB2 Database for Linux, UNIX, and Windows XML data through the federated server. XML support includes the following key features:
- A type mapping between the XML type on the federation server and a remote XML type
- Use of the SQL/XML and XQuery languages to provide flexible XML data manipulation

Application savepoints

Federated application savepoints offer you more control over transactions, reduce locking contention, and improve integration with application logic. You can logically divide a transaction into a single level or nested levels of savepoint units. Each savepoint can be individually released or rolled back as per application logic.

WebSphere Federation Server now supports application savepoints for insert, update, and delete operations against the DB2 Database for Linux, UNIX, and Windows data source.

WITH HOLD cursors

WebSphere Federation Server now allows cursors that you declare using the WITH HOLD attribute to remain open across multiple units of work for the DRDA® wrapper and the DB2 Database for Linux, UNIX, and Windows data source. This enhancement provides better control over processing result sets with cursors.

Previously, cursors were closed when a unit of work was completed by a commit or a rollback operation. Now, cursors at the federated server are synchronized with cursors at the data source, and the federated cursor is closed at the correct point in time.

Security has been enhanced for federation

New security enhancements for federation include trusted contexts, database roles, and a new C-based interface to external user mapping repositories.

Trusted contexts

A trusted context is a database security object that defines a trusted relationship between the federated server and an external entity, such as an application server or data source server. When an explicit connection request that matches the definition of a trusted context is made, the federated server establishes an inbound trusted connection from the application server to the federated server and attempts to establish an outbound trusted connection from the federated server to the data source server.

In a federated system, trusted contexts provide these benefits:
- End-to-end user identity. The identity of the user is propagated through the entire federated system.
- Accountability. The audit log identifies the transactions that a server performs for its own purposes and the transactions that each individual user performs. Therefore, individual users can be held accountable for specific transactions.
- Privilege control. You can grant a default role to all users of a trusted context. In addition, you can grant additional specific roles to specific users so that they have the additional privileges only while they work in that particular context.
- Security. The system authorization ID obtains only the privileges that the system needs, not the superset of all privileges that all users need. If the system authorization ID is compromised, data is less vulnerable than it would be if the authorization ID had all privileges.
- Administrative maintenance. The number of user mappings and groups is significantly reduced.
- Performance. After it is established, the trusted connection remains in place for the life of the connection. Each time that another user connects, the server does not close the physical connection and establish a new one. Instead, the server switches the current user ID of the connection to a different user ID. Depending on how the trusted context is defined, authentication might not be required.

Federated trusted contexts are supported for use with the DRDA wrapper and these data sources: DB2 Universal Database for Linux, UNIX and Windows and DB2 Universal Database for z/OS.

C-based interface to an external user mapping repository

By default, as the administrator, when you create user mappings, they are encrypted and stored in the global catalog on each federated server. In general, users require one or more mappings on every federated server that they use. Each time that a remote password for a data source changes, you must update the user mappings in one or more global catalogs.

To improve the security of user mapping information and reduce the maintenance of it, store user mapping information in an external repository, such as on an LDAP server, which can use additional security features such as SSL and strong encryption to secure the information. After you configure the federated servers to use the external repository, when a remote password changes, you need to update the user mapping only once.

The federated server uses a plug-in to provide the interface to the external repository. Previously, WebSphere Federation Server supported using only Java to build the plug-in. Now, C and C++ are supported. An interface library, a header file, and a sample C plug-in are provided.

Configuration has been enhanced for federation

The configuration enhancements for federation available in Version 9.5 include automatic collection of nickname statistics and an improved method for generating nickname column and index names.

Automatic refresh of nickname statistics

Automatic statistics collection is extended to keep nickname statistics up-to-date by running the nickname statistics (NNSTAT) stored procedure to refresh the statistics automatically. Current statistics enable the optimizer on the federated server to make informed choices that improve the performance of query plans.

For more information, see Automatic refresh of nickname statistics at http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/topic/com.ibm.swg.im.iis.fed.query.doc/topics/iiyfqnnsatm.html.

Improved nickname column and index name generation

The method used for generating nickname column and index names for relational nicknames has been enhanced so that the generated names more closely match the original names.

For more information, see Nickname column and index names at http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/topic/com.ibm.swg.im.iis.fed.query.doc/topics/iiyfqnnonam.html.

Chapter 13. Replication enhancements

Replication enhancements in Version 9.5 include a new CCD target type and support for the DECFLOAT decimal floating-point data type.

Review this section to find out more details about Version 9.5 replication enhancements.

New CCD target type avoids join of UOW and CD tables

You now have the option of replicating to a consistent-change-data (CCD) target table without joining the change-data (CD) table and the IBMSNAP_UOW table.

To use the new table type, specify a value of 9 in the TARGET_STRUCTURE column of the IBMSNAP_SUBS_MEMBR table.

Even though the type 9 CCD table includes the IBMSNAP_LOGMARKER column, the Apply program does not require a join of the CD table and IBMSNAP_UOW table to obtain the source commit information for this column. Instead, the Apply program generates same value in the IBMSNAP_LOGMARKER column for all of the rows in the same cycle.

The new CCD table type has the same structure as a type 3 CCD table. It contains four mandatory IBM columns in addition to the user columns:

IBMSNAP_COMMITSEQ
IBMSNAP_INTENTSEQ
IBMSNAP_OPERATION
IBMSNAP_LOGMARKER

You can register the new target table type as a source table for a three-tier replication configuration.

DECFLOAT data type is supported for replication

The new DECFLOAT decimal floating-point data type supports business applications that require exact decimal values, with precisions of 16 or 34 digits. You can replicate data that you define with the DECFLOAT data type on both DB2 Database for Linux, UNIX, and Windows and DB2 for z/OS.

You can use columns that you define with DECFLOAT as part of a replication key column.

SQL replication does not support mapping source columns of DECFLOAT (34) to target columns of DECFLOAT (16), because of data truncation.

Chapter 14. National language enhancements

National language enhancements provide more options for ordering data, new Unicode string literal and character-based processing for new scalar functions.

Review the following sections for national language enhancements introduced in Version 9.5.

Language-aware collation provides more options for ordering data

When you create a Unicode database, you can now specify a language-aware collation. A language-aware collation allows you to choose data ordering that is more customary for your language and territory.

Language-aware collations are based on the SYSTEM collation for a non-Unicode database. Each language-aware collation orders Unicode data as if it had been code-page converted to the non-Unicode code page and then had the corresponding SYSTEM collation applied.

If you convert a non-Unicode database to Unicode, using language-aware collation ensures that you do not see a difference in the ordering of data.

You can also use non-Unicode SYSTEM collations with the new COLLATION_KEY_BIT scalar function. For example, you can specify which collation should be used to sort the results of an SQL query.

Related concepts

"Language-aware collations for Unicode data" in Internationalization Guide

Unicode string literal enables specifying any Unicode character

Version 9.5 introduces the new Unicode string literal to accommodate supported characters that you cannot enter directly from a keyboard.

This literal makes it possible to specify a code point that you cannot type easily or at all from your keyboard. For example, you cannot type Japanese Hiragana characters on the English keyboard, but you can specify the Unicode equivalent character code.

This literal has the same semantics as a non-Unicode string literal except that you can specify characters by Unicode code point in addition to entering them directly.

Related reference

"Constants" in SQL Reference, Volume 1

Character-based processing for scalar functions supports variable character sizes

Version 9.5 contains new scalar functions that allow you to process your string data as a series of characters. You do not need to be aware of the byte representation for each character.

Each character in Unicode is made up of 1 to 4 bytes. To accommodate the variable character size, the following functions now have character-based processing by default:

- INSERT
- LEFT
- RIGHT
- STRIP
- TRIM

If you want to use byte-based processing, use the versions of these functions stored in the SYSFUN schema. For example, to use byte-based processing for the INSERT function, request SYSFUN.INSERT.

The OVERLAY scalar function, which is new for Version 9.5, also supports character-based processing.

Related reference

"INSERT scalar function" in SQL Reference, Volume 1

"LEFT scalar function" in SQL Reference, Volume 1

"RIGHT scalar function" in SQL Reference, Volume 1

"STRIP scalar function" in SQL Reference, Volume 1

"TRIM scalar function" in SQL Reference, Volume 1

"OVERLAY scalar function" in SQL Reference, Volume 1

Big5-HKSCS–Unicode conversion tables increase support for storing HKSCS data in Unicode databases

Version 9.5 provides new Unicode conversion tables that allow Big5-HKSCS clients to connect to and store HKSCS (Hong Kong Supplementary Character Set) data in Unicode databases.

The new conversion tables support HKSCS-2004 and Unicode 4.1 characters. Converting from Big5-HKSCS to Unicode does not generate Private Use Area (PUA) code points because all HKSCS-2004 characters have non-PUA assignments as of Unicode 4.1. However, during conversion from Unicode to Big5-HKSCS, PUA code points used in previous Unicode versions for HKSCS characters are converted to the appropriate Big5-HKSCS code points.

Related concepts

"Windows clients connecting to code page 950 databases" in Internationalization Guide

Related tasks

"Converting a code page 950 database containing HKSCS data to a Unicode database" in Internationalization Guide

Related reference

"Supported territory codes and code pages" in Internationalization Guide

UPPER (UCASE) and LOWER (LCASE) scalar functions support locales

The scalar functions UPPER (UCASE) and LOWER (LCASE) can now change the case of text using a locale-sensitive conversion. By default, UPPER and LOWER convert characters in the string without considering locale. For some characters, there is a different mapping between uppercase and lowercase characters when using a locale-based conversion.

For example, in Turkish, there are four distinct versions of the character i. The dotted i and the dotless i can be either upper or lower case. The dotted lower case i and the dotted upper case i are distinct from their dotless versions. When you specify the Turkish locale tr_TR, the UPPER scalar function converts the Latin small letter i to the Latin capital letter i with dot above, which is Unicode character code U&'\0130'. When you specify the Turkish locale, the LOWER scalar function converts the Latin capital letter i to the Latin small letter dotless i, which is Unicode character code U&'\0131'. If you do not specify the locale, the UPPER scalar function converts the Latin small letter i to the Latin capital letter i, which is Unicode character U&'\0049' and the LOWER scalar function converts the Latin capital letter i to the Latin small letter i, which is Unicode character U&'\0069'.

Related reference

"UPPER scalar function" in SQL Reference, Volume 1

"LOWER scalar function" in SQL Reference, Volume 1

"LOWER (Locale sensitive) scalar function" in SQL Reference, Volume 1

"UPPER (Locale sensitive) scalar function" in SQL Reference, Volume 1

"LCASE scalar function" in SQL Reference, Volume 1

"UCASE scalar function" in SQL Reference, Volume 1

"LCASE (Locale sensitive) scalar function" in SQL Reference, Volume 1

"UCASE (Locale sensitive) scalar function" in SQL Reference, Volume 1

FP1: Locale-sensitive UCA-based collations provide more options for ordering data

Starting with Version 9.5 Fix Pack 1, when you create a Unicode database, you can now specify a collation which provides culturally-expected sorting.

A locale-sensitive collation provides the expected ordering of data based on a specified locale, which includes information such as language and territory. These collations can also be tailored to provide case- and accent-insensitive ordering.

Locale-sensitive collations in DB2 Version 9.5 Fix Pack 1 are based on version 5.0 of the Unicode Collation Algorithm, which provides a specification for comparing two Unicode strings in a way that conforms to the requirements of the Unicode Standard.

You can also use locale-sensitive UCA-based collations with the COLLATION_KEY_BIT SQL scalar function.

Related concepts

"Locale-sensitive UCA-based collation" in Internationalization Guide

Chapter 15. Troubleshooting and problem determination enhancements

This section summarizes enhancements to troubleshooting and problem determination, including database resiliency and data consistency improvements, simplified error and log viewing, and new tools for tracking and logging errors.

Review this section to find out more details about improvements in troubleshooting and problem determination.

Data collection tool tracks unexpected errors

A new tool, db2fodc (first occurrence data capture), collects symptom-based data when there are unexpected error conditions detected within a DB2 instance. You can use this data to help troubleshoot error conditions that might occur in a database.

You can have the database manager automatically run the db2fodc tool, or you can run the tool manually. You must specify which symptoms will cause the database manager to run the tool. The symptoms that initiate the db2fodc tool include unexpected DB2 system errors due to signals (on UNIX operating systems), exceptions (on Windows operating systems), and data corruptions that result in outages, machine hangs, or panics. Specify the type of data to be collected by setting values in the **db2pdcfg** configuration parameter or the **DB2FODC** registry variable and its related parameters. Following the completion of the data collection, you must run the db2support tool to prepare and collect the resulting diagnostic files and prepare the package for submission to IBM Support.

This tool replaces some data gathering operations and incorporates other data gathering operations that are similar to those found in other troubleshooting tools such as db2support and tools used by IBM Support.

Related reference

"db2support - Problem analysis and environment collection tool command" in Command Reference

"General registry variables" in Data Servers, Databases, and Database Objects Guide

"db2pdcfg - Configure DB2 database for problem determination behavior command" in Command Reference

"db2fodc - DB2 first occurrence data collection command" in Command Reference

SQL administrative routine has been added for logging facilities

You can use the new PD_GET_DIAG_HIST system-defined routine to return event, notification, and diagnostic log records from various facilities, such as the optimizer statistics and the administration notification logs.

The routine also supports pre-filtering according to the customer impact value, the record type, and the record time stamp. The severity of the customer impact can range from informational to critical. You can control the scope and focus of diagnostic information.

Storage keys detect memory access problems

Version 9.5 supports storage keys, a new feature in IBM POWER6 processors and the AIX operating system that protects ranges of memory using hardware keys at a kernel thread level.

You can use storage keys to protect the buffer pool memory. Updates to a database are made while using the buffer pool; storage key protection reduces buffer pool memory corruption problems and limits errors that might halt the database. Attempts to illegally access the buffer pool by programming means cause an error condition that the database manager can detect and deal with.

Use the new **DB2_MEMORY_PROTECT** registry variable to enable storage key support.

Online data consistency checking is improved

You can use two new keywords, **high** and **indexdata**, with the INSPECT command and the db2inspect API to perform index checking.

The INSPECT command and the db2inspect API inspect a database for architectural integrity by checking the database for page consistency. The changes allow the INSPECT command and the db2inspect API to perform checks similar to those performed by db2dart and RUNSTATS to find data corruption. Access to all database objects except for the one being checked is allowed. The default behaviors for INSPECT and db2inspect are unchanged.

Index data consistency problem toleration is higher

In Version 9.5, when an index is no longer consistent, an error message (SQL0901N) is returned to the application instead of the database and instance being brought down.

When the error message is returned, you can use the INSPECT command or the db2inspect API to do online checking for the cause of the index inconsistency, while still allowing other applications access to those database objects not being evaluated. This type of error toleration is enabled only for regular indexes, on both partitioned and nonpartitioned tables, and for indexes on multidimensional

clustering indexes. This type of error toleration is not enabled for multidimensional clustering block indexes, composite block indexes, spatial indexes, and XML indexes.

Related reference

"db2Inspect API - Inspect database for architectural integrity" in Administrative API Reference

"INSPECT command" in Command Reference

Database resilience is improved during unexpected errors

Formerly, some error conditions in an application could bring down the database and the instance (database manager). Now, if the data integrity is unaffected and a single DB2 agent for the application with the error can be suspended or terminated, only that application is suspended or terminated.

When a trap, segmentation violation, or other exception occurs, the necessary diagnostic information is kept for review, the error condition is returned to the application, the status of the DB2 agent is changed, and the application is rolled back. Other applications unaffected by the error can run to completion. You can decide when to bring down and restart the database and instance.

Related concepts

"Troubleshooting DB2" in Partitioning and Clustering Guide

Part 2. What's changed

This section describes changed functionality, deprecated functionality, and discontinued functionality that you should keep in mind when coding new applications or when modifying existing applications.

Being aware of these changes facilitates your current application development and plans to migrate to Version 9.5.

In the following sections, topics that are associated with a specific fix pack include an ″FPx″ prefix at the beginning of the topic title, where *x* represents a fix pack level.

Chapter 16, "Changed functionality," on page 123
This chapter describes the changes to existing DB2 functionality, including changes related to database setup, database administration, application development, and CLP and system commands.

Chapter 17, "Deprecated functionality," on page 163
This chapter lists the deprecated functionality, which refers to specific functions or features that are supported but are no longer recommended and might be removed in a future release.

Chapter 18, "Discontinued functionality," on page 173
This chapter lists features and functionality that are unsupported in Version 9.5.

Chapter 19, "DB2 Version 9.1 fix pack changes that affect DB2 Version 9.5 usage," on page 179
This chapter describes features and functionality added or changed as part of Version 9.1 Fix Pack 3 (and earlier fix packs) that are applicable also to Version 9.5 but are not described in other sections of this book.

Chapter 16. Changed functionality

Changed functionality typically involves changes in default values or an outcome different from what would have occurred in previous releases. For example, an SQL statement that you used in Version 9.1 might produce different results in Version 9.5.

The changes are grouped by category.

Administration changes summary

Default code page for new databases is Unicode

When you create a new DB2 database, the default code page is Unicode. Previously, the default database code page was based on the environment of the application that you used to create the database.

Details

A Unicode database can accommodate characters from any language. Using Unicode allows a database to grow across language and territory boundaries, as your business grows.

Many modern development environments, such as Java and .NET, default to Unicode. Therefore, Unicode databases fit in better with these development environments, which reduces client/server communication costs.

Resolution

If you want to create a non-Unicode database, explicitly set the code set and territory for the database.

> **Related concepts**
>
> "Unicode implementation in DB2 Database for Linux, UNIX, and Windows" in Internationalization Guide
>
> **Related tasks**
>
> "Choosing the code page, territory, and collation for your database" in Internationalization Guide
>
> "Creating databases" in Data Servers, Databases, and Database Objects Guide

Some database manager configuration parameters have changed

Version 9.5 contains a number of new and changed database manager configuration parameters. In addition, some parameters have been deprecated or discontinued to reflect changes in DB2 functionality, the introduction of new parameters, or the removal of support.

The following new and changed database manager configuration parameters apply to DB2 Connect.

New database manager configuration parameters

Due to new features and functionality, Version 9.5 contains a number of new configuration parameters.

Table 7. Summary of new Version 9.5 database manager configuration parameters

Parameter name	Description	Details
cluster_mgr	Cluster manager name	Enables the database manager to communicate incremental cluster configuration changes to the specified cluster manager.

Changed database manager configuration parameters

The following table lists the configuration parameters with changes to their default values. All of these parameters can be dynamically updated without having to stop and restart the database instance.

Table 8. Summary of configuration parameters with changed default values

Parameter name	Description	Version 9.1 default value	Version 9.5 default value
agent_stack_sz	Agent stack size	64-bit Linux operating systems: 256	64-bit Linux operating systems: 1 024
comm_bandwidth	Communications bandwidth	The default value is calculated based on whether a high speed switch is being used.	The default value is calculated based on the speed of the underlying communications adapter. A value of 100 can be expected for systems using Gigabit Ethernet.
fenced_pool	Maximum number of fenced processes	The value of **max_coordagents**	AUTOMATIC
java_heap_sz	Maximum Java interpreter heap size	All operating systems: 2 048	HP-UX operating systems: 4 096 All other operating systems: 2 048
max_connections	Maximum number of client connections	The value of **max_coordagents**	AUTOMATIC
mon_heap_sz	Database system monitor heap size	UNIX: 90 Windows database server with local and remote clients: 66 Windows database server with local clients: 46	AUTOMATIC
num_poolagents	Agent pool size	The value of **maxagents**/2	AUTOMATIC

The following database manager configuration parameters have changed behaviors or have new ranges in Version 9.5.

Table 9. Summary of database manager configuration parameters with changed behaviors or new ranges

Parameter name	Description	Version 9.5 change
agent_stack_sz	Agent stack size	On Linux 64-bit platforms, the range of valid values has increased from 256 to 32 768, instead of 16 to 1 024.
federated_async	Maximum asynchronous TQs per query	The maximum range is no longer the value of **maxagents**/4. It is now 32 767.

Parameter name	Description	Version 9.5 change
instance_memory	Instance memory	The maximum range is no longer 4 294 967 295. On 32-bit platforms, it is 1 000 000, and on 64-bit platforms, it is 68 719 476 736. Instance memory now represents the partition-wide limit on memory consumption, instead of just the DBMS memory set size.
intra_parallel	Enable intra-partition parallelism	This parameter still controls the SQL access plan SMP parallelism, but it no longer controls index build parallelism. Instead, index build parallelism is enabled dynamically, on demand, but only temporarily for the duration of the CREATE INDEX operation, based upon some preliminary checks by the index manager.
max_coordagents	Maximum number of coordinating agents	In addition to supporting AUTOMATIC, the maximum range is no longer the value of **maxagents** minus the value of **num_initagents**. It is 64 000.
num_initagents	Initial number of agents in pool	The maximum range is no longer the value of **num_poolagents**. It is now 64 000.
num_initfenced	Initial number of fenced processes	The maximum range is no longer the sum of **max_connections** + (**maxagents** - **max_coordagents**). It is 64 000.

Deprecated and discontinued database manager configuration parameters

Due to changes in functionality, the introduction of new parameters, or the removal of support, the following database manager configuration parameters have been either deprecated or discontinued.

Table 10. Summary of deprecated configuration parameters

Parameter name	Description	Details and resolution
agentpri	Priority of agents	With the introduction of the new workload management capabilities, the need for this configuration parameter is reduced so this configuration parameter might be removed in a future release.
maxagents	Maximum number of agents	Rather than using the **maxagents** parameter to control the maximum number of database manager agents, set limits on the total number of connections allowed to the instance.
maxcagents	Maximum number of concurrent agents	Similarly, rather than using the **maxcagents** parameter to control the maximum concurrent number of database manager agents, use the Connection Concentrator and DB2 workload management capabilities to manage workload and resources on the system.
query_heap_sz	Query heap size	This configuration parameter is deprecated because it was used to support DB2 Universal Database for Linux, Windows, and UNIX Version 7 (or earlier) clients using the DB2RA protocol, and this protocol is no longer supported.

The following configuration parameters are discontinued:

Table 11. Summary of discontinued database manager configuration parameters

Parameter name	Description	Details and resolution
priv_mem_thresh	Private memory threshold	This parameter is not required because the database manager now uses a multithreaded architecture.

Related concepts

"Some database configuration parameters have changed" on page 139

Related reference

"RESET DATABASE CONFIGURATION command" in Command Reference

"Configuration parameters summary" in Data Servers, Databases, and Database Objects Guide

"DB2 server behavior changes" in Migration Guide

Some registry and environment variables have changed

In Version 9.5, there are a number of changes to registry and environment variables.

New defaults

Table 12. Registry variables with new default values

Registry variable	Version 9.1 default setting	Version 9.5 default setting
DB2INSTPROF	NULL	ProgramData\IBM\DB2\ on the Windows Vista operating system, and Documents and Settings\All Users\Application Data\IBM\DB2\ on the Windows 2003 or XP operating systems.

New values

Table 13. Registry variables with new values

Registry variable	New values
DB2_EVMON_STMT_FILTER	Starting in DB2 Version 9.5 Fix Pack 1, this variable has new options that allow users to determine which rules to apply to which event monitors. Each option represents an integer value mapping to a specific SQL operation.
DB2_MDC_ROLLOUT	This variable has a new value, DEFER, and a new description, IMMEDIATE (which is the same as the current ON, YES, 1, and TRUE settings). You can now control whether rollout deletes on multidimensional clustering tables use immediate index cleanup rollout (the default behavior) or deferred index cleanup rollout. For more information, see "MDC rollout deletions are faster with the deferred index cleanup option" on page 54.

Table 13. Registry variables with new values (continued)

Registry variable	New values
DB2_WORKLOAD	This variable has two new values: 1C and TPM. These settings allow you to configure a set of registry variables in your database for applications provided by 1C and for Tivoli Provisioning Manager.

Changed behaviors

Table 14. Registry variables with changed behaviors

Registry variable	Changed behavior
• **DB2_RESOURCE_POLICY** • **DB2_LARGE_PAGE_MEM** • **DB2_PINNED_BP** • **DB2PRIORITIES**	These registry variables are not supported for non-root installations. Updates to them are not permitted. For more information, see "Non-root users can install and configure DB2 products (Linux and UNIX)" on page 102.
DB2CLIINIPATH	If you choose the migrate action when installing DB2 Version 9.5 on Windows operating systems, **DB2CLIINIPATH** is set to the location of the existing db2cli.ini configuration file for the DB2 copy that you want to migrate. However, if you migrate your instances after installing a DB2 copy and you want to maintain the pre-migration location for your configuration file, set this variable to that location.
DB2MEMMAXFREE	If you set this registry variable on Linux and UNIX operating systems, instance migration sets it to null. Setting this variable is no longer necessary because the database manager now uses a threaded engine model, so private memory is shared by all database manager threads. **Note:** Do not set this variable. Doing so will likely hurt performance and might lead to unexpected behavior. For more information, see "Simplified multithreaded architecture reduces the Total Cost of Ownership (TCO)" on page 28.
DB2_EXTENDED_IO_FEATURES	You can no longer use this registry variable option to set the I/O priority for online backups. The API that was used to set the I/O priority is process based. Version 9.5 uses a thread-based model, and there is currently no equivalent API for setting thread-based I/O priority. For more information, see "Simplified multithreaded architecture reduces the Total Cost of Ownership (TCO)" on page 28.

Table 14. Registry variables with changed behaviors (continued)

Registry variable	Changed behavior
DB2_USE_DB2JCCT2_JROUTINE	The default setting of this registry variable now means that the default driver for Java stored procedures and user-defined functions is the IBM Data Server Driver for JDBC and SQLJ. In previous releases, the default JDBC driver was the DB2 JDBC Type 2 driver. For more information, see "Default JDBC driver has changed for Java routines" on page 148.

New variables

These registry variables are new in Version 9.5:

Table 15. Registry variables added

Registry variable	Description
DB2_CAPTURE_LOCKTIMEOUT	This registry variable specifies to log descriptive information about lock timeouts at the time that they occur. A text report is written and stored in a file for each lock timeout. For more information, see "Lock timeout diagnosis has been enhanced" on page 38
DB2_EVMON_EVENT_LIST_SIZE	This registry variable specifies the maximum number of bytes that can be queued to be written to a particular event monitor. After this limit is reached, agents attempting to send event monitor records wait until the queue size drops below this threshold. For more information, see "Workload management improvements provide better control" on page 41.
DB2FODC	This registry variable controls a set of troubleshooting-related parameters used in First Occurrence Data Collection (FODC) by extending the functionality previously available in the **DB2FFDC** registry variable. It is intended as a way for you and IBM service analysts to control what is collected by the DB2 product during FODC scenarios. Use **DB2FODC** to control different aspects of data collection in outage situations. For more information, see "Data collection tool tracks unexpected errors" on page 117.

Table 15. Registry variables added (continued)

Registry variable	Description
DB2_HADR_PEER_WAIT_LIMIT	Starting DB2 Version 9.5 Fix Pack 1, when this registry variable is set, the HADR primary database will break out of peer state if logging on the primary database has been blocked for the specified number of seconds because of log replication to the standby.
DB2_KEEP_AS_AND_DMS_CONTAINERS_OPEN	This registry variable enables each DMS table space container to have a file handle opened until the database is deactivated and as a result might improve query performance. It should be used in pure DMS environments only. This variable is available with DB2 Version 9.5 Fix Pack 1.
DB2LDAPSecurityConfig	This registry variable specifies the location of the IBM LDAP security plug-in configuration file.
DB2_LOGGER_NON_BUFFERED_IO	Starting in DB2 Version 9.5 Fix Pack 1, this registry variable enables direct I/O on the log file system.
DB2_MEMORY_PROTECT	This registry variable enables a memory protection feature that uses storage keys to prevent data corruption in the buffer pool caused by invalid memory access. Memory protection works by identifying the times when the DB2 engine threads should have access to the buffer pool memory. When you set **DB2_MEMORY_PROTECT** to YES, any time that a DB2 engine thread tries to illegally access buffer pool memory, that engine thread traps. For more information, see "Database resilience is improved during unexpected errors" on page 119.
DB2_OPTSTATS_LOG	**DB2_OPTSTATS_LOG** specifies the attributes of the statistics event logging files that are used to monitor and analyze statistics collection-related activities. When you do not set **DB2_OPTSTATS_LOG** or you set it to ON, statistics event logging is enabled, allowing you to monitor system performance and keep a history for better problem determination. For more information, see "Real-time statistics gathering ensures that the latest statistics are used for optimization" on page 27.

Table 15. Registry variables added (continued)

Registry variable	Description
DB2_SET_MAX_CONTAINER_SIZE	This registry variable sets the maximum container size for a database. When you use this registry variable, a new container is created in the database, based on the existing storage path, after the specified container limit is reached by an automatic storage-managed table space. For more information, see "Table spaces use space more efficiently" on page 38.
DB2_SYSTEM_MONITOR_SETTINGS	The registry variable controls a set of parameters that allows you to modify the behavior of various aspects of DB2 monitoring. For instance, the parameter **OLD_CPU_USAGE** controls how an instance obtains CPU times on Linux platforms.
DB2_THREAD_SUSPENSION	This registry variable enables or disables the DB2 thread suspension feature. The variable allows you to control whether a DB2 instance sustains a trap by suspending a faulty engine thread (a thread that has tried to access the buffer pool memory illegally). For more information, see "Database resilience is improved during unexpected errors" on page 119.
DB2_UPDDBCFG_SINGLE_DBPARTITION	This registry variable allows you to specify whether database updates and resets affect only a specific partition. If you do not set the variable, updates and changes to a database configuration apply to all database partitions. For more information, see "Database configuration across multiple partitions has been simplified" on page 29.

Related concepts

"Some registry and environment variables are deprecated" on page 163
"Some registry and environment variables are discontinued" on page 175

Database audits now require SECADM authority

Managing audits at the database level now lies with the security administrator (who holds SECADM authority). The security administrator can now manage (create, alter, drop, and comment on) trusted context objects, roles, and audit policies.

Details

In addition to having extended abilities, the security administrator is the *only* user who can manage the auditing of a database; the system administrator (who holds SYSADM authority) no longer has this ability but can still manage the instance-level audit. As well as being able to configure the audit for a database, the security administrator has sufficient SQL access to perform the following tasks:

- List archived audit log files available for extract
- Issue the ARCHIVE command
- Extract an audit log file into a delimited file

Resolution

Ensure that the proper authorities are assigned. The system administrator still has sufficient authority to manage the audit logs after they are on disk by using the db2audit command, but cannot control what events are audited. The security administrator does not have sufficient authority to run the db2audit command; SYSADM authority is required.

> **Related concepts**
>
> "Audit facility performance and administration have been enhanced" on page 49
>
> "Storage and analysis of audit logs" in Database Security Guide
>
> **Related reference**
>
> "db2audit - Audit facility administrator tool command" in Command Reference

Data compression dictionary is created automatically

In Version 9.5, a data compression dictionary is created automatically in some cases.

Details

In Version 9.1, you have to manually create the data compression dictionary by performing a classic (offline) table reorganization. You must first set the COMPRESS attribute for the table to YES. However, in Version 9.5, after you set the COMPRESS attribute set to YES, the table is eligible for automatic data compression dictionary creation after there is enough data in the table. As a result, using the INSERT statement, the LOAD command with the **INSERT** or **REPLACE** options, the IMPORT command with the **INSERT** option, or the REDISTRIBUTE command will result in automatic creation of the data compression dictionary if the database system determines that there is enough data in the table to warrant creation of the dictionary. You do not need to perform an explicit classic (offline) table reorganization to create the data compression dictionary.

In Version 9.1, if the table COMPRESS attribute is set to YES, no compression dictionary exists in the table yet, and at least one record of valid length exists in the table, then a request to reorganize the table with the **KEEPDICTIONARY** option will build a compression dictionary for the table. For the same scenario in Version 9.5, a compression dictionary will not be built unless the size of the table exceeds a threshold of approximately 2 MB and there is sufficient user data (at least 700KB) contained within the table once the threshold size is reached.

In Version 9.1, all of the data rows of a valid record size are used to build the data compression dictionary. If all of the records in the table at the time that the data

compression dictionary is being built are smaller than the minimum record length, an SQL2220W error message is returned. As long as there is at least one record of valid length in the table, the data compression dictionary is built. However, in Version 9.5, there is no record length criteria that governs which row participates in the creation of the data compression dictionary. The SQL2220W error message is not generated if the lengths of all data records sampled for use during the data compression dictionary build are less than the applicable minimum record length.

Resolution

After you create a table or alter a table with the COMPRESS attribute set to YES, you do not need to do any work to create a data compression dictionary.

Related concepts

"Automatic (compression) dictionary creation (ADC)" in Data Servers, Databases, and Database Objects Guide

Target tables for write-to-table event monitors have changed

The data types or lengths in the target tables for write-to-table event monitors have changed between Version 9.1 and Version 9.5 for a subset of monitor elements. These changes enable you to capture more monitoring data in the target tables.

Details

The following monitor elements have changed data types or lengths:

Table 16. Monitor elements with changed data types or lengths

Element name	Description	Version 9.1 data type and length	Version 9.5 data type and length
appl_id	Application ID	CHAR (64)	VARCHAR (64)
appl_id_holding_lk	Application ID holding lock	CHAR (64)	VARCHAR (64)
auth_id	Authorization ID	VARCHAR (30)	VARCHAR (128)
corr_token	DRDA correlation token	CHAR (64)	VARCHAR (64)
creator	Application creator	VARCHAR (30)	VARCHAR (128)
execution_id	User login ID	VARCHAR (30)	VARCHAR (128)
package_name	Package name	CHAR (8)	VARCHAR (128)
rolled_back_appl_id	Rolled back application	CHAR (64)	VARCHAR (64)
table_schema	Table schema name	VARCHAR (30)	VARCHAR (128)

Resolution

Applications querying the monitor elements listed in the table for write-to-table event monitors return errors unless you do the following steps:

1. Query the SYSCAT.EVENTTABLES view to determine all of the target tables for write-to-table event monitors, and rename these target tables. If you no longer need the data in these target tables, you can drop these tables instead of renaming them.
2. Drop the event monitors, and then redefine them.
3. Modify any applications that access the target tables to accept the VARCHAR data type rather than the fixed CHAR data type, or change the host variables to accept the new length. For details on the type of structure required for these

data types, see "Supported SQL data types in C and C++ embedded SQL applications" in *Developing Embedded SQL Applications* .

Some system catalog views and built-in routines have been added and changed

To support new features in Version 9.5, system catalog views, system built-in routines, administrative routines and views have been added and modified.

System catalog view changes

The following system catalog views have changed in Version 9.5. Most modifications to catalog views consist of new columns, changed column data types, and increased column lengths.

- 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.PASSTHRUAUTH
- 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

The following system catalog views have been added in Version 9.5:
- 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

System-defined built-in function changes

The following system-defined built-in functions have been added or modified in Version 9.5:

Table 17. New and modified system-defined built-in functions

Function name	Summary of changes
• BITAND, BITOR, BITANDNOT, BITNOT, and BITXOR • COLLATION_KEY_BIT • COMPARE_DECFLOAT • DECFLOAT • DECODE • GREATEST • LEAST • MAX • MIN • NORMALIZE_DECFLOAT • NVL • QUANTIZE • RID_BIT and RID • TOTALORDER	If you have user-defined functions with the same names as these built-in functions and you do not fully qualify them in your queries, the resolution path calls the new built-in functions instead. Fully qualify your user-defined function calls with the schema names if you do not want to use these built-in functions.
• INSERT • LEFT • OVERLAY • RIGHT • STRIP • TRIM	To accommodate the variable character size in Unicode, there are new character-sensitive versions of these functions with the SYSIBM schema and one additional parameter to indicate the string unit. If you use the default SQL path and do not specify the string unit, the new versions of these functions are called. The behavior is compatible with the behavior in previous releases, but there are some differences. To call the same function available in previous releases, explicitly qualify the function name with the SYSFUN schema.

System-defined administrative routine and view changes

The following administrative views and routines have changed in Version 9.5:
• ADMIN_CMD procedure
• ADMINTABINFO administrative view
• AUTHORIZATIONIDS administrative view
• ENV_PROD_INFO administrative view
• PRIVILEGES administrative view
• SNAPAPPL administrative view
• SNAPAPPL_INFO administrative view
• SNAPBP administrative view
• SNAPDB administrative view
• SNAPDBM administrative view
• SNAPDYN_SQL administrative view
• SNAPTAB_REORG administrative view and SNAP_GET_TAB_REORG table function

The following administrative views and routines have been added in Version 9.5:

- ADMIN_GET_DBP_MEM_USAGE table function
- ADMINTABCOMPRESSINFO administrative view and ADMIN_GET_TAB_COMPRESS_INFO table function
- AUDIT_ARCHIVE procedure and table function
- AUDIT_DELIM_EXTRACT procedure
- AUDIT_LIST_LOGS table function
- AUTH_LIST_AUTHORITIES_FOR_AUTHID table function
- AUTH_LIST_ROLES_FOR_AUTHID table function
- AUTOMAINT_GET_POLICY procedure
- AUTOMAINT_GET_POLICYFILE procedure
- AUTOMAINT_SET_POLICY procedure
- AUTOMAINT_SET_POLICYFILE procedure
- ENV_FEATURE_INFO administrative view
- ENV_SYS_RESOURCES administrative view
- EXPLAIN_FORMAT_STATS scalar function
- PD_GET_DIAG_HIST table function
- SNAP_GET_APPL_V95 table function
- SNAP_GET_APPL_INFO_V95 table function
- SNAP_GET_BP_V95 table function
- SNAP_GET_DB_V95 table function
- SNAP_GET_DBM_V95 table function
- SNAP_GET_DYN_SQL_V95 table function
- WLM_CANCEL_ACTIVITY procedure
- WLM_CAPTURE_ACTIVITY_IN_PROGRESS procedure
- WLM_COLLECT_STATS procedure
- WLM_GET_ACTIVITY_DETAILS table function
- WLM_GET_QUEUE_STATS table function
- WLM_GET_SERVICE_CLASS_AGENTS table function
- WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES table function
- WLM_GET_SERVICE_SUBCLASS_STATS table function
- WLM_GET_SERVICE_SUPERCLASS_STATS table function
- WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES table function
- WLM_GET_WORKLOAD_STATS table function

The following table functions have been deprecated in Version 9.5:
- ADMIN_GET_TAB_INFO table function
- SNAP_GET_APPL table function
- SNAP_GET_APPL_INFO table function
- SNAP_GET_BP table function
- SNAP_GET_DB_V91 table function
- SNAP_GET_DBM table function
- SNAP_GET_DYN_SQL_V91 table function

Review the list of the "Deprecated SQL administrative routines and their replacement routines or views" in *Administrative Routines and Views* to determine additional changes that might impact your applications and scripts.

Memory Visualizer displays maximum memory consumption

Memory Visualizer now displays the maximum application memory consumption by database, from the new **appl_memory** configuration parameter, and the maximum memory consumption by an instance, from the updated **instance_memory** configuration parameter.

The Memory Visualizer also displays the values for the following configuration parameters, which now accept the AUTOMATIC setting:

- **mon_heap_sz**
- **stmtheap**
- **stat_heap_sz**
- **applheapsz**

Values for the following deprecated configuration parameters are not displayed for Version 9.5 databases but are still supported for databases from earlier DB2 versions:

- **appgroup_mem_sz**
- **groupheap_ratio**
- **app_ctl_heap_sz**
- **query_heap_sz**

  Related concepts

  "Memory Visualizer overview" in System Monitor Guide and Reference

  "Memory configuration has been simplified" on page 30

  Related reference

  "instance_memory - Instance memory configuration parameter" in Data Servers, Databases, and Database Objects Guide

  "appl_memory - Application Memory configuration parameter" in Data Servers, Databases, and Database Objects Guide

Backup image read and write privileges have been changed

Backup images are now readable and writable by the instance owner only.

Details

Starting with Version 9.5, backup images are generated with file mode 600 on Linux and UNIX operating systems, which provides read and write privileges to the instance owner only. On Windows operating systems with extended security enabled, only members of the DB2ADMNS (and Administrators) group have access to the backup images.

In previous versions, backup images on Linux and UNIX operating systems were generated with file mode 640, which meant that they were readable by other members of the instance owner's primary group. Because members of this group might not have the authority to read backup images, they are now excluded by default.

The file mode for load-copy images has not changed in Version 9.5 because the HADR feature requires load copy images to be readable by a different instance.

Resolution

If you need other users to have access to backup images, you can change the file permissions after generating the backups.

Migrate button on the DB2 Launchpad has been moved (Windows)

The **Migrate** button on the DB2 Launchpad has been moved.

Details

Prior to Version 9.5 on Windows operating systems, the **Migrate** button was on the DB2 Launchpad. Starting with Version 9.5, the **Migrate** button is on the DB2 Setup wizard, on the panel Select the DB2 copy to work with.

Resolution

To access the **Migrate** action:
1. Start the DB2 Launchpad.
2. From the **Install a Product** tab, click **Work with Existing**
3. On the Select the DB2 copy to work with panel, select the DB2 copy that you want to migrate.
4. Click **Launch DB2 Setup wizard**.

Table index size has increased

Each index on every non-empty table now contains one more page.

Details

The new real time statistics functionality and faster multidimensional clustered table (MDC) rollout require additional index space. If you have an index on a non-empty table created in a previous release, the index size might increase the first time one of the following occurs.
- The index statistics are collected by the RUNSTATS utility.
- The index is accessed or updated by MDC rollout with deferred index cleanup.
- The index is rebuilt or recreated.
- A REORG INDEX command with the CLEANUP option is issued against the index.
- A large volume of index maintenance (including updates, deletes, and inserts) changes the index statistics.

In this case, the operation might fail with error message SQL0289N (Unable to allocate new pages in table space *tablespace-name*).

Resolution

Increase the size of the index table space.

Related tasks

"Adding or extending DMS containers" in Data Servers, Databases, and Database Objects Guide

Related reference

"ALTER TABLESPACE statement" in SQL Reference, Volume 2

"CREATE TABLESPACE statement" in SQL Reference, Volume 2

Table truncation invalidates dynamic statement cache

If table truncation occurs and real time statistics collection is enabled, all entries depending on the truncated table in the dynamic statement cache will be invalidated.

Details

When you use the IMPORT command with the REPLACE option, it deletes all existing data from the table by truncating the data object, and inserts the imported data. Previously, table truncation did not invalidate the statement cache. This behavior has changed when real time statistics collection is enabled.

Invalidating dynamic statements forces them to be recompiled the next time they are executed. This provides an opportunity to choose a more optimal access plan with the most current statistics. However, it may also cause a performance impact.

Resolution

Be aware of the potential performance impact.

Concurrency is improved for the ALLOW NO ACCESS option of the REFRESH TABLE and SET INTEGRITY statements

When you use the REFRESH TABLE statement or the SET INTEGRITY statement with the ALLOW NO ACCESS option, statements and utilities which use the Uncommitted Read isolation level will now be allowed concurrent read access to the table.

Details

By acquiring an X lock instead of a Z lock on the target table, transactions that are running concurrently under the Uncommitted Read isolation level now have read access on the target table being processed by the REFRESH TABLE statement or the SET INTEGRITY statement. Furthermore, utilities running concurrently that require only the Uncommitted Read isolation level can also read the target table.

Resolution

You do not need to make any code changes. The concurrency of applications accessing the target table being processed will improve.

Database setup changes summary

Some database configuration parameters have changed

Version 9.5 contains a number of new and changed database configuration parameters. In addition, some parameters have been deprecated or discontinued to reflect changes in DB2 functionality, the introduction of new parameters, or the removal of support.

New database configuration parameters

Due to new features and functionality, Version 9.5 contains a number of new database configuration parameters.

Table 18. New Version 9.5 database configuration parameters

Parameter name	Description	Details
appl_memory	Application memory	Allows you to control the maximum amount of application memory that can be allocated by all DB2 database agents to service application requests. By default, its value is set to AUTOMATIC, meaning that all application memory requests will be allowed if the total amount of memory allocated by the database partition is within the **instance_memory** limits.
auto_del_rec_obj	Automated deletion of recovery objects	Specifies whether database log files, backup images, and load copy images are deleted when their associated recovery history file entry is pruned.
auto_stmt_stats	Automatic statement statistics	Enables and disables the real-time statistics gathering functionality. It is a child parameter of the **auto_runstats** configuration parameter and is enabled only if the parent parameter is also enabled.
decflt_rounding	Decimal floating point rounding	Allows you to specify the rounding mode for decimal floating-point type (DECFLOAT). The rounding mode affects decimal floating-point operations in the server, and when you use the LOAD command.
enable_xmlchar	Enable conversion to XML	Determines whether XMLPARSE operations can be performed on non-BIT DATA CHAR (or CHAR-type) expressions in an SQL statement. When you use pureXML features in a non-Unicode database, the XMLPARSE function can cause character substitutions to occur as SQL string data is converted from the client code page into the database code page, and then into Unicode for internal storage. This parameter is also valid for Unicode databases, even though there is no danger of substitution characters with such databases.
hadr_peer_window	HADR peer window configuration	Helps to ensure data consistency by ensuring that a HADR primary-standby database pair continues to behave as though still in the peer state, for the configured amount of time, even if the primary database loses the connection with the standby database.
wlm_collect_int	Workload management collection interval	Specifies the interval of time, in minutes, between attempts to collect workload management statistics and attempts to reset such statistics. The interval that you specify is used only for the catalog partition. You can use the statistics collected by the workload management monitoring function to monitor both short-term and long-term system behavior.

Changed database configuration parameters

The following table lists the database configuration parameters with changes to their default values. All of these parameters can be dynamically updated without having to stop and restart the database instance.

Table 19. Database configuration parameters with changed default values

Parameter name	Description	Version 9.1 default value	Version 9.5 default value
applheapsz	Application heap size	32-bit and 64-bit database servers with local and remote clients: 256 32-bit partitioned database server with local and remote clients: 64 64-bit partitioned database server with local and remote clients: 128	AUTOMATIC
catalogcache_sz	Catalog cache size	The default value used to calculate the page allocation is four times the value specified for the **maxappls** configuration parameter.	The default value used to calculate the page allocation is five times the value specified for the **maxappls** configuration parameter.
database_memory	Database shared memory size	On AIX and Windows operating systems: AUTOMATIC On Linux, HP-UX, Solaris operating systems: COMPUTED	On the Linux operating system, the value of this parameter defaults to the AUTOMATIC setting on RHEL5 and on SUSE 10 SP1 and newer distributions. On all other validated Linux distributions, the value of this parameter defaults to COMPUTED if the kernel does not support AUTOMATIC For all other operating systems: AUTOMATIC
dbheap	Database heap	UNIX: 1200 Windows database server with local and remote clients: 600 Windows 64-bit database server with local clients: 600 Windows 32-bit database server with local clients: 300	AUTOMATIC
stat_heap_sz	Statistics heap size	4384	AUTOMATIC
stmtheap	Statement heap size	32-bit platforms: 2048 64-bit platforms: 4096	AUTOMATIC

The following database configuration parameters have changed behaviors or have new ranges in Version 9.5.

Table 20. Database configuration parameters with changed behaviors or new ranges

Parameter name	Description	Version 9.5 change
applheapsz	Application heap size	In previous releases, this parameter indicated the amount of memory for each database agent. Now, this parameter indicates the total amount of memory for an application.

Table 20. Database configuration parameters with changed behaviors or new ranges (continued)

Parameter name	Description	Version 9.5 change
maxfilop	Maximum number of files that can be concurrently open for a database	This parameter now indicates the maximum number of file handles that can be concurrently open for a database. In previous releases, this parameter indicated the maximum number of file handles that could be open for each database agent.
sortheap	Sort heap size	The OLAP functions now use sort heap memory instead of application heap memory to provide a higher limit for memory resources.

Deprecated and discontinued configuration parameters

Due to changes in functionality, the introduction of new parameters, or the removal of support, the following configuration parameters have been either deprecated or discontinued.

Table 21. Summary of deprecated database configuration parameters

Parameter name	Description	Details and resolution
app_ctl_heap_sz	Application control heap size	Under the new memory model, a single application shared memory set is created for a database (on each partition) and shared by all applications connected to it. In Version 9.5, instead of using these three configuration parameters to determine how many applications can fit in each application group, use the new **appl_memory** configuration parameter to set the maximum limit on total application memory consumption.
appgroup_mem_sz	Maximum size of application group memory set	
groupheap_ratio	Percent of memory for application group heap	
logretain	Log retain enable	Both of these parameters have been replaced with **logarchmeth1**, the primary log archive method configuration parameter. Rather than using **logretain** to retain active log files for rollforward recovery, specify the LOGRETAIN value for **logarchmeth1**. Similarly, rather than using **userexit** to enable log archiving through a userexit program, specify the USEREXIT value for **logarchmeth1**. • Turning on **logretain** sets the LOGRETAIN value to **logarchmeth1**.
userexit	User exit enable	• Turning on **userexit** sets the USEREXIT value to **logarchmeth1**. Turning on both **logretain** and **userexit** sets the USEREXIT value to **logarchmeth1**. For instance, issuing the command `update db cfg using logretain on` sets LOGARCHMETH1 to LOGRETAIN.
numsegs	Default number of SMS containers	This configuration parameter is deprecated because you specify multiple containers for SMS table spaces in the CREATE DATABASE command.

The following database configuration parameters are discontinued:

Table 22. Summary of discontinued database configuration parameters

Parameter name	Description	Details and resolution
estore_seg_sz	Extended storage memory segment size	Support for the Extended Storage feature is discontinued. You should remove the use of extended storage configuration parameters. If you want to be able to allocate more memory, consider upgrading to a 64-bit operating system.
num_estore_segs	Number of extended storage memory segments	

Related concepts

"Some database manager configuration parameters have changed" on page 123

Related reference

"RESET DATABASE CONFIGURATION command" in Command Reference

"Configuration parameters summary" in Data Servers, Databases, and Database Objects Guide

"DB2 server behavior changes" in Migration Guide

Concurrent I/O and Direct I/O are enabled by default (AIX, Linux, Solaris and Windows)

Prior to Version 9.5, FILE SYSTEM CACHING was the default attribute for all of the table spaces created with the CREATE TABLESPACE statement and CREATE DATABASE command. In Version 9.5, the NO FILE SYSTEM CACHING attribute is implied on system configurations where it can be used.

Details

The FILE SYSTEM CACHING and NO FILE SYSTEM CACHING attributes specify whether I/O operations are to be cached at the file system level. You specify these attributes using the following interfaces: the CREATE TABLESPACE statement, the CREATE DATABASE command, and the sqlecrea() API (using the **sqlfscaching** field of the SQLETSDESC structure).

In Version 9.5, NO FILE SYSTEM CACHING is used by default if you do not specify it on the CREATE TABLESPACE statement or the CREATE DATABASE command on many system configurations. For the list of the system configurations that support CIO, DIO, or file system caching, see "File system caching configurations".

Resolution

If you do not want to use the new behavior, specify FILE SYSTEM CACHING when you create a table space.

If you experience performance degradation with the new behavior and self tuning memory is disabled, use one of the following workarounds:

- Enable self tuning memory, and set the buffer pool size and **database_memory** configuration parameter to AUTOMATIC.
- Increase the buffer pool size manually.
- Disable DIO and CIO by using the ALTER TABLESPACE statement with the FILE SYSTEM CACHING attribute.

Related concepts

"Management of multiple database buffer pools" in Tuning Database Performance

"Self-tuning memory" in Data Servers, Databases, and Database Objects Guide

"Table spaces without file system caching" in Data Servers, Databases, and Database Objects Guide

Related reference

"ALTER BUFFERPOOL statement" in SQL Reference, Volume 2

"ALTER TABLESPACE statement" in SQL Reference, Volume 2

"CREATE TABLESPACE statement" in SQL Reference, Volume 2

"sqlecrea API - Create database" in Administrative API Reference

"CREATE DATABASE command" in Command Reference

Extended security requires users to belong to the DB2ADMNS or DB2USERS group (Windows Vista)

If you enable extended security on Windows Vista, users must belong to the DB2ADMNS or DB2USERS group to run local DB2 commands and applications because of an extra security feature (User Access Control) that limits the privileges that local administrators have by default.

Details

If users do not belong to one of these groups, they do not have read access to local DB2 configuration or application data.

Resolution

- Add users who need to run local DB2 applications and tools to the DB2ADMNS or DB2USERS group if you enable extended security. When you make group membership changes, they take effect the next time that users log in.
- Use the **DB2 Command Window - Administrator** shortcut to launch DB2 commands and tools that require operating system local administrator authority.

  Related concepts

  "Extended Windows security using DB2ADMNS and DB2USERS groups" in Database Security Guide

  Related reference

  "Required user accounts for installation of DB2 server products (Windows)" in Quick Beginnings for DB2 Servers

Configuration and runtime data file default locations have changed (Windows)

The default locations of all configuration and runtime data files, such as instance directories and the db2cli.ini file, have been changed to comply with requirements for Windows Vista certification.

Details

The default locations are as follows:

- On the Windows XP and Windows 2003 operating systems: Documents and Settings\All Users\Application Data\IBM\DB2*Copy name*
- On the Windows Vista operating system: ProgramData\IBM\DB2\ *Copy name*

Fix pack installations do not require subsequent manual steps (Linux and UNIX)

By default, the installFixPack command now automatically updates the instances and DAS related to a given installation path. The BIND command is also now launched automatically when the database is reconnected or when applications are restarted.

Details

In previous releases, you had to update your instance after applying fix packs, resulting in manual steps, and you also had to manually bind packages.

Resolution

Modify any deployment scripts that update instances and the DAS after fix pack installations.

> **Related tasks**
>
> "Applying fix packs" in Troubleshooting Guide

Some configuration parameters are impacted by simplified memory configuration

Due to simplified memory configuration in Version 9.5, the database manager now sets a number of configuration parameters to AUTOMATIC: specifically, during instance migration or creation and during database migration or creation.

Details

The AUTOMATIC setting indicates that the database configuration parameters are automatically tuned for you according to your system resources. The following table lists the configuration parameters that are impacted:

Table 23. Configuration parameters set to AUTOMATIC in Version 9.5

Configuration parameters	Set to AUTOMATIC upon instance migration or instance creation	Set to AUTOMATIC upon database migration	Set to AUTOMATIC upon database creation
applheapsz		X	X
dbheap		X	X
instance_memory	X		
mon_heap_sz	X		
stat_heap_sz		X	X
stmtheap			X

As part of the simplified memory configuration, the following elements are deprecated:

- Configuration parameters **appgroup_mem_sz**, **groupheap_ratio**, **app_ctl_heap_sz**, and **query_heap_sz**. These configuration parameters are replaced with the new **appl_memory** configuration parameter.
- The **-p** parameter of the db2mtrk memory tracker command. This parameter, which lists private agent memory heaps, is replaced with the **-a** parameter, which lists all application memory consumption.

Related concepts

"Memory configuration has been simplified" on page 30

"db2mtrk command has changed" on page 158

"Some database configuration parameters have changed" on page 139

"Some database manager configuration parameters have changed" on page 123

Related reference

"db2mtrk - Memory tracker command" in Command Reference

Information Integrator product identifier values have changed

In Version 9.5, the product identifer values for Information Integrator products have changed to match the names of the Version 9.5 Information Integrator products.

Details

A product identifier value is an input parameter that the License Management Tool (db2licm utility) uses. You can list product identifier values using the db2licm command with the **-l** option.

The product identifier values for the following Information Integrator products changed in Version 9.5:

Table 24. Changed product identifier values

Product name	Version 9.5 product identifier	Version 9.1 product identifier
WebSphere Data Event Publisher	wsep	wsiip
WebSphere Federation Server	wsfs	wsiif
WebSphere Replication Server	wsrs	wsiir

Resolution

Update any scripts or applications that parse the output of the db2licm command.

Related reference

"db2licm - License management tool command" in Command Reference

Database partitioning is now available only through DB2 Warehouse

In Version 9.5, database partitioning is available only as part of the DB2 Warehouse products.

Details

In DB2 Version 9.1, the Database Partitioning Feature was available with DB2 Enterprise Server Edition (ESE). In DB2 Version 9.5, the Database Partitioning Feature is available only as part of the DB2 Warehouse products. Existing DB2 ESE customers using the Database Partitioning Feature will be upgraded to the IBM Base Warehouse Feature for DB2. New DB2 Version 9.5 customers who would like to use the Database Partitioning Feature should upgrade to a DB2 Warehouse product.

Resolution

To use database partitioning, you can use the DB2 Warehouse installer. If you do not need the additional warehousing tools, you can run the DB2 ESE installer and then apply the Warehouse license certificate to activate database partitioning. The database partitioning code continues to be installed by the DB2 ESE installer.

The DB2 ESE setup program is located in *DB2W INSTALL FILES ROOT*/dwe/Ese in the DB2 Warehouse media. After you install the DB2 product, you can deploy the DB2 license certificate that comes with the DB2 Warehouse activation CD. The license certificate is located in *DB2W ACTIVATION CD*/profile/license. For example, if you installed DB2 Warehouse Enterprise Edition, you would issue the following command to deploy the DB2 license certificate:

```
db2licm -a /mnt/db2w_activation_cd/profile/license/dwee.lic
```

For more information, see:
* "DB2 Version 9.5 database partitioning" at www.ibm.com/support/docview.wss?&uid=swg21284831
* "Setting up a partitioned database environment" in *Quick Beginnings for DB2 Servers*
* "Applying DB2 licenses" in *Quick Beginnings for DB2 Servers*
* "DB2 Version 9.5 editions: feature and function support" at http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/topic/com.ibm.db2.luw.licensing.doc/doc/r0053238.html

Header files are no longer installed by default

When installing DB2 database products, the Typical install option does not install header files in the /include directory.

Details

In previous releases, header files were included as part of the typical install options.

Resolution

To install header files as part of your installation, select the Custom install option.

To add header files after the product installation is completed, run the installation program again, and select to modify an existing install. Select the Custom install option and select the features which contain the desired header files.

Catalogs collate using IDENTITY sequence in Unicode databases

In DB2 Version 9.5, the database catalog tables and views are created with the IDENTITY collation in a Unicode database, regardless of the collation that you specified at database creation.

Details

Queries against non-catalog tables and views are not affected by this change.

Queries against the catalog tables or views might produce results in a different order than earlier versions of the DB2 database or when compared to queries against non-catalog tables and views.

Queries that combine data from catalog and non-catalog tables and views might produce results in a different order than earlier versions of DB2 or when compared to queries against non-catalog tables and views. Additionally, these queries may incur a noticeable performance degradation.

Resolution

To avoid performance degradation when joining catalog and non-catalog data in a query, define the non-catalog column as FOR BIT DATA.

Application development changes summary

Default JDBC driver has changed for Java routines

The default driver for Java routines such as Java stored procedures and user-defined functions is now the IBM Data Server Driver for JDBC and SQLJ.

Details

Prior to Version 9.5, if you wanted to use the IBM DB2 Driver for JDBC and SQLJ (renamed in Version 9.5 to the IBM Data Server Driver for JDBC and SQLJ) for Java routines, you needed to set the **DB2_USE_DB2JCCT2_JROUTINE** environment variable. The IBM Data Server Driver for JDBC and SQLJ is now the default driver, so this action is no longer necessary.

Resolution

If you want to use the deprecated DB2 JDBC Type 2 Driver for Linux, UNIX, and Windows to serve SQL requests for Java routines, set **DB2_USE_DB2JCCT2_JROUTINE** to OFF. However, you should consider migrating applications that use this deprecated driver to the IBM Data Server Driver for JDBC and SQLJ to avoid possible support problems in future releases.

> **Related concepts**
>
> "Specification of a driver for Java routines" in pureXML Guide
>
> "JDBC and SQLJ support has been enhanced" on page 74
>
> **Related reference**
>
> "Miscellaneous variables" in Data Servers, Databases, and Database Objects Guide

ResultSetMetaData returns different values for the IBM Data Server Driver for JDBC and SQLJ Version 4.0

For the IBM Data Server Driver for JDBC and SQLJ Version 4.0, the values that are returned for `ResultSetMetaData.getColumnName` and `ResultSetMetaData.getColumnLabel` are changed to conform to the JDBC 4.0 standard. Those values differ from the values that are returned for the IBM Data Server Driver for JDBC and SQLJ Version 3.50 and earlier JDBC drivers.

Details

The results vary, depending on the following factors:

- Which data source type and version you are using. DB2 for z/OS and OS/390 Version 7 and DB2 for i5/OS V5R2 are unaffected by this change. Only later versions of these database products and all versions of DB2 Database for Linux, UNIX, and Windows and IBM Informix Dynamic Server are affected.
- Whether a column in the SELECT list of a query has an AS clause. For JDBC drivers before the IBM Data Server Driver for JDBC and SQLJ Version 4.0, if a column in the SELECT list of a query contains an AS clause, ResultSetMetaData.getColumnName returns the argument of the AS clause. Under the IBM Data Server Driver for JDBC and SQLJ Version 4.0, ResultSetMetaData.getColumnName returns the table column name.
- Whether a column in the SELECT list of a query has a label from a LABEL statement but no AS clause. DB2 for z/OS and DB2 for System i support the LABEL statement, which assigns a label to a column. For JDBC drivers before the IBM Data Server Driver for JDBC and SQLJ Version 4.0, if a column in the SELECT list of a query has a label, ResultSetMetaData.getColumnName returns the table column name, and ResultSetMetaData.getColumnLabel returns the column label from the LABEL statement. Under the IBM Data Server Driver for JDBC and SQLJ Version 4.0, ResultSetMetaData.getColumnName and ResultSetMetaData.getColumnLabel return the table column name. The column label from the LABEL statement is not used.
- Whether a column in the SELECT list has a label from a LABEL statement and an AS clause. For JDBC drivers before the IBM Data Server Driver for JDBC and SQLJ Version 4.0, if a column in the SELECT list of a query has a label and an AS clause, ResultSetMetaData.getColumnName returns the argument of the AS clause, and ResultSetMetaData.getColumnLabel returns the column label from the LABEL statement. Under the IBM Data Server Driver for JDBC and SQLJ Version 4.0, ResultSetMetaData.getColumnName returns the table column name, and ResultSetMetaData.getColumnLabel returns the argument of the AS clause. The column label from the LABEL statement is not used.

Resolution

If you cannot change your applications to conform to the new ResultSetMetaData behavior but you need other features of JDBC 4.0, set the useJDBC4ColumnNameAndLabelSemantics Connection or DataSource property to DB2BaseDataSource.NO (2) to keep the old behavior.

Related reference

"Common IBM Data Server Driver for JDBC and SQLJ properties for DB2 for z/OS and DB2 Database for Linux, UNIX, and Windows" in Developing Java Applications

Batch updates with automatically generated keys cause an SQLException

With the IBM Data Server Driver for JDBC and SQLJ Version 3.50, or later, preparing an SQL statement for retrieval of automatically generated keys and using the PreparedStatement object for batch updates causes an SQLException.

Details

Versions of the IBM Data Server Driver for JDBC and SQLJ before Version 3.50 do not throw an SQLException when an application calls the addBatch or executeBatch method on a PreparedStatement object that is prepared to return automatically generated keys. However, the PreparedStatement object does not return automatically generated keys. IBM Data Server Driver for JDBC and SQLJ

Version 3.50 and later versions throw an SQLException to indicate that the application is attempting an incorrect operation.

Resolution

Modify applications so that they do not do batch updates on PreparedStatement objects that are prepared to return automatically generated keys.

Unfenced routines, unfenced wrapper libraries, and security plug-ins must be threadsafe (Linux and UNIX)

The new multithreaded database manager requires that unfenced routines (such as stored procedures and user-defined functions), unfenced wrapper libraries, and security plug-ins be threadsafe.

Details

Prior to Version 9.5, the agents which ran unfenced routines on Linux and UNIX systems were separate processes. The execution of unfenced stored procedures, unfenced user defined functions, unfenced wrapper libraries as well as security plug-ins that are not threadsafe in the multithreaded database manager could result in incorrect results, database corruption, or abnormal termination of the database manager.

Unfenced wrapper libraries that will be loading third party client libraries (such as Sybase and Teradata wrappers) must also be threadsafe. This applies to user-defined unfenced wrapper libraries, as the DB2-provided unfenced wrapper libraries are already threadsafe. Similarly, user-defined security plug-ins must be threadsafe.

Threadsafe refers to a specific property of code, such that if multiple operating system threads within the same process are all running the same piece of code concurrently, each thread will provide correct results, and will not adversely affect the operation of the other threads. Since the DB2 server is now multithreaded, the same unfenced stored procedure code may be executed by multiple database agents concurrently. Guaranteeing thread safety is a difficult task, and can only be done through code inspection. A few common examples of thread-unsafe code that you must avoid are:

*   The use of global variables that are not properly protected by some means of synchronization: for example, semaphores. A host variable in routine code is one example of a global variable.
*   Calls to library functions that are not threadsafe or that affect the entire process (as opposed to only the calling thread). Examples of this include library functions that change the current working directory or change the locale of the process.
*   Installation of signal handlers or changes to signal masks. The DB2 server installs its own signal handlers, and to guarantee the integrity of the DB2 server, these signal handlers must not be altered.
*   Creation of new threads or processes.

Resolution

If you are in doubt about the thread-safety of code or if access to the source code is unavailable, catalog your routines or wrapper libraries as FENCED and NOT THREADSAFE. Run them unfenced only if you have carefully inspected them to

ensure that they are threadsafe and if the performance of running that code in fenced mode is not acceptable.

Identifier length limits have increased

Large identifier support lets you more easily port applications from other DBMS vendors. You will also find it easier to migrate data definition language (DDL) because you no longer need to shorten identifiers.

Identifiers with longer maximum length are listed in the following table:

Table 25. Identifier length limits in Version 9.1 and 9.5

Identifier Name	Length in Version 9.1 (bytes)	Length in Version 9.5 (bytes)
Attribute	18	128
Authorization ID (Authid)	30	128
Column	30	128
Constraint	18	128
Cursor	18	128
Database partition group	18	128
Event monitor	18	128
Group	30	128
Package	8	128
Schema	30	128
Specific name	18	128
SQL path (specified by the FUNCPATH BIND option and CURRENT PATH special register)	254	2048
Statement	18	128
Trigger	18	128
User-defined type	18	128

Note that the 128-byte limit applies only to non-embedded SQL only because the SQLDA is still limited to 8-byte schema names for user-defined types (UDTs), 18-byte names for UDTs, and 30-byte names for columns.

The 128-byte limit refers to the limit that is stored in the system catalog by the database manager. Because the code page that is used to represent an identifier in an application can vary, the limit on the application side is undefined. DB2 utilities on both the application and the server side use a limit of 128 bytes regardless of the application code page.

You can find a useful sample file, checkv9limits in samples/admin_scripts. You can use this file to find identifiers in a database that might be using the larger Version 9.5 limits.

Related reference

"SQL and XML limits" in Data Servers, Databases, and Database Objects Guide

Columns and application buffers require larger default values

The USER and SCHEMA special registers now return values up to 128 bytes long. This support for longer identifiers requires you to increase the default length values for columns and application buffers to 128 bytes.

Details

When you create or alter a table using the WITH DEFAULT option and specify a user special register (CURRENT USER, SESSION_USER, SYSTEM_USER) or CURRENT SCHEMA, a warning is returned if the target column is too small as shown in the following example:

```
SQL20114W  Column "COL1" in table "TAB1" is not long enough
for the defined length of the USER default value.  SQLSTATE=01642
```

Resolution

Regardless of whether you use longer identifier names, you must modify default length values for columns and application buffers that might be stored in the USER or SCHEMA special register values to accept the new maximum length.

Related concepts

"Identifier length limits have increased" on page 67

Some CLI/ODBC applications can consume more memory

CLI/ODBC clients that set **BlockLobs** to 1 and bind the LOB values directly to buffers can consume more memory than they did in previous releases.

The amount of additional memory that a CLI/ODBC application consumes depends on the amount of data that it retrieves for a request. CLI/ODBC clients can specify the **MaxLOBBlockSize** configuration keyword to limit the amount of LOB data returned for a single request. Alternatively, you can set the connection attribute **SQL_ATTR_MAX_LOB_BLOCK_SIZE** or the DB2 registry variable **DB2_MAX_LOB_BLOCK_SIZE**.

Related concepts

"db2cli.ini initialization file" in Call Level Interface Guide and Reference, Volume 2

Related reference

"Miscellaneous variables" in Data Servers, Databases, and Database Objects Guide

"Statement attributes (CLI) list" in Call Level Interface Guide and Reference, Volume 2

"BlockLobs CLI/ODBC configuration keyword" in Call Level Interface Guide and Reference, Volume 2

"MaxLOBBlockSize CLI/ODBC configuration keyword" in Call Level Interface Guide and Reference, Volume 2

db2Load and db2Import parameters have been changed to support longer identifiers

To support longer names, a new input action string, **piLongActionString**, has been added for the db2Load and db2Import APIs. It uses the sqllob instead of the sqlchar data structure.

Details

The **piActionString** data structure is deprecated and might be removed in a future release. You should use the new data structure **piLongActionString** instead.

Resolution

The APIs check whether you initialized only one of the data structures. If you initialized both, message SQL3009N is returned, indicating that the data structures are mutually exclusive.

> **Related reference**
>
> ″db2Import API - Import data into a table, hierarchy, nickname or view″ in Data Movement Utilities Guide and Reference
>
> ″db2Load API - Load data into a table″ in Data Movement Utilities Guide and Reference

Identifiers that are too long result in errors and warnings being returned earlier

In Version 9.5, additional boundary and length checking takes place for identifiers. If identifiers exceed the limits, errors or warnings might be generated earlier in the precompilation, binding, or execution of an application than in previous DB2 product releases.

Details

For example, a SQL0102N precompile error is now returned for an embedded SQL statement that contains a GRANT statement for an AUTHID longer than 128 bytes. In previous versions of the DB2 product, an AUTHID longer than 128 bytes would generate an error when the GRANT statement was executed.

Resolution

Correct the identifier name to use an allowable length.

> **Related concepts**
>
> "Identifier length limits have increased" on page 67

Downlevel utilities and APIs might not handle longer identifiers correctly

Utilities and APIs shipped in previous DB2 versions might be unable to accommodate the longer identifiers in Version 9.5.

Details

When a utility or API processes longer identifier names, possible outcomes in how the longer format of the data will be handled are as follows:

- Everything works correctly.
- The utility or API returns a warning or error message referring to the longer data.
- The utility or API returns a warning or error message and fails.
- The longer data is truncated without a message.

Resolution

If your Version 9.5 databases contain large identifiers, use only Version 9.5 clients and utilities to access these databases. If a utility needs to access large identifiers, use only the Version 9.5 level of that utility.

> **Related concepts**
> "Identifier length limits have increased" on page 67

Unqualified SYSFUN functions might return SYSIBM error messages

Some SYSFUN functions are now available as SYSIBM functions. When an error occurs, the SYSIBM version returns different SQLCODES than the SYSFUN version.

Details

The following eight SYSFUN functions are also now available as built-in functions in the SYSIBM schema: LN (or LOG), LOG10, DEGREES, RADIANS, SIGN, SQRT, POWER, and EXP. An unqualified reference to any of these eight functions resolves to the SYSIBM schema, and consequently, a different SQLCODE than expected might be returned.

For instance, if you make an unqualified function call such as `values (sqrt(-1))` and an error occurs, you receive an SQLCODE similar to this one:

```
1
-----------------------
SQL0802N  Arithmetic overflow or other arithmetic exception occurred.
SQLSTATE=22003
```

A fully qualified function call that explicitly calls the SYSFUN version, such as `values (sysfun.sqrt(-1)` returns a different kind of SQLCODE:

```
1
-----------------------
SQL0443N  Routine "SYSFUN.SQRT" (specific name "SQRT") has returned an error SQLSTATE
with diagnostic text "SYSFUN:01".
SQLSTATE=38552
```

Different error codes are returned because SQL0443N is specific to user-defined functions and the SYSIBM versions of the functions are now implemented as built-in functions. Note that the error code of the unqualified call does not contain information about the function that failed or the type of failure.

Resolution

To ensure that you are calling the SYSFUN version of these functions, always fully qualify the function call.

Special registers are longer

The CURRENT DEFAULT TRANSFORM GROUP and CURRENT PATH special registers are longer in Version 9.5.

Details

The length of the CURRENT DEFAULT TRANSFORM GROUP special register has increased from 18 to 128 bytes. The CURRENT PATH special register has increased in length from 254 to 2048 bytes. If these increased special register lengths are

assigned to application buffers or columns that cannot accommodate the length, an error is returned.

Resolution

Increase the lengths of columns or application buffers if they cannot accommodate the values of the special registers that you assign to them.

Related concepts

"Identifier length limits have increased" on page 67

CLP and system command changes summary

Command Line Processor (CLP) output has changed

The output of some CLP commands has changed to show different information and to accommodate the display of larger identifiers (128 bytes and, in the case of SQL Path, 2048 bytes). The changed output might affect applications that parse and depend on the output formatting of CLP command output.

Details

In cases where the commands have a **SHOW DETAIL** option but you did not specify it, the identifier fields are truncated to the current length, and a > character appears in the last position of the name. This is the convention that CLP uses to show that a field was truncated. If you specified the **SHOW DETAIL** option, the full name is shown. In cases where there is no **SHOW DETAIL** option, the full length is displayed.

The outputs of the following commands have changed as follows:

- DESCRIBE: The output of the command no longer shows the SQLDA fields and uses more generic terms such as `Column name` instead of `sqlname`. In addition, the **TABLE** command parameter now returns information about implicitly hidden columns, and the **OUTPUT** command parameter now returns information about an implicitly hidden column only if you specify the column in the SELECT list of the described query.
- GET DB CFG: The output does not show deprecated database configuration parameters.
- GET SNAPSHOT: The command output is slightly different due to changes in the DB2 application memory model.

Resolution

You might need to update applications that parse and depend on the output formatting of the CLP commands.

Related concepts

"Identifier length limits have increased" on page 67

"Some database configuration parameters have changed" on page 139

"Some database manager configuration parameters have changed" on page 123

Backup operation simultaneously backs up multiple database partitions

You can now back up multiple database partitions at once with a single call to the BACKUP DATABASE command, the db2Backup API, or the ADMIN_CMD

procedure with the BACKUP DATABASE parameter. Changes to the backup utilities to support this new feature can impact existing applications or scripts.

Details

In Version 9.1, to back up a partitioned database, you have to invoke the backup utility on each database partition. In Version 9.5, you can back up multiple database partitions at once by performing a single system view (SSV) backup on the catalog database partition. When you perform a backup operation from the catalog database partition, you can use the **ON DBPARTITIONNUMS** parameter or the **iAllNodeFlag** option to specify which partitions to include in the backup. The specified partitions will be backed up simultaneously, and the backup timestamp associated with all specified partitions will be the same.

As a result of this feature, IBM Tivoli Storage Manager (TSM) return codes 41 (`exceeded mount points`) and -51 (`connection attempt timed out`) are no longer fatal errors, except when there are no more sessions available.

Resolution

If you are using TSM, modify your applications or scripts to handle the return code changes.

Related concepts

"Backup overview" in Data Recovery and High Availability Guide and Reference

Related tasks

"Using backup" in Data Recovery and High Availability Guide and Reference

"Backing up partitioned databases" in Data Recovery and High Availability Guide and Reference

Related reference

"BACKUP DATABASE command" in Command Reference

"db2Backup API - Back up a database or table space" in Administrative API Reference

"BACKUP DATABASE command using the ADMIN_CMD procedure" in Administrative Routines and Views

db2audit command has changed

Due to new features provided for the audit facility in Version 9.5, some aspects of the db2audit command have changed.

Details

In Version 9.5, the audit facility provides the ability to audit at both the instance and the individual database level, independently recording all instance- and database-level activities with separate logs for each level. The system administrator (who holds SYSADM authority) can use the db2audit tool to configure an audit at the *instance* level and to control when information for that audit is collected. The system administrator can also use the db2audit tool to archive both instance and database audit logs and to extract audit data from archived logs of either type.

The security administrator (who holds SECADM authority) can use audit policies with the SQL statement AUDIT to configure and control the audit requirements for an individual database. The security administrator can use the

SYSPROC.AUDIT_ARCHIVE and SYSPROC.AUDIT_DELIM_EXTRACT stored procedures and the SYSPROC.AUDIT_LIST_LOGS table function to archive audit logs, locate logs of interest, and extract data into delimited files for analysis.

The following command parameters of the db2audit command have changed to accommodate this new functionality:

- The **prune** command parameter has been removed.

  In previous releases, you first extracted the audit data to a delimited ASCII file, then loaded it into tables. Then you could run db2audit with the **prune** command parameter to clean up the audit log. Instead, in Version 9.5, you should archive the audit logs on a regular basis (such as once a day or week), and after you have extracted the data that you need from the archived files, you can delete them or store them offline.

- The syntax for the **configure** command parameter has changed.

  In Version 9.5, you can specify success or failure for each audit category; you no longer need to specify success for all categories or failure for all categories. Additionally, only the categories that you specify on the command are modified; all others are unchanged. In previous releases, if you did not specify a category, it was set to false: that is, not audited.

  You can use the **configure** command parameter only for instance-level audits, not database-level audits. For database-level audits, the security administrator can use audit policies to configure auditing.

- The syntax for the **extract** command parameter has changed.

  Because the audit log for each database is now stored in a separate file, the **database** parameter has been removed. Also, you must supply an archived audit log file name.

  The **extract** command parameter no longer causes an instance to hang until the extract operation is completed because it now uses an archived audit log file instead of the current db2audit.log file. You must archive the audit log before using the **extract** command parameter. It is also unnecessary to run an extract as often as in previous releases. In Version 9.5, you need to run it only when you want to view audit data.

  The **extract** command parameter now allows you to specify which categories to extract and whether to extract success or failure events (or both).

- The items displayed by the **describe** command parameter have changed to support the new syntax of the **configure** command parameter.

  In previous releases, you could specify only the status as SUCCESS or FAILURE for all categories indicated in the scope. Now, you can specify a status with the values SUCCESS, FAILURE, NONE, or BOTH for each category. The following table maps the values of category event, log errors, and log success of the previous release, displayed in the standard output, to the Version 9.5 values displayed in the standard output:

Table 26. Mapping of previous release values in the standard output to Version 9.5 values

Category event, Log errors, Log success (Previous release)	Category event (Version 9.5)
TRUE, FALSE, FALSE	NONE
TRUE, FALSE, TRUE	SUCCESS
TRUE, TRUE, FALSE	FAILURE
TRUE, TRUE, TRUE	BOTH
FALSE, Any values, Any values	NONE

- The **start** and **stop** command parameters affect only instance-level audits, not database-level audits.

Resolution

Use the new syntax of the db2audit command.

> **Related concepts**
>
> "Audit policies" in Database Security Guide
>
> "Storage and analysis of audit logs" in Database Security Guide
>
> **Related reference**
>
> "db2audit - Audit facility administrator tool command" in Command Reference

db2ckmig command has changed

The db2ckmig command now checks whether a database is in the restore pending state and whether you have external unfenced routines on the Linux and UNIX operating platforms that do not depend on the DB2 engine library in your database.

Details

You can use the db2ckmig command to verify that a database can be migrated. This command now fails if the database is in the restore pending state. Refer to "Verifying that your databases are ready for migration" in *Migration Guide* for a full description of the usage and output of the command.

If you have external unfenced routines on Linux and UNIX operating systems that do not depend on the DB2 engine library in your database, this command now returns the warning message SQL1349W and generates a file with a list of all of the external unfenced routines that are redefined as FENCED and NOT THREADSAFE when you migrate the database.

Resolution

To remove a database from the restore pending state, you must perform a database restore operation.

If you receive the SQL1349W warning message and you can run your external routines as FENCED and NOT THREADSAFE, you can proceed to migrate your database. If you need to run your external routines as NOT FENCED and THREADSAFE in your migrated database, you must verify that they can safely run as NOT FENCED and THREADSAFE before you migrate your database. Refer to "Migrating 32-bit external routines to run on 64-bit instances" in *Migration Guide* for information on how to do this verification.

> **Related tasks**
>
> "Using restore" in Data Recovery and High Availability Guide and Reference
>
> "Migrating routines" in Migration Guide

db2mtrk command has changed

The db2mtrk command, which provides a complete report of memory status, is changed. The **-p** option (which lists private agent memory heaps) is deprecated and has been replaced with the **-a** option (which lists all application memory consumption).

Details

The command output is different due to changes in the DB2 application memory model and shows more information.

Resolution

If you have scripts that parse output of the db2mtrk command, modify the parsing conditions to account for the new format.

> **Related reference**
>
> "db2mtrk - Memory tracker command" in Command Reference

Customized callout scripts are searched for (Linux and UNIX)

The database manager now checks for customized versions of the db2cos, db2cos_datacorruption, db2cos_hang, and db2cos_trap scripts. If none exists, a default version is used.

Details

The db2cos, db2cos_datacorruption, db2cos_hang, and db2cos_trap scripts are run to collect troubleshooting information when a trap, hang, or data corruption outage happens.

On Linux and UNIX operating systems, the database manager now checks first to see whether customized versions of the callout scripts exist in `INSTHOME/sqllib/adm/`, where `INSTHOME` is the instance home directory, and runs these scripts. If no scripts are found, the database manager runs the system-provided scripts in the `INSTHOME/sqllib/bin/` directory.

Resolution

Do not change the system-provided settings and do not modify the system-provided scripts.

> **Related concepts**
>
> "db2cos (callout script) output files" in Troubleshooting Guide

Listing of OS processes and threads has changed (Linux and UNIX)

Due to the move to the multithreaded architecture in Version 9.5, the output for the ps command has changed. Also, the db2pd command now has a new **-edus** option that lists all engine dispatchable units (EDUs) for a database partition.

Details

In Version 9.5, on UNIX and Linux operating systems, almost all operating system processes in an DB2 instance are operating system threads, all within a single process for the instance. This reduces the number of DB2 operating system processes and makes it easier to troubleshoot problems with your systems.

When invoking the ps command with the **-fu** *instancename* option, the output that displays lists only two DB2 processes, db2sysc and db2acd as shown in the example.

Resolution

To display the individual threads associated with the db2sysc process, you must use the applicable thread options on the ps command. On the Linux operating system, for example, you can use the **-lLfp** option. On the AIX operating system, you can use the **-m -o THREAD** options.

Example

The ps **-fu** command now displays only two processes, as shown in the following example:

```
$ 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      27688 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
```

To get the details of process ID 27688, invoke the ps command with the new **-lLfp** option, as shown in the following example:

```
$ps -lLfp 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
```

The following example shows the information that is provided with the **-edus** option:

```
$ 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         db21fr (TESTDB)
52          183303661920 30291         db2loggw (TESTDB)
51          183316244832 30290         db2loggr (TESTDB)
```

```
50        183257524576  28156        db2evmli (DB2DETAILDEADLOCK)
49        183261718880  28153        db2taskd (TESTDB)
46        183274301792  28150        db2wlmd (TESTDB)
26        183312050528  27943        db2stmm (TESTDB)
17        183324633440  27827        db2agent (TESTDB)
16        183328827744  27714        db2resync
15        183333022048  27697        db2ipccm
14        183337216352  27696        db2licc
13        183341410656  27695        db2thcln
12        183345604960  27694        db2alarm
1         183085558112  27688        db2sysc
```

Related concepts

"Simplified multithreaded architecture reduces the Total Cost of Ownership (TCO)" on page 28

Related reference

"db2pd - Monitor and troubleshoot DB2 database command" in Command Reference

Chapter 17. Deprecated functionality

Functionality gets marked as *deprecated* when a specific function or feature is supported in the current release but might removed in a future release. In some cases, it might be advisable to plan to discontinue the use of deprecated functionality.

For example, a registry variable might be deprecated in this release because the behavior triggered by the registry variable has been enabled by default in this release, and the obsolete registry variable will be removed in a future release.

Review this section to find out more details about Version 9.5 deprecated functionality and to plan for future changes.

Some registry and environment variables are deprecated

A number of registry and environment variables are deprecated in Version 9.5. These variables are still available, but you should not use them because they will likely be removed in future versions of the product.

The following table lists deprecated registry and environment variables. They have been replaced by another feature, or the function that they support is obsolete.

Table 27. Registry and environment variables deprecated in Version 9.5

Registry or environment variable	Details
DB2_ALLOCATION_SIZE	This variable is deprecated and might be removed in a later release.
DB2ATLD_PORTS	This variable is deprecated and might be removed in a later release.
DB2_ASYNC_IO_MAXFILOP	This variable is obsolete because of the shared file handle table maintained by the threaded database manager . It can still be set in Version 9.5, but it will have no effect.
DB2_BAR_AUTONOMIC_DISABLE	This variable is needed for IBM internal use only (for example, testing).
DB2BPVARS	This variable is deprecated and may be removed in a later release.
DB2COUNTRY	This variable is replaced with the **DB2TERRITORY** registry variable. **DB2TERRITORY** allows you to specify the region or territory code of a client application, which influences the date and time formats. **DB2TERRITORY** accepts the same values as **DB2COUNTRY**: for example, setting **DB2COUNTRY** to 68 is equivalent to setting **DB2TERRITORY** to 68.
DB2DEFPREP	Use this variable only at the recommendation of IBM Service.

Table 27. Registry and environment variables deprecated in Version 9.5 (continued)

Registry or environment variable	Details
DB2DMNBCKCTLR	This variable is no longer necessary because backup domain controllers in the Active Directory are only on the Windows NT platform, not on the Windows 2003 and Windows XP platforms. Version 9.5 does not run on the Windows NT platform.
DB2_ENABLE_SINGLE_NIS_GROUP	This variable is deprecated because support for the Network Information Services (NIS and NIS+) will be removed in a future release. For more information, see Network Information Services support is deprecated (Linux and UNIX).
DB2FFDC	This variable is replaced with the **DB2FODC** registry variable. The same functionality that **DB2FFDC** provided is available if you use the DUMPCORE option of **DB2FODC**. By default, the DUMPCORE option is set to ON to enable core file generation and to maintain compatibility with previous releases.
DB2_HASH_JOIN	This variable, created to provide control of a DB2 feature, is unnecessary as this registry control is no longer required.
DB2_INDEX_FREE	This variable has the same functionality as the PCTFREE clause in a CREATE INDEX statement. The PCTFREE clause specifies what percentage of each index page to leave as free space when building the index. For example, the equivalent to setting **DB2_INDEX_FREE** to 20 is CREATE INDEX *IndexName* ON *TableName* (*Columns*) PCTFREE 20. The PCTFREE value only comes into effect at index build or index rebuild time and remains the same for the lifetime of the index. The PCTFREE clause only affects the index that is being created, unlike **DB2_INDEX_FREE**, which affects all indexes.
DB2_MAP_XML_AS_CLOB_FOR_DLC	This variable is deprecated because most existing DB2 applications that access XML values do so with an XML-capable client (Version 9.1 and newer). You only need this variable for previous applications that generically fetched table data and couldn't parse UTF-8 XML data in a BLOB.
DB2MEMMAXFREE	This variable is no longer necessary because the database manager now uses a threaded engine model. For more information, see Multi-threaded architecture reduces total cost of ownership.

Note: Do not set this variable. Doing so will likely degrade performance and might cause unexpected behavior. |

Table 27. Registry and environment variables deprecated in Version 9.5 (continued)

Registry or environment variable	Details
DB2_NO_FORK_CHECK	This variable is no longer necessary because the procedure for getting the current process id (PID) is improved in Version 9.5.
DB2NTNOCACHE	This variable has been deprecated since DB2 Universal Database (DB2 UDB) Version 8.2. You can achieve everything that this registry variable was designed to do by using the CREATE TABLESPACE and ALTER TABLESPACE SQL statements.
DB2_PARTITIONEDLOAD_DEFAULT	This variable is deprecated because the LOAD command has various options that can be used to achieve the same behavior.
DB2PRIORITIES, DB2NTPRICLASS	These variables are deprecated. Use DB2 service classes to adjust agent priority and prefetch priority.
DB2ROUTINE_DEBUG	This variable is no longer necessary because this stored procedure debugger has been replaced by the Unified Debugger.
DB2_RR_TO_RS	Use this variable only at the recommendation of IBM Service.
DB2_SNAPSHOT_NOAUTH	This variable is not needed because you can achieve the same functionality by using the SYSMON authority group.
DB2_TRUSTED_BINDIN	This variable is deprecated because it is no longer useful.
DB2_UPDATE_PART_KEY	This variable is deprecated and may be removed in a later release. It is obsolete because partitioning key updates are permitted by default.
DB2_VENDOR_INI	This variable is no longer necessary because you can put the environment variable settings that it contains into the file specified by the **DB2_DJ_INI** variable.
DB2YIELD	This variable was only used on Windows 3.1, which Version 9.5 does not support.

Related concepts

"Some registry and environment variables have changed" on page 126

"Some registry and environment variables are discontinued" on page 175

GET AUTHORIZATIONS command is deprecated

The GET AUTHORIZATIONS command is deprecated in Version 9.5. Instead, use the AUTH_LIST_AUTHORITIES_FOR_AUTHID table function.

Details

The GET AUTHORIZATIONS command reports the authorities of the current user from values found in the database configuration file and the authorization system catalog view (SYSCAT.DBAUTH). Due to internal changes in the DB2 authorization model, it is deprecated in Version 9.5.

The command reports authorities held through roles as direct or indirect, based on to whom the role is granted: the user or the group.

Resolution

Use the AUTH_LIST_AUTHORITIES_FOR_AUTHID table function to return the authorities for a particular user.

Related reference

"AUTH_LIST_AUTHORITIES_FOR_AUTHID table function" in Administrative Routines and Views

sqluadau API is deprecated

The sqluadau API is deprecated in Version 9.5. Instead, use the AUTH_LIST_AUTHORITIES_FOR_AUTHID table function.

Details

Due to internal changes in the DB2 authorization model, the sqluadau API is deprecated. It returns the current user's authorities. It reports authorities held through roles as direct or indirect, based on to whom the role is granted.

Resolution

Use the AUTH_LIST_AUTHORITIES_FOR_AUTHID table function to get the information that sqluadau provides.

Related reference

"AUTH_LIST_AUTHORITIES_FOR_AUTHID table function" in Administrative Routines and Views

Some monitor elements are deprecated

A subset of monitor elements is deprecated to reflect changes in Version 9.5 functionality.

Details

The following monitor elements are no longer recommended and might be removed in a future release:

- **agents_waiting_top** - Maximum Number of Agents Waiting
- **agents_waiting_on_token** - Agents Waiting for a Token
- **authority_lvl** - User Authorization Level
- **cat_cache_size_top** - Catalog Cache High Water Mark
- **db_heap_top** - Maximum Database Heap Allocated
- **max_agents_overflows** - Maximum Agent Overflows
- **pkg_cache_size_top** - Package Cache High Water Mark
- **priv_workspace_num_overflows** - Private Workspace Overflows
- **priv_workspace_section_inserts** - Private Workspace Section Inserts
- **priv_workspace_section_lookups** - Private Workspace Section Lookups
- **priv_workspace_size_top** - Maximum Private Workspace Size
- **shr_workspace_num_overflows**- Shared Workspace Overflows
- **shr_workspace_section_inserts** - Shared Workspace Section Inserts

- **shr_workspace_section_lookups** - Shared Workspace Section Lookups
- **shr_workspace_size_top** - Maximum Shared Workspace Size

Resolution

The deprecated elements might have been removed from the snapshot output and might be defined in the SNAPDBM administrative view and SNAP_GET_DBM table function. If the element is found, its value will not be valid. The db2GetSnapshot API does not return values for these deprecated elements for requests using an iVersion (Version ID of the database monitor data to collect) of SQLM_DBMON_VERSION6 or later, but for SQLM_DBMON_VERSION5_2 or earlier a value of zero is returned. Use the replacement element monitors when available.

Deprecated monitor element	Change
• **agents_waiting_top** - Maximum Number of Agents Waiting • **agents_waiting_on_token** - Agents Waiting for a Token • **max_agents_overflows** - Maximum Agent Overflows	These monitor elements are no longer needed because the configuration mechanism for process model parameters has been simplified in Version 9.5. Using them will not generate an error. However, they do not return a valid value.
authority_lvl - User Authorization Level	Use the **authority_bitmap** monitor element instead, which shows the authorities and privileges granted to a user and to the groups to which the user belongs. These authorities and privileges include the ones granted to roles that are granted to the user and to the groups to which the user belongs.
cat_cache_size_top - Catalog Cache High Water Mark	Using this monitor element will not generate an error. However, it does not return a valid value. Use the memory pool element **pool_watermark** and the **pool_id** element with a value of SQLM_HEAP_CAT_CACHE.
db_heap_top - Maximum Database Heap Allocated	Using this monitor element will not generate an error. However, it does not return a valid value. Use the memory pool element **pool_watermark** and the **pool_id** element with a value of SQLM_HEAP_DATABASE.
pkg_cache_size_top - Package Cache High Water Mark	Using this monitor element will not generate an error. However, it does not return a valid value. Use the memory pool element **pool_watermark** and the **pool_id** element with a value of SQLM_HEAP_PACKAGE_CACHE.

Deprecated monitor element	Change
• **priv_workspace_num_overflows** - Private Workspace Overflows • **priv_workspace_section_inserts** - Private Workspace Section Inserts • **priv_workspace_section_lookups** - Private Workspace Section Lookups • **priv_workspace_size_top** - Maximum Private Workspace Size • **shr_workspace_num_overflows**- Shared Workspace Overflows • **shr_workspace_section_inserts** - Shared Workspace Section Inserts • **shr_workspace_section_lookups** - Shared Workspace Section Lookups • **shr_workspace_size_top** - Maximum Shared Workspace Size	These monitor elements are no longer needed because the memory configuration has been simplified in Version 9.5. Using them will not generate an error. However, they do not return a valid value.

Related concepts

"Simplified multithreaded architecture reduces the Total Cost of Ownership (TCO)" on page 28

Related reference

"pool_id - Memory Pool Identifier monitor element" in System Monitor Guide and Reference

"pool_watermark - Memory Pool Watermark monitor element" in System Monitor Guide and Reference

"authority_bitmap - User Authorization Level monitor element" in System Monitor Guide and Reference

Log control file SQLOGCTL.LFH has been renamed and copied

In Version 9.1, the database manager maintained one log control file: SQLOGCTL.LFH. In Version 9.5, the database manager maintains two copies of the log control file: SQLOGCTL.LFH.1 and SQLOGCTL.LFH.2.

Details

When a database restarts after a failure, the database manager applies transaction information stored in log files to return the database to a consistent state. The database manager uses a log control file to determine which entries in the log files need to be applied.

If the log control file is damaged, it might not be possible for the database manager to return the database to a consistent state. However, having two copies of the log control file can make database recovery more resilient because if one copy of the log control file is damaged, the database manager can use the other copy during a restart.

Resolution

If you run applications or scripts that refer to the log control file, update those applications and scripts to refer to one of the duplicate log control files now. For

tools like db2flsn, use the **-path** parameter to specify the path to both log control files. This enables the database manager to handle the case where one of the log control files is missing, damaged, or stale.

Example

If you use the db2flsn command with the **-file** parameter, pass the name of either one of the log control files, SQLOGCTL.LFH.1 or SQLOGCTL.LFH.2, to the command. See the topic "db2flsn - Find Log Sequence Number" in *Command Reference* for more information.

> **Related concepts**
>
> "Log control files" in Data Recovery and High Availability Guide and Reference
>
> **Related reference**
>
> "db2flsn - Find log sequence number command" in Command Reference

IMPORT command options CREATE and REPLACE_CREATE are deprecated

The **CREATE** and **REPLACE_CREATE** options of the IMPORT command are deprecated and might be removed in a future release.

Details

The **CREATE** and **REPLACE_CREATE** options allow you to create a table and insert data into it. However, these options are deprecated because not all of the properties of the table are re-created when you use **CREATE** and **REPLACE_CREATE**.

Resolution

Instead of using the **CREATE** and **REPLACE_CREATE** options, use the db2look command in a two-step process. First, use db2look to capture the original table definitions and re-create the table. Then, after re-creating the table, issue a LOAD or an IMPORT command to add the data to the table. The db2look command preserves all of the properties of a table and, when followed by a separate IMPORT or LOAD operation, provides a superior option for table re-creation.

> **Related tasks**
>
> "Creating tables like existing tables" in Data Servers, Databases, and Database Objects Guide
>
> **Related reference**
>
> "db2look - DB2 statistics and DDL extraction tool command" in Data Movement Utilities Guide and Reference

XML Extender is deprecated

As of Version 9.5, the functions provided by XML Extender have been replaced by the pureXML feature. As a result, XML Extender is deprecated.

Details

With the introduction of the pureXML feature in DB2 Version 9.1, XML Extender is deprecated.

The pureXML feature allows you to store well-formed XML documents in database table columns that have the XML data type. By storing XML data in XML columns, the data is kept in its native hierarchical form, rather than stored as text or mapped to a different data model. Database functions such as XMLQUERY and XSLTRANSFORM can be applied directly to database tables having the XML data type. Because the database provides a comprehensive set of XML tools, XML Extender functions are no longer required.

Static data stream snapshot output is deprecated

When an input version of Version 5.2 (or earlier) is specified to the snapshot monitor API, the snapshot output is passed back in static sized structures whose descriptions are found in the sqlmon.h file. This snapshot output format is deprecated and might be removed in a future release.

Details

The following input versions are deprecated and support for them might be removed in a future release:

- SQLM_DBMON_VERSION1
- SQLM_DBMON_VERSION2
- SQLM_DBMON_VERSION5
- SQLM_DBMON_VERSION5_2

Version 6 and later snapshot monitors use a self-describing data stream rather than static structures.

Resolution

Change any monitoring applications that use the deprecated input versions to use a newer version and change them to use the self-describing snapshot monitor format. For examples of monitoring applications that use the self-describing monitor format, see dbsnap.c (C sample) or dbsnap.C (C++ sample).

Web Object Runtime Framework (WORF) is deprecated

The Web Objects Runtime Framework (WORF) is deprecated and might be removed in a future release. Starting with this release, IBM Data Studio provides a more simple and more intuitive development environment for quickly developing and deploying Web services.

Details

The Web Objects Runtime Framework (WORF) provides the tooling and runtime support for creating and invoking DADX documents as Web services.

WORF is now replaced with a new feature within IBM Data Studio that allows you to create Web services without writing document access definition extension (DADX) files. Moreover, you can use the IBM Data Studio to create the SQL statements and stored procedures that you want to base the operations of your Web services on. Finally, in many scenarios, deploying a Web service requires a single mouse click.

You can read more information about this feature in the Developing and deploying Web services topic in the IBM Data Studio Information Center located at

http://publib.boulder.ibm.com/infocenter/dstudio/v1r1m0.

Resolution

You should migrate your WORF Web services to IBM Data Studio Web services. Instructions for migrating are in the Migrating Web applications that were developed for the Web Object Runtime Framework (WORF) topic in the IBM Data Studio Information Center located at http://publib.boulder.ibm.com/infocenter/dstudio/v1r1m0.

piActionString data structure of the db2Import and db2Load API is deprecated

The **piActionString** data structure of the db2Import and db2Load APIs is deprecated and might be removed in a future release.

Details

To support longer names, a new input action string, **piLongActionString**, has been added for the db2Load and db2Import APIs. It uses the sqllob instead of the sqlchar data structure.

Resolution

The APIs check whether you initialized only one of the data structures. If you initialized both, message SQL3009N is returned, indicating that the data structures are mutually exclusive.

> **Related reference**
>
> "db2Import API - Import data into a table, hierarchy, nickname or view" in Data Movement Utilities Guide and Reference
>
> "db2Load API - Load data into a table" in Data Movement Utilities Guide and Reference

Network Information Services support is deprecated (Linux and UNIX)

Support for the Network Information Services (NIS) and Network Information Services Plus (NIS+) features are deprecated.

Details

NIS and NIS+ support for user authentication is deprecated on Linux and UNIX operating systems. As a result, the **DB2_ENABLE_SINGLE_NIS_GROUP** registry variable is deprecated and might be removed in a future release.

Resolution

Lightweight Directory Access Protocol (LDAP) is the recommended solution for centralized user-management services. Version 9.5 supports LDAP based authentication and group lookup functionality through the use of LDAP security plug-in modules.

For information on how to use the NIS and NIS+ features in DB2 environments, see the Version 9.1 Information Center.

> **Related concepts**

"LDAP-based authentication and group lookup support" in Database Security Guide

Chapter 18. Discontinued functionality

Discontinued functionality is functionality that is no longer available. You need to make changes if you were using that functionality in previous releases.

Review this section to find out more details about Version 9.5 discontinued functionality.

Extended storage (ESTORE) feature support is discontinued

Extended storage (ESTORE) feature support is discontinued.

Details

The Extended Storage option for buffer pools was discontinued in Version 9.1. In Version 9.5, the associated configuration parameters, monitor elements, and other interfaces to ESTORE are removed. The predominance of computers with 64-bit operating systems removes the need for the ESTORE feature because these computers do not have the memory limitation of 32-bit operating systems.

The discontinuation of ESTORE affects the performance of all 32-bit operating systems where it was supported and impacts DB2 functionality as follows:

- The **ESTORE_SEG_SZ** and **NUM_ESTORE_SEGS** configuration parameters are discontinued.
- The following ESTORE monitor elements are discontinued and are no longer displayed in snapshots or event monitor output: **pool_index_to_estore**, **pool_data_to_estore**, **pool_index_from_estore**, and **pool_data_from_estore**.
- The following deprecated table functions return a NULL value for the discontinued monitor elements: SNAP_GET_TBSP, SNAP_GET_DB, SNAPSHOT_BP, SNAPSHOT_TBS, SNAPSHOT_DATABASE, and SNAPSHOT_APPL.
- The ALTER BUFFERPOOL and CREATE BUFFERPOOL statements no longer accept the EXTENDED STORAGE and NOT EXTENDED STORAGE options.

Resolution

To allocate more memory, you need to upgrade to 64-bit hardware, operating system, and DB2 products. You should also modify applications and scripts to remove references to the discontinued functionality.

Address Windowing Extensions (AWE) feature support is discontinued (Windows)

Support for the Address Windowing Extensions (AWE) feature is discontinued.

Details

Earlier DB2 versions supported the AWE feature, a set of memory management extensions that allowed applications running on 32-bit DB2 servers to manipulate memory above certain limits. For instance, by setting the **DB2_AWE** registry

variable, you were able to allocate buffer pools that used up to 64 GB of memory. However, with the predominance of 64-bit platforms, the need for the AWE feature has been reduced.

The discontinuation of AWE impacts performance on 32-bit Windows platforms by restricting access to system memory and affects DB2 functionality as follows:

- The **DB2_AWE** registry variable is discontinued.
- The **physical_page_maps** monitor element is discontinued.
- The SNAPBP administrative view no longer includes a physical_page_maps column.
- The SNAP_GET_BP table function returns a NULL value for the physical_page_maps column.

Resolution

If you are using 32-bit servers that currently use AWE to enable greater amounts of system memory, you should upgrade to 64-bit hardware, operating system, and DB2 products. You should also update scripts containing references to the **DB2_AWE** registry variable or to the **physical_page_maps** monitor element.

-w option for db2icrt, db2ilist, and db2iupdt is discontinued (Linux and UNIX)

The **-w** WordWidth parameter of the db2icrt, db2iupdt, and db2ilist commands is discontinued.

Details

The bitwidth (**-w**) option of the **db2icrt**, **db2ilist**, and **db2iupdt** commands is not valid and returns an error. This option was valid only on AIX 5L, HP-UX, Linux, and Solaris operating systems. In supported Linux and UNIX operating systems, the instance bit size is now determined by the operating system where the DB2 product is installed.

Resolution

Do not use the **-w** option for the db2icrt, db2ilist, and db2iupdt commands. In Version 9.1, this option returned a warning message, but in Version 9.5, it returns a syntax error.

DB2 Web Tools support is discontinued

Support for DB2 Web Tools is discontinued.

Details

Earlier DB2 versions supported DB2 Web Tools, a suite that consisted of the DB2 Web Command Center and the DB2 Web Health Center, targeted for use with HTTP clients.

Resolution

Modify applications and scripts to remove references to the discontinued functionality.

Related concepts

"Data server tools have been added" on page 27

Some registry and environment variables are discontinued

There are a number of registry variables that are discontinued in Version 9.5. You should remove all references to them.

The following registry and environment variables are discontinued in Version 9.5:

Table 28. Registry variables discontinued in Version 9.5

Registry or environment variable	Details
DB2_ASYNC_APPLY	This variable is discontinued in Version 9.5 because it was used to support DataJoiner®, which is no longer supported.
DB2_AWE	If you were using AWE buffer pools, consider migrating to a Version 9.5 64-bit database product to eliminate the virtual addressable memory limit. For more information, see Address Windowing Extensions (AWE) feature support discontinued (Windows)
DB2_BLOCK_ON_LOG_DISK_FULL	This variable is discontinued in Version 9.5 because it has been replaced by the **blk_log_dsk_ful** configuration parameter.
DB2CCMSRV	This variable is discontinued in Version 9.5 because it is no longer necessary.
DB2_FORCE_FCM_BP	This variable is discontinued in Version 9.5 because only 64-bit kernels of AIX operating systems are supported, and they do not have shared memory segment size restrictions. The default is to enable shared memory communications between logical nodes to improve performance and to provide consistency with other platforms.
DB2_LGPAGE_BP	This variable is discontinued in Version 9.5 because it is no longer needed. To enable large page support, use the **DB2_LARGE_PAGE_MEM** registry variable.
DB2LINUXAIO	This registry variable is discontinued because asynchronous I/O (AIO) functionality on Linux operating systems is enabled by default in Version 9.5.
DB2_MEMALLOCATE_HIGH	This registry variable is discontinued in Version 9.5 because it is no longer needed. Large memory allocations are requested by default from the high end of the virtual address space, which helps reduce address space fragmentation on Windows platforms.
DB2_MIGRATE_TS_INFO	This variable is discontinued in Version 9.5 because its usage was only required for migration to Version 5 from previous releases.
DB2_NR_CONFIG	This variable is discontinued in Version 9.5 because its functionality is no longer needed.

Table 28. Registry variables discontinued in Version 9.5 (continued)

Registry or environment variable	Details
DB2_NEWLOGPATH2	This variable is discontinued in Version 9.5 because it was replaced by the database configuration parameter **mirrorlogpath**, which provides you with more flexibility in setting a mirrored log path.
DB2_OLAP_BUFFER_SIZE	This registry variable is discontinued because On-Line Analytical Processing (OLAP) functions use sort heap memory in Version 9.5. In previous releases, OLAP functions used application heap memory, and **DB2_OLAP_BUFFER_SIZE** limited how much they could use. Because sort heap memory can be enabled for self-tuning, this variable's role is no longer required.
DB2UPMPR	This variable is discontinued because it was only used on OS/2®, which Version 9.5 does not support.
DB2UPMSINGLE	This variable is discontinued in Version 9.5 because it is no longer necessary.

Related concepts

"Some registry and environment variables are deprecated" on page 163

"Some registry and environment variables have changed" on page 126

db2undgp command is discontinued

The db2undgp (revoke execute privilege) command is no longer available.

Details

In previous versions, you could use db2undgp to prevent users from accessing SQL objects for which they did not have privileges.

Resolution

During database migration to DB2 Universal Database (DB2 UDB) Version 8, the EXECUTE privilege for all existing functions, methods, and external stored procedures was granted to all users (PUBLIC). You could use the db2undgp command to prevent users from accessing SQL objects for which they did not have privileges. In Version 9.5, you may revoke the EXECUTE privilege from the PUBLIC group.

-n option of the db2licm command is discontinued

The -n option of the db2licm command is discontinued.

Details

In the past, you could use the -n option to update the number of processors that you are entitled to use with a DB2 product. Licenses are now determined by the number of value units rather than by the number of physical processors. This option therefore does not produce any results if used with Version 9.5 or later products.

Resolution

You should not use the discontinued option. You are not required to update the number of entitled processors.

Related reference

"db2licm - License management tool command" in Command Reference

CLISchema CLI keyword is discontinued

The **CLISchema** keyword is discontinued for Version 9.5 clients connecting to Version 9.5 database servers.

Details

Setting the **CLISchema** keyword improved performance, primarily for client applications connecting to DB2 for z/OS. In Version 9.1, support for this keyword was deprecated for Version 9.1 clients connecting to DB2 Version 9.1 for Linux, UNIX, and Windows database servers and discontinued for Version 9.1 clients connecting to DB2 for z/OS database servers.

Resolution

You can use the **SysSchema** keyword as a replacement to indicate an alternative schema.

Related reference

"SysSchema CLI/ODBC Configuration Keyword" in Call Level Interface Guide and Reference, Volume 2

Chapter 19. DB2 Version 9.1 fix pack changes that affect DB2 Version 9.5 usage

Version 9.1 Fix Pack 3 (and earlier fix packs) include changes to features and functionality that might affect your use of Version 9.5.

Details

If you did not apply the Version 9.1 Fix Pack 3 or earlier fix packs or did not update your local Information Center since Version 9.1 was available, you might not be aware of all of the changes that might affect your use of Version 9.5.

Resolution

Review the following topics if you are unfamiliar with the technical changes included in the Version 9.1 fix packs. Fix packs are cumulative: they contain all of the changes and functionality shipped in previous fix packs.

DB2 Version 9.1 Fix Pack 1

Fix Pack 1 includes the following changes to existing functionality:
- Address Windowing Extensions (AWE) support is deprecated (Windows)
- Availability of LOB or XML values is changed in JDBC applications with progressive streaming
- Modification level of product identifier can contain alphanumeric characters

Fix Pack 1 also includes the following enhancements:
- DB2Binder utility includes two new options
- Optimization profile table can be created through a stored procedure

DB2 Version 9.1 Fix Pack 2

Fix Pack 2 contains the Fix Pack 1 functionality and includes also the following changes to existing functionality:
- Configuration parameter default value changes for java_heap_sz
- Network Information Services (NIS and NIS+) support is deprecated (Linux and UNIX)
- The -schema option of the db2sampl command is discontinued

Fix Pack 2 also includes the following enhancements:
- Annotated XML schema decomposition performance is improved
- BINARY, VARBINARY, and DECFLOAT data type support added for C and C++ embedded SQL applications
- DB2 .NET Data Provider enhancements and support for the .NET Framework 2.0
- DRDA and Informix wrapper support added for HP-UX
- IBM Database Add-Ins for Visual Studio 2005 enhancements
- IBM Software Development Kit (SDK) for Java 5.x support added for the Solaris Operating System
- New keyword can improve query performance in DPF environments

- Secure Sockets Layer (SSL) protocol is supported by the database server
- Windows Vista support added (Windows)

DB2 Version 9.1 Fix Pack 3

Fix Pack 3 contains the Fix Pack 2 functionality and includes also the following change to existing functionality:

- LDAP Security plug-ins are updated and include support for the Open LDAP server

Fix Pack 3 also includes the following enhancements:

- Additional territory codes and code pages supported
- Change password support added (Linux)
- COLLATION_KEY_BIT scalar function added
- JDBC and SQLJ enhancements
- Query Patroller support added for HP-UX

Part 3. Appendixes

Appendix A. File system caching configurations

The operating system, by default, caches file data that is read from and written to disk.

A typical read operation involves physical disk access to read the data from disk into the file system cache, and then to copy the data from the cache to the application buffer. Similarly, a write operation involves physical disk access to copy the data from the application buffer into the file system cache, and then to copy it from the cache to the physical disk. This behavior of caching data at the file system level is reflected in the FILE SYSTEM CACHING clause of the CREATE TABLESPACE statement. Since the database manager manages its own data caching using buffer pools, the caching at the file system level is not needed if the size of the buffer pool is tuned appropriately.

Note: The database manager already prevents caching of most DB2 data, except temporary data and LOBs on AIX, by invalidating the pages from the cache.

In some cases, caching at the file system level and in the buffer pools causes performance degradation because of the extra CPU cycles required for the double caching. To avoid this double caching, most file systems have a feature that disables caching at the file system level. This is generically referred to as *non-buffered I/O*. On UNIX, this feature is commonly known as *Direct I/O (or DIO)*. On Windows, this is equivalent to opening the file with the FILE_FLAG_NO_BUFFERING flag. In addition, some file systems such as IBM JFS2 or Symantec VERITAS VxFS also support enhanced Direct I/O, that is, the higher-performing *Concurrent I/O (CIO)* feature. The database manager supports this feature with the NO FILE SYSTEM CACHING table space clause. When this is set, the database manager automatically takes advantage of CIO on file systems where this feature exists. This feature might help to reduce the memory requirements of the file system cache, thus making more memory available for other uses.

Prior to Version 9.5, the keyword FILE SYSTEM CACHING was implied if neither NO FILE SYSTEM CACHING nor FILE SYSTEM CACHING was specified. With Version 9.5, if neither keyword is specified, the default, NO FILE SYSTEM CACHING, is used. This change affects only newly created table spaces. Existing table spaces created prior to Version 9.5 are not affected. This change applies to AIX, Linux, Solaris, and Windows with the following exceptions, where the default behavior remains to be FILE SYSTEM CACHING:
- AIX JFS
- Solaris non-VxFS
- Linux for System z™
- All SMS temporary table space files
- SMS permanent table space files, except for long field (LF) data and large object (LOB) data files.

To override the default setting, specify FILE SYSTEM CACHING or NO FILE SYSTEM CACHING.

Supported configurations

Table 29 shows the supported configuration for using table spaces without file system caching. It also indicates: (a) whether DIO or enhanced DIO will be used in each case, and (b) the default behavior when neither NO FILE SYSTEM CACHING nor FILE SYSTEM CACHING is specified for a table space based on the platform and file system type.

Table 29. Supported configurations for table spaces without file system caching

Platforms	File system type and minimum level required	DIO or CIO requests submitted by the database manager when NO FILE SYSTEM CACHING is specified	Default behavior when neither NO FILE SYSTEM CACHING nor FILE SYSTEM CACHING is specified
AIX 5.3+	Journal File System (JFS)	DIO	FILE SYSTEM CACHING (See Note 1.)
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 Version 11i v2 (Itanium®)	VERITAS Storage Foundation 4.1 (VxFS)	CIO	FILE SYSTEM CACHING
Solaris 9	UNIX File System (UFS)	DIO	FILE SYSTEM CACHING (See Note 2.)
Solaris 10	UNIX File System (UFS)	CIO	FILE SYSTEM CACHING (See Note 2.)
Solaris 9, 10	VERITAS Storage Foundation for DB2 4.1 (VxFS)	CIO	NO FILE SYSTEM CACHING
Linux distributions SLES 9+ and RHEL 4+ (on these architectures: x86, x86_64, IA64, POWER)	ext2, ext3, reiserfs	DIO	NO FILE SYSTEM CACHING
Linux distributions SLES 9+ and RHEL 4+ (on these architectures: x86, x86_64, IA64, POWER)	VERITAS Storage Foundation 4.1 (VxFS)	CIO	NO FILE SYSTEM CACHING
Linux distributions SLES 9+ and RHEL 4+ (on this architecture: zSeries®)	ext2, ext3 or reiserfs on a Small Computer System Interface (SCSI) disks using Fibre Channel Protocol (FCP)	DIO	FILE SYSTEM CACHING
Windows	No specific requirement, works on all DB2 supported file systems	DIO	NO FILE SYSTEM CACHING

Note:

1. On AIX JFS, FILE SYSTEM CACHING is the default.

2. On Solaris UFS, FILE SYSTEM CACHING is the default.
3. The VERITAS Storage Foundation for the database manager might have different operating system prerequisites. The platforms listed above are the supported platforms for the current release. Consult the VERITAS Storage Foundation for DB2 support for prerequisite information.
4. If SFDB2 5.0 is used instead of the above minimum levels, the SFDB2 5.0 MP1 RP1 release must be used. This release includes fixes that are specific to the 5.0 version.
5. If you do not want the database manager to choose NO FILE SYSTEM CACHING for the default setting, specify FILE SYSTEM CACHING in the relevant SQL, commands, or APIs.

Examples

Example 1: By default, this new table space will be created using non-buffered I/O; the NO FILE SYSTEM CACHING clause is implied:

```
CREATE TABLESPACE table space name ...
```

Example 2: On the following statement, the NO FILE SYSTEM CACHING clause indicates that file system level caching will be OFF for this particular table space:

```
CREATE TABLESPACE table space name ... NO FILE SYSTEM CACHING
```

Example 3: The following statement disables file system level caching for an existing table space:

```
ALTER TABLESPACE table space name ... NO FILE SYSTEM CACHING
```

Example 4: The following statement enables file system level caching for an existing table space:

```
ALTER TABLESPACE table space name ... FILE SYSTEM CACHING
```

Appendix B. Overview of the DB2 technical information

DB2 technical information is available through the following tools and methods:
- DB2 Information Center
  - Topics (Task, concept and reference topics)
  - Help for DB2 tools
  - Sample programs
  - Tutorials
- DB2 books
  - PDF files (downloadable)
  - PDF files (from the DB2 PDF DVD)
  - printed books
- Command line help
  - Command help
  - Message help

Note: The DB2 Information Center topics are updated more frequently than either the PDF or the hard-copy books. To get the most current information, install the documentation updates as they become available, or refer to the DB2 Information Center at ibm.com®.

You can access additional DB2 technical information such as technotes, white papers, and IBM Redbooks® publications online at ibm.com. Access the DB2 Information Management software library site at http://www.ibm.com/software/data/sw-library/.

Documentation feedback

We value your feedback on the DB2 documentation. If you have suggestions for how to improve the DB2 documentation, send an email to db2docs@ca.ibm.com. The DB2 documentation team reads all of your feedback, but cannot respond to you directly. Provide specific examples wherever possible so that we can better understand your concerns. If you are providing feedback on a specific topic or help file, include the topic title and URL.

Do not use this email address to contact DB2 Customer Support. If you have a DB2 technical issue that the documentation does not resolve, contact your local IBM service center for assistance.

DB2 technical library in hardcopy or PDF format

The following tables describe the DB2 library available from the IBM Publications Center at www.ibm.com/shop/publications/order. English DB2 Version 9.5 manuals in PDF format and translated versions can be downloaded from www.ibm.com/support/docview.wss?rs=71&uid=swg2700947.

Although the tables identify books available in print, the books might not be available in your country or region.

The form number increases each time a manual is updated. Ensure that you are reading the most recent version of the manuals, as listed below.

Note: The DB2 Information Center is updated more frequently than either the PDF or the hard-copy books.

Table 30. DB2 technical information

Name	Form Number	Available in print
Administrative API Reference	SC23-5842-01	Yes
Administrative Routines and Views	SC23-5843-01	No
Call Level Interface Guide and Reference, Volume 1	SC23-5844-01	Yes
Call Level Interface Guide and Reference, Volume 2	SC23-5845-01	Yes
Command Reference	SC23-5846-01	Yes
Data Movement Utilities Guide and Reference	SC23-5847-01	Yes
Data Recovery and High Availability Guide and Reference	SC23-5848-01	Yes
Data Servers, Databases, and Database Objects Guide	SC23-5849-01	Yes
Database Security Guide	SC23-5850-01	Yes
Developing ADO.NET and OLE DB Applications	SC23-5851-01	Yes
Developing Embedded SQL Applications	SC23-5852-01	Yes
Developing Java Applications	SC23-5853-01	Yes
Developing Perl and PHP Applications	SC23-5854-01	No
Developing User-defined Routines (SQL and External)	SC23-5855-01	Yes
Getting Started with Database Application Development	GC23-5856-01	Yes
Getting Started with DB2 installation and administration on Linux and Windows	GC23-5857-01	Yes
Internationalization Guide	SC23-5858-01	Yes
Message Reference, Volume 1	GI11-7855-00	No
Message Reference, Volume 2	GI11-7856-00	No
Migration Guide	GC23-5859-01	Yes
Net Search Extender Administration and User's Guide	SC23-8509-01	Yes
Partitioning and Clustering Guide	SC23-5860-01	Yes
Query Patroller Administration and User's Guide	SC23-8507-00	Yes
Quick Beginnings for IBM Data Server Clients	GC23-5863-01	No

Table 30. DB2 technical information (continued)

Name	Form Number	Available in print
Quick Beginnings for DB2 Servers	GC23-5864-01	Yes
Spatial Extender and Geodetic Data Management Feature User's Guide and Reference	SC23-8508-01	Yes
SQL Reference, Volume 1	SC23-5861-01	Yes
SQL Reference, Volume 2	SC23-5862-01	Yes
System Monitor Guide and Reference	SC23-5865-01	Yes
Troubleshooting Guide	GI11-7857-01	No
Tuning Database Performance	SC23-5867-01	Yes
Visual Explain Tutorial	SC23-5868-00	No
What's New	SC23-5869-01	Yes
Workload Manager Guide and Reference	SC23-5870-01	Yes
pureXML Guide	SC23-5871-01	Yes
XQuery Reference	SC23-5872-01	No

Table 31. DB2 Connect-specific technical information

Name	Form Number	Available in print
Quick Beginnings for DB2 Connect Personal Edition	GC23-5839-01	Yes
Quick Beginnings for DB2 Connect Servers	GC23-5840-01	Yes
DB2 Connect User's Guide	SC23-5841-01	Yes

Table 32. Information Integration technical information

Name	Form Number	Available in print
Information Integration: Administration Guide for Federated Systems	SC19-1020-01	Yes
Information Integration: ASNCLP Program Reference for Replication and Event Publishing	SC19-1018-02	Yes
Information Integration: Configuration Guide for Federated Data Sources	SC19-1034-01	No
Information Integration: SQL Replication Guide and Reference	SC19-1030-01	Yes
Information Integration: Introduction to Replication and Event Publishing	SC19-1028-01	Yes

Ordering printed DB2 books

If you require printed DB2 books, you can buy them online in many but not all countries or regions. You can always order printed DB2 books from your local IBM representative. Keep in mind that some softcopy books on the *DB2 PDF Documentation* DVD are unavailable in print. For example, neither volume of the *DB2 Message Reference* is available as a printed book.

Printed versions of many of the DB2 books available on the DB2 PDF Documentation DVD can be ordered for a fee from IBM. Depending on where you are placing your order from, you may be able to order books online, from the IBM Publications Center. If online ordering is not available in your country or region, you can always order printed DB2 books from your local IBM representative. Note that not all books on the DB2 PDF Documentation DVD are available in print.

Note: The most up-to-date and complete DB2 documentation is maintained in the DB2 Information Center at http://publib.boulder.ibm.com/infocenter/db2luw/v9r5.

To order printed DB2 books:

- To find out whether you can order printed DB2 books online in your country or region, check the IBM Publications Center at http://www.ibm.com/shop/publications/order. You must select a country, region, or language to access publication ordering information and then follow the ordering instructions for your location.
- To order printed DB2 books from your local IBM representative:
  1. Locate the contact information for your local representative from one of the following Web sites:
     - The IBM directory of world wide contacts at www.ibm.com/planetwide
     - The IBM Publications Web site at http://www.ibm.com/shop/publications/order. You will need to select your country, region, or language to the access appropriate publications home page for your location. From this page, follow the "About this site" link.
  2. When you call, specify that you want to order a DB2 publication.
  3. Provide your representative with the titles and form numbers of the books that you want to order. For titles and form numbers, see "DB2 technical library in hardcopy or PDF format" on page 187.

Displaying SQL state help from the command line processor

DB2 returns an SQLSTATE value for conditions that could be the result of an SQL statement. SQLSTATE help explains the meanings of SQL states and SQL state class codes.

To invoke SQL state help, open the command line processor and enter:

```
? sqlstate or ? class code
```

where *sqlstate* represents a valid five-digit SQL state and *class code* represents the first two digits of the SQL state.

For example, ? 08003 displays help for the 08003 SQL state, and ? 08 displays help for the 08 class code.

Accessing different versions of the DB2 Information Center

For DB2 Version 9.5 topics, the DB2 Information Center URL is http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/

For DB2 Version 9 topics, the DB2 Information Center URL is http://publib.boulder.ibm.com/infocenter/db2luw/v9/

For DB2 Version 8 topics, go to the Version 8 Information Center URL at: http://publib.boulder.ibm.com/infocenter/db2luw/v8/

Displaying topics in your preferred language in the DB2 Information Center

The DB2 Information Center attempts to display topics in the language specified in your browser preferences. If a topic has not been translated into your preferred language, the DB2 Information Center displays the topic in English.

- To display topics in your preferred language in the Internet Explorer browser:
  1. In Internet Explorer, click the **Tools** —> **Internet Options** —> **Languages...** button. The Language Preferences window opens.
  2. Ensure your preferred language is specified as the first entry in the list of languages.
     - To add a new language to the list, click the **Add...** button.

       Note: Adding a language does not guarantee that the computer has the fonts required to display the topics in the preferred language.
     - To move a language to the top of the list, select the language and click the **Move Up** button until the language is first in the list of languages.
  3. Clear the browser cache and then refresh the page to display the DB2 Information Center in your preferred language.
- To display topics in your preferred language in a Firefox or Mozilla browser:
  1. Select the button in the **Languages** section of the **Tools** —> **Options** —> **Advanced** dialog. The Languages panel is displayed in the Preferences window.
  2. Ensure your preferred language is specified as the first entry in the list of languages.
     - To add a new language to the list, click the **Add...** button to select a language from the Add Languages window.
     - To move a language to the top of the list, select the language and click the **Move Up** button until the language is first in the list of languages.
  3. Clear the browser cache and then refresh the page to display the DB2 Information Center in your preferred language.

On some browser and operating system combinations, you might have to also change the regional settings of your operating system to the locale and language of your choice.

Updating the DB2 Information Center installed on your computer or intranet server

If you have installed the DB2 Information Center locally, you can obtain and install documentation updates from IBM.

Updating your locally-installed DB2 Information Center requires that you:

1. Stop the DB2 Information Center on your computer, and restart the Information Center in stand-alone mode. Running the Information Center in stand-alone mode prevents other users on your network from accessing the Information Center, and allows you to apply updates. Non-Administrative and Non-Root DB2 Information Centers always run in stand-alone mode. .

2. Use the Update feature to see what updates are available. If there are updates that you would like to install, you can use the Update feature to obtain and install them

   Note: If your environment requires installing the DB2 Information Center updates on a machine that is not connected to the internet, you have to mirror the update site to a local file system using a machine that is connected to the internet and has the DB2 Information Center installed. If many users on your network will be installing the documentation updates, you can reduce the time required for individuals to perform the updates by also mirroring the update site locally and creating a proxy for the update site.
   If update packages are available, use the Update feature to get the packages. However, the Update feature is only available in stand-alone mode.

3. Stop the stand-alone Information Center, and restart the DB2 Information Center on your computer.

Note: On Windows Vista, the commands listed below must be run as an administrator. To launch a command prompt or graphical tool with full administrator privileges, right-click on the shortcut and then select **Run as administrator**.

To update the DB2 Information Center installed on your computer or intranet server:

1. Stop the DB2 Information Center.
   - On Windows, click **Start** → **Control Panel** → **Administrative Tools** → **Services**. Then right-click on **DB2 Information Center** service and select **Stop**.
   - On Linux, enter the following command:
     ```
     /etc/init.d/db2icdv95 stop
     ```
2. Start the Information Center in stand-alone mode.
   - On Windows:
     a. Open a command window.
     b. Navigate to the path where the Information Center is installed. By default, the DB2 Information Center is installed in the <Program Files>\IBM\DB2 Information Center\Version 9.5 directory, where <Program Files> represents the location of the Program Files directory.
     c. Navigate from the installation directory to the doc\bin directory.
     d. Run the help_start.bat file:
        ```
        help_start.bat
        ```
   - On Linux:

a. Navigate to the path where the Information Center is installed. By default, the DB2 Information Center is installed in the /opt/ibm/db2ic/V9.5 directory.

b. Navigate from the installation directory to the doc/bin directory.

c. Run the help_start script:

```
help_start
```

The systems default Web browser launches to display the stand-alone Information Center.

3. Click the **Update** button ( ). On the right hand panel of the Information Center, click **Find Updates**. A list of updates for existing documentation displays.

4. To initiate the installation process, check the selections you want to install, then click **Install Updates**.

5. After the installation process has completed, click **Finish**.

6. Stop the stand-alone Information Center:
   - On Windows, navigate to the installation directory's doc\bin directory, and run the help_end.bat file:
     ```
     help_end.bat
     ```

     Note: The help_end batch file contains the commands required to safely terminate the processes that were started with the help_start batch file. Do not use `Ctrl-C` or any other method to terminate help_start.bat.

   - On Linux, navigate to the installation directory's doc/bin directory, and run the help_end script:
     ```
     help_end
     ```

     Note: The help_end script contains the commands required to safely terminate the processes that were started with the help_start script. Do not use any other method to terminate the help_start script.

7. Restart the DB2 Information Center.
   - On Windows, click **Start** → **Control Panel** → **Administrative Tools** → **Services**. Then right-click on **DB2 Information Center** service and select **Start**.
   - On Linux, enter the following command:
     ```
     /etc/init.d/db2icdv95 start
     ```

The updated DB2 Information Center displays the new and updated topics.

DB2 tutorials

The DB2 tutorials help you learn about various aspects of DB2 products. Lessons provide step-by-step instructions.

Before you begin

You can view the XHTML version of the tutorial from the Information Center at http://publib.boulder.ibm.com/infocenter/db2help/.

Some lessons use sample data or code. See the tutorial for a description of any prerequisites for its specific tasks.

DB2 tutorials

To view the tutorial, click on the title.

"pureXML" in *pureXML Guide*
> Set up a DB2 database to store XML data and to perform basic operations with the native XML data store.

"Visual Explain" in *Visual Explain Tutorial*
> Analyze, optimize, and tune SQL statements for better performance using Visual Explain.

DB2 troubleshooting information

A wide variety of troubleshooting and problem determination information is available to assist you in using DB2 products.

DB2 documentation
> Troubleshooting information can be found in the DB2 Troubleshooting Guide or the Support and Troubleshooting section of the DB2 Information Center. There you will find information on how to isolate and identify problems using DB2 diagnostic tools and utilities, solutions to some of the most common problems, and other advice on how to solve problems you might encounter with your DB2 products.

DB2 Technical Support Web site
> Refer to the DB2 Technical Support Web site if you are experiencing problems and want help finding possible causes and solutions. The Technical Support site has links to the latest DB2 publications, TechNotes, Authorized Program Analysis Reports (APARs or bug fixes), fix packs, and other resources. You can search through this knowledge base to find possible solutions to your problems.
>
> Access the DB2 Technical Support Web site at http://www.ibm.com/ software/data/db2/udb/support.html

Terms and Conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

Personal use: You may reproduce these Publications for your personal, non commercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these Publications, or any portion thereof, without the express consent of IBM.

Commercial use: You may reproduce, distribute and display these Publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these Publications, or reproduce, distribute or display these Publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the Publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the Publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Appendix C. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country/region or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country/region where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

This document may provide links or references to non-IBM Web sites and resources. IBM makes no representations, warranties, or other commitments whatsoever about any non-IBM Web sites or third-party resources that may be referenced, accessible from, or linked from this document. A link to a non-IBM Web site does not mean that IBM endorses the content or use of such Web site or

its owner. In addition, IBM is not a party to or responsible for any transactions you may enter into with third parties, even if you learn of such parties (or use a link to such parties) from an IBM site. Accordingly, you acknowledge and agree that IBM is not responsible for the availability of such external sites or resources, and is not responsible or liable for any content, services, products, or other materials on or available from those sites or resources. Any software provided by third parties is subject to the terms and conditions of the license that accompanies that software.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information that has been exchanged, should contact:

IBM Canada Limited
   Office of the Lab Director
   8200 Warden Avenue
   Markham, Ontario
   L6G 1C7
   CANADA

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems, and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements, or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious, and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information may contain sample application programs, in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (*your company name*) (*year*). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. *_enter the year or years_*. All rights reserved.

Trademarks

The following terms are trademarks or registered trademarks of the International Business Machines Corporation in the United States, other countries, or both.

pureXML	Informix
DB2	POWER6
AIX	System z
i5/OS	POWER
WebSphere	OS/390
DB2 Connect	DB2 Universal Database
UniData	Redbooks
z/OS	PowerPC
System i	IBM
zSeries	Cloudscape
Tivoli	DRDA
OS/400	ibm.com
pSeries	OS/2
UniVerse	DataJoiner
iSeries	

The following terms are trademarks or registered trademarks of other companies
- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
- Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
- UNIX is a registered trademark of The Open Group in the United States and other countries.
- Itanium is a registered trademark of Intel Corporation or its subsidiaries in the United States and other countries.
- Microsoft, and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Index

A

ADC (automatic dictionary creation)
   overview   31
Address Windowing Extensions (AWE)
   discontinued support   173
ADMIN_CMD procedure
   multi-partition backup enhancement   156
ADMIN_GET_TAB_INFO function
   deprecated functionality   133
administration
   modifications summary   123
   non-root users   102
administrative routines
   additions   36, 133
   changes   133
administrative views
   additions   36, 133
   changes   133
   ENV_FEATURE_INFO   37
agent_stack_sz database manager configuration parameter
   default value change   123
agentpri configuration parameter
   deprecated functionality   123
agents_waiting_on_token monitor element
   deprecated functionality   166
agents_waiting_top monitor element
   deprecated functionality   166
ALTER BUFFERPOOL statement
   ESTORE changes   173
ALTER SECURITY LABEL COMPONENT statement
   overview   51
ALTER SECURITY POLICY statement
   overview   51
ALTER TABLESPACE statement
   space reclamation enhancement   38
APIs
   db2Backup
      enhancements   156
   db2inspect
      index checking enhancements   118
   db2Rollforward
      minimum recovery time enhancement   98
   incompatibilities   153
   piActionString data structure
      deprecated functionality   171
   sqluadau
      deprecated functionality   166
app_ctl_heap_sz database configuration parameter
   deprecated functionality   140, 145
appgroup_mem_sz database manager configuration parameter
   deprecated functionality   140, 145
appl_memory database configuration parameter
   overview   30, 140
applheapsz database configuration parameter
   changes   140
   enhancements   31, 145
application development
   enhancements summary   14, 67
   federation enhancements   107
   JDBC 3.0 enhancements   74
   JDBC 4.0 support   81

application development *(continued)*
   Linux and UNIX   150
   modifications summary   148
   new samples summary   87
   SQLJ enhancements   74, 81
application savepoints
   federation support   107
applications
   deployment enhancement on Windows   25
   distribution simplification on Windows   25
   error toleration improvement   119
   new samples summary   87
ARRAY data type
   overview   72
audit facility
   authority requirement changes   131
   enhancements   49, 156
authority_lvl monitor element
   deprecated functionality   166
auto_del_rec_obj database configuration parameter
   overview   94, 140
auto_stmt_stats database configuration parameter
   overview   140
AUTOMAINT_GET_POLICY stored procedure
   overview   93
AUTOMAINT_GET_POLICYFILE stored procedure
   overview   93
AUTOMAINT_SET_POLICY stored procedure
   overview   93
AUTOMAINT_SET_POLICYFILE stored procedure
   overview   93
automatic dictionary creation (ADC)
   overview   31
automatic maintenance
   configuring policies   93
   policy information collection   93
autonomic enhancements
   summary   3
AWE (Address Windowing Extensions)
   discontinued support   173

B

BACKUP DATABASE command
   multi-partition backup enhancement   156
backups
   enhancements summary   93
   file permission changes   137
   multi-partition database enhancement   97, 156
   partitioned database enhancement   97
   privilege changes   137
   snapshot   94, 99
Big5-HKSCS
   Unicode conversion   114
BIND command
   post-installation automatiion   145
bitwise scalar functions
   overview   91
books
   printed
      ordering   190

IBM ®

Spine information:

DB2 Version 9.5 for Linux, UNIX, and Windows

What's New

IBM