

Ingres® 2006 Release 3

Release Summary

INGRES®

October 2007

This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Ingres Corporation ("Ingres") at any time.

This Documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of Ingres. This Documentation is proprietary information of Ingres and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of this Documentation for their own internal use, provided that all Ingres copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software are permitted to have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. The user consents to Ingres obtaining injunctive relief precluding any unauthorized use of the Documentation. Should the license terminate for any reason, it shall be the user's responsibility to return to Ingres the reproduced copies or to certify to Ingres that same have been destroyed.

To the extent permitted by applicable law, INGRES PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IN NO EVENT WILL INGRES BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USER OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF INGRES IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

The use of any product referenced in this Documentation and this Documentation is governed by the end user's applicable license agreement.

The manufacturer of this Documentation is Ingres Corporation.

For government users, the Documentation is delivered with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227-7013 or applicable successor provisions.

Copyright © 2007 Ingres Corporation. All Rights Reserved.

Ingres, OpenROAD, and EDBC are registered trademarks of Ingres Corporation. All other trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Contents

Chapter 1: New Features and Enhancements in Ingres 2006 Release 3 **7**

DBMS Server Enhancements.....	7
Scrollable Cursors.....	7
LOB Locators	7
UTF8 Character Set.....	8
Improved Out of the Box Defaults	9
Automatic Storage Structure for New Tables.....	9
Additional SQL Functions	10
Incremental Rollforwarddb	11
Improved Exception Handling	11
Increased Precision for Decimal Data Type	11
Improved Performance of String Comparisons	11
Changes to ANSI Date Feature	12
Supportability Enhancements.....	12
Logging of Verifydb, Chkpdb, and Rollforwarddb.....	12
Ability to View Cursor Definition Text for an Executing Fetch	12
Server Type Reported for Terminated Programs	13
Connectivity Enhancements	13
LOB Locator Support in JDBC and OpenAPI.....	13
Scrollable Cursors in JDBC and OpenAPI	14
Connection Pooling in ODBC CLI (UNIX and VMS)	14
.NET Data Provider Enhancements	15
Performance Improvements in Network Communications (UNIX and Windows)	15

Chapter 2: New Features and Enhancements in Ingres 2006 Release 2 **17**

New Features in the DBMS Server	17
Derived Tables	17
ANSI Date and Time Support.....	18
BEFORE Triggers	18
SQL Language Enhancement—Describe Input Statement	19
Indexes on Temporary Tables.....	19
Syntax for Referencing Temporary Tables	19
Sequence Defaults.....	19
Automatic Coercion Between Integers and Strings	20
Date Functions	20
Ease of Use Enhancements.....	20
Name Server Registration Management	20

Server Management for GCF Servers	21
Connectivity Enhancements	21
ODBC Enhancements	22
JDBC Enhancements	23
Ingres .NET Data Provider 2.0	23
PHP 5 Support.....	23
Support for IPv6 Networks	24
Supportability Enhancement	31
Usability Enhancements	32
Removed or Deprecated Features	32

Chapter 3: New Features and Enhancements in Ingres 2006 **33**

What Is Ingres 2006?	33
Features Included in Open Source	33
Features Not Included in Open Source	34
New Features for Database Administrators.....	34
Parallel Query	34
Key Range Table Partitioning	34
Online Modify	35
Ingres Cluster Solution for Linux.....	35
Ingres High Availability Option.....	35
Unextenddb Utility	35
Killing Queries.....	35
Numeric Overflow Support in Report-Writer	35
Collation Specification at the Column Level.....	36
System-wide Setting for Default Lock Level	36
New Features for Application Developers	37
Automatic Sequence Number Generation	37
No Wait for Lock Requests	37
Support for New Data Types.....	38
Additions to the Visual DBA Suite	38
Visual Database Objects Differences Analyzer	39
Visual Configuration Differences Analyzer.....	40
Export Assistant	41
Connectivity Enhancements	41
Support for JDBC 3.0 API.....	41
Data Access Server	42
JDBC Driver	42
Updateable Result Sets in JDBC	43
.NET Data Provider and Visual Studio .NET Integration.....	44
Ingres ODBC Administrator	46
WinSock 2.2 API TCP/IP Protocol Driver for Windows	46

ODBC Call-level Interface	47
New Features for Linux	47
RPM Packaging	47
KDE/GNOME Desktop Integration	47
Linux Cluster Support	47
Changes to Existing Features	48
Enhanced Unicode Support	48
Complex Query Optimization	48
Increased Range Table Limit	49
JDBC User ID Enhancements	49
Increased Column Limit	50
VDBA Enhancements	50
Ingres Visual Manager Enhancements	54
Configuration Rules System Enhancements	56
Shadow Copy of the Symbol Table	56
Additional Join Functionality	56
Improved Out-of-the-Box Configuration Defaults	56
Improved IMA Support	56
CREATEDB Enhancements	57
ALTERDB Enhancements	57
Terminal Monitor Enhancements	57
Enhancements for Log Full	57
Extended B-tree Limits	57
Installer Enhancements	58
Installation as a User Other Than ingres	58
Supportability Enhancements	58
Help System Enhancements	59

Chapter 1: New Features and Enhancements in Ingres 2006 Release 3

This section contains the following topics:

[DBMS Server Enhancements](#) (see page 7)

[Supportability Enhancements](#) (see page 12)

[Connectivity Enhancements](#) (see page 13)

DBMS Server Enhancements

Scrollable Cursors

An ANSI/ISO SQL-92 feature, scrollable cursors are database query result sets that are maintained in the database server as long as the cursor is open and that allow the user to retrieve rows of a result set in any sequence. The number of elements of the result set is easily determined by the application that is using the cursor.

Scrollable cursors greatly ease the display of information, for example, in scrolling web applications.

Scrollable cursors are supported in Ingres OpenAPI and JDBC.

For more information, see the *SQL Reference Guide*, *OpenAPI User Guide*, and the *Connectivity Guide*.

LOB Locators

Ingres supports LOB locators in compliance with the ANSI SQL 2003 Standard. LOB locators, which are passed between the client and server, are pointers to the LOB data, which can be stored in or outside a table. They permit the client to access LOB data outside the scope of row data retrieval, so that the client can control the processing and storage requirement of the LOB data.

LOB locators are supported in Ingres JDBC.

UTF8 Character Set

Ingres supports the UTF8 character set. The UTF8 character set can be selected during installation.

This feature lets you store multi-byte UTF-8 encoded Unicode characters into char, varchar, and long varchar strings.

If the server character set is UTF8, then by default any database that is created on the server, is created Unicode-enabled with Normalization Form C (NFC) with default UNICODE collation, even if it is not explicitly defined. Thus char, varchar, and long varchar columns (as well as nchar, nvarchar, and long nvarchar) use the UNICODE collation by default.

A new collation UNICODE_FRENCH is added to support French Unicode collation.

If the database you are connecting to is Unicode-enabled, the UNICODE collation is loaded.

Note: When creating a table in an installation set to the UTF8 character set, the column specification for char and varchar columns is in number of bytes (not number of characters).

String functions such as length(), substring(), position(), and so on, operate similarly on UTF-8 strings.

Coercion is supported between different string types and between string types and other Ingres data types (numeric, datetime, binary, and so on) in a UTF-8 database.

Clients installed with the UTF8 character set can connect only to a DBMS Server that uses the UTF8 character set. Only char, varchar, and long varchar columns support UTF-8. Ingres character based tools (such as the terminal monitor and ABF) show the data in UTF-8.

Support for the UTF8 character set provides compatibility and portability with other database architectures.

For more information, see the *SQL Reference Guide*, *System Administrator Guide*, and *Command Reference Guide*.

Improved Out of the Box Defaults

Ingres has new defaults for the following configuration parameters, set during installation.

The default page size (`default_page_size`) for tables is changed from 2K to 8K.

The default buffer cache size (`cache_guideline`) is configured as medium for all page sizes, resulting in increased size for the following page sizes:

- For 8K page size, increased from 16 MB to 48 MB.
- For 32K page size, increased from 12 MB to 48 MB.
- For 64K page size, increased from 11 MB to 48 MB.

DMF Cache 2K and 8K are enabled by default.

The default transaction log file size (`II_LOG_FILE_SIZE_MB`) is changed from 32 MB to 256 MB. As a result, the required minimum log file size is increased from 16 MB to 32 MB.

Automatic Storage Structure for New Tables

The storage structure of a base table, when created, is automatically determined based on the syntax used for the `CREATE TABLE` statement. If the `CREATE TABLE` statement includes at least a primary key, unique constraint, or referential (foreign key) constraint, the base table structure is set to B-tree and the usual secondary index is not built.

If the table definition includes more than one constraint, it chooses the primary key constraint over a unique constraint, and the first unique constraint over any referential constraint. For primary key or unique constraints, it also adds the `UNIQUE_SCOPE=STATEMENT` attribute to the base table structure. A dependency is added between the constraint and the base table structure so that the constraint must be explicitly dropped and re-added if the base table structure is modified.

This feature improves the initial performance of the table. Previously, all new tables were created with a heap structure, by default.

To allow for backward compatibility, this feature is enabled in a given DBMS Server by setting the configuration parameter `table_auto_structure` to ON or OFF. The default is OFF.

Additional SQL Functions

New SQL functions ease application migration.

New numeric scalar functions include:

- round - Returns a numeric value, rounded to the specified length or precision
- ceiling - Returns smallest integer greater than or equal to the argument
- floor - Returns largest integer less than or equal to the argument
- truncate - Truncates x to y decimal places
- atan2 - Arctangent of angle defined by coordinate pair (x, y)
- acos(n) - Arccosine of cosine value n
- asin - Arcsine value of sine value n
- tan - Tangent value of angle n
- pi - Value of pi (ratio of the circumference of a circle to its diameter)
- sign - Returns -1 if n < 0, 0 if n = 0, +1 if n > 0

New string scalar functions include:

- chr - Converts integer into corresponding ASCII code.
- ltrim - Returns a character expression with leading blanks removed
- rtrim - Returns a character string with trailing blanks removed
- lpad - Returns specified character string of specified length left-padded by blanks or copies of the second expression
- rpad - Returns specified character string of specified length right-padded by blanks or copies of the second expression
- replace - Replaces all occurrences of a specified string value with another string value

For more information, see the *SQL Reference Guide*.

Incremental Rollforwarddb

The incremental rollforwarddb feature allows the journals from a database to be incrementally applied, as they are generated, to a backup copy of the database. The database must have been successfully rolled forward up to that journal, and no subsequent updates taken.

This feature can be used to minimize downtime in the event that the backup database is needed for disaster recovery.

For this feature, two new options are added to the rollforwarddb command:

- -incremental
- -rollback

For details, see the *Command Reference Guide*.

Improved Exception Handling

Exception handling in the DBMS Server is improved, resulting in better responsiveness when a user interrupts a process, removes a session, or aborts a query using iimonitor.

Increased Precision for Decimal Data Type

The maximum precision for the decimal type is increased from 31 digits to 39 digits. The resulting byte size of decimal columns is increased from a maximum of 16 bytes to a maximum of 20 bytes.

Improved Performance of String Comparisons

The performance of comparisons involving char and varchar values is improved.

Changes to ANSI Date Feature

The configuration parameter `date_alias` (which was introduced in Ingres 2006 Release 2 and replaces `date_type_alias` parameter) controls whether the keyword `DATE` used to define the column data type refers to the `INGRESDATE` (default) or `ANSIDATE` data type.

In Ingres 2006 Release 3, the value of `date_alias` is communicated from the client to the server. If the client and server have different values for `date_alias`, then the server uses the `date_alias` value set in the `config.dat` file on the client. This ensures that the definition of `date_alias` intended at the client is preserved in communication to the server.

Supportability Enhancements

Logging of Verifydb, Chkpdb, and Rollforwarddb

Recovery tools `chkpdb`, `rollforwarddb`, and `verifydb` now log messages when they are executed, stating the database affected and other information. This supportability enhancement makes it easier to verify that recovery tools have been used. For example, it is now easier to tell that `verifydb` has been used to force a database consistent.

Ability to View Cursor Definition Text for an Executing Fetch

When a cursor `FETCH` is executing for a user session, the text of the cursor as defined by `DECLARE CURSOR` is visible using `iimonitor` or the Interactive Performance Monitor (`ipm`). For example:

```
Query: open ~Q cursor for select a, b, c from t1 for readonly
```

This feature makes the query string being executed available; such information can be useful for performance tuning or problem resolution.

Server Type Reported for Terminated Programs

When a program attached to the Ingres shared memory segment terminates unexpectedly, a message is written to the log to identify the running program and, if it is a command line program, the program arguments. For example:

```
Process (000033B4) died with info 'auditdb -a tstdb'.
```

When database servers and database utilities such as auditdb are executing, they attach themselves to the Ingres shared memory segment. In previous releases, when an error occurred, it was difficult to determine which memory-attached program terminated and, thus, whether the failure was critical or merely a benign error that could, for example, be resolved by re-running the utility program that failed or was canceled.

The new information logged makes it easier to debug problems associated with the shared memory segment.

Note: Because the messages appear in the log when Ingres notices the abnormal termination, their writing may be asynchronous with the program termination itself.

Connectivity Enhancements

LOB Locator Support in JDBC and OpenAPI

JDBC supports an abstraction of LOB locators through the BLOB and CLOB classes. LOB locators allow the client to request data from a specific offset in the LOB without having to retrieve the data in between. For details, see the *Connectivity Guide*.

With OpenAPI, an application can request a reference to the long data, called a locator, by setting the `IIAPI_QF_LOCATOR` flag when calling `IIapi_query()`. Locators are 4-byte integer values that reference the long data where it resides in the database. For details, see the *OpenAPI User Guide*.

Scrollable Cursors in JDBC and OpenAPI

Scrollable cursors are supported in JDBC. ResultSet types TYPE_SCROLL_INSENSITIVE and TYPE_SCROLL_SENSITIVE and the ResultSet scrolling methods are supported. For details, see the *Connectivity Guide*.

Scrollable cursors are supported in OpenAPI. The application requests a scrollable cursor by setting the IIAPI_QF_SCROLL flag when opening the cursor using IIApi_query() with query type IIAPI_QT_OPEN. A scrollable cursor can be positioned prior to calling IIApi_getColumns() using either IIApi_scroll() or IIApi_position(). IIApi_getColumns() then returns rows starting with the row specified by IIApi_scroll() or IIApi_position(). For details, see the *OpenAPI User Guide*.

Connection Pooling in ODBC CLI (UNIX and VMS)

The Ingres ODBC Call-level Interface (ODBC CLI) now supports ODBC connection pooling. Connection pooling allows connections to be shared in ODBC applications and improves performance, especially in multi-threaded applications and applications with a large number of connections.

Connection pooling is set on a per-process basis and is supported in the SQLSetEnvAttr() function. By default, ODBC connection pooling is disabled.

If the ODBC CLI detects that a pooled connection has remained connected past the defined time-out interval, the connection is terminated. A new screen in the Ingres ODBC Administrator utility (iiodbcadm) allows users to specify the connection timeout value. The minimum timeout value is 30 seconds; the maximum value is the MAXI4 number of seconds.

If a user upgrades his or her Ingres installation without running iiodbcinst, iisuodbc, or iiodbcadm, the default time-out value is -1, which indicates no timeouts.

Note: ODBC connection pooling is already supported in Windows environments.

.NET Data Provider Enhancements

Enhancements to the .NET Data Provider include:

- New keywords in the connection string:

dbms_user

Specifies the user name to be associated with the DBMS session. This keyword is equivalent to the Ingres -u flag, which can require administrator privileges.

dbms_password

Specifies the DBMS password for the user. This flag is equivalent to the Ingres -P flag.

character encoding

Specifies the .NET character encoding used for conversions between Unicode in the .NET application and character data types in the database. Typically, the character encoding is determined automatically by the data provider from the Data Access Server installation character set. This keyword allows an alternate character encoding to be specified or a valid character encoding to be used if the data provider is unable to map the server's character set.

- Integration with Visual Studio 2005 is enabled by additional properties to IngresConnectionStringBuilder. New edit boxes are added to the Connection String Editor dialog.
- The current Interval data type is split into two data types: IntervalDayToSecond and IntervalYearToMonth. The IntervalYearToMonth remains mapped to .NET String data type. The IntervalDayToSecond is mapped to .NET TimeSpan data type.

For more information, see the *Connectivity Guide*.

Performance Improvements in Network Communications (UNIX and Windows)

The code for TCP/IP communications has been optimized so that connections are faster, especially in heavy network traffic.

Chapter 2: New Features and Enhancements in Ingres 2006 Release 2

This section contains the following topics:

[New Features in the DBMS Server](#) (see page 17)

[Ease of Use Enhancements](#) (see page 20)

[Connectivity Enhancements](#) (see page 21)

[Supportability Enhancement](#) (see page 31)

[Usability Enhancements](#) (see page 32)

[Removed or Deprecated Features](#) (see page 32)

New Features in the DBMS Server

Ingres 2006 Release 2 enhancements to the DBMS Server allow for better integration by Ingres partners. New features for application developers include:

- Additional flexibility in application design.
- Simplified migration of existing applications that run against non-Ingres database architectures.

Derived Tables

A *derived table* results when you code a SELECT in the FROM clause of a SELECT or UPDATE statement.

Derived tables let you create or simplify complex queries. Useful in data warehousing applications, they provide a way to isolate complex portions of query syntax from the rest of a query.

Some complex queries cannot be implemented without using either pre-defined views or derived tables. The derived table behaves like an inline view, but is more concise and avoids having to define persistent objects that may be used for a single query only.

For details on derived tables, see the *SQL Reference Guide*.

ANSI Date and Time Support

Ingres now supports the ANSI date and time data types DATE, TIME, TIMESTAMP, and INTERVAL. This enhancement makes it easier for applications to migrate to Ingres from a non-Ingres database.

Previously, Ingres supported one date data type that could store dates, times, intervals, and time stamps. The previous date type is renamed to INGRESDATE.

The configuration parameter `date_alias` controls whether the keyword DATE used for a column data type refers to INGRESDATE or to ANSIDATE. The `date_alias` parameter is set during installation and defaults to INGRESDATE.

In Ingres 2006 Release 3, the server interprets the `date_alias` according to the client's `config.dat` setting. In Ingres2006 Release 2, however, the client and server each interpret `date_alias` according to their own `config.dat` settings. We strongly recommend, therefore, that in Ingres 2006 Release 2, both the client and server have the same value for `date_alias`.

When migrating from an earlier version of Ingres, the existing date data in the database is not affected. The data is still a valid INGRESDATE data type.

If you set `II_DATE_ALIAS` to the ANSIDATE format, existing scripts and database procedures that use the keyword `date` to imply old DATE column definitions, may need to be changed to explicitly use the INGRESDATE type.

Note: The `date_alias` parameter replaces the `date_type_alias` parameter, which is deprecated.

For details, see the *SQL Reference Guide*.

BEFORE Triggers

BEFORE triggers let an application call an Ingres database procedure before a triggering operation (INSERT, UPDATE, or DELETE) is executed. The procedure can change the values of columns in rows being inserted or updated, or can inhibit the deletion of rows, depending on their contents. In the CREATE RULE statement, the keyword BEFORE can be used in defining the table condition that triggers the rule.

In addition, in the CREATE PROCEDURE statement, you can optionally assign formal parameters a mode, IN, OUT, or INOUT. For OUT and INOUT parameters, modified parameter values can be passed back to the calling procedure or triggering operation.

For details, see the *SQL Reference Guide* and *Database Administrator Guide*.

SQL Language Enhancement—Describe Input Statement

To ease application migration, Ingres SQL supports the DESCRIBE INPUT statement. This statement obtains the number and type of input parameters of a prepared statement. Such information is necessary for products that support user-supplied queries.

For details, see the *SQL Reference Guide*.

Indexes on Temporary Tables

Ingres supports indexes on temporary tables. This feature can be used on global temporary tables for more efficient access.

For more information, see the *SQL Reference Guide*.

Syntax for Referencing Temporary Tables

This release introduces a new syntax for referencing global temporary tables. This enhancement facilitates porting applications to Ingres from other database architectures.

The new syntax drops the requirement of prefixing a temporary table name with the SESSION qualifier in a DECLARE GLOBAL TEMPORARY TABLE statement. If the SESSION schema qualifier is omitted in the declaration, all subsequent DML and DDL statements referencing the table can optionally omit it. When this syntax is used, creating temporary and permanent tables with the same name is not allowed, to avoid confusion in referencing tables.

Using the "SESSION." schema qualifier when referencing temporary tables is still required if the DECLARE GLOBAL TEMPORARY TABLE statement includes the SESSION qualifier in the table name. When this syntax is used, temporary and permanent tables can have the same name.

For more information, see the *Database Administrator Guide* and *SQL Reference Guide*.

Sequence Defaults

A column can be defined that automatically takes an increasing sequence value when one is not provided. This feature can be used to create surrogate keys.

For more information, see the *SQL Reference Guide*.

Automatic Coercion Between Integers and Strings

The DBMS Server performs automatic coercion between integer/float data types and char, varchar, nchar, or nvarchar data types. For example, an INSERT can place a quoted string value into an integer column or a database procedure can have an assignment statement that assigns a float value to a char variable—all without using explicit coercion functions such as CHAR() or INT4().

Date Functions

The following date functions extract the specified portion of a date or timestamp: year(), quarter(), month(), week(), week_iso(), day(), hour(), minute(), second(), and microsecond(). For example, year('2006-12-15') returns 2006.

For more information, see the *SQL Reference Guide*.

Ease of Use Enhancements

The following enhancements make Ingres easier to use and support:

- Name Server registration management
- Server management for GCF servers

Name Server Registration Management

The General Communications Facility is enhanced to improve the reliability, performance, and ease of use of your Ingres system. The Name Server registration mechanism no longer erroneously de-registers servers nor requires you to recycle the installation to recover servers. These problems occasionally occurred under heavy DBMS connectivity loads.

The enhancement also solves the problem with manually registering servers. Now when the iinamu utility is used to manually register a server, all necessary information is restored, clients are correctly validated, and attempts to connect with an installation password are not rejected.

As a result of this enhancement, the output of the SHOW SERVERS command in the iinamu utility is now identical in format to other SHOW *class* commands.

Server Management for GCF Servers

The iimonitor utility is an Ingres command line utility used to monitor DBMS servers, Recovery servers, and now GCF servers. The GCF servers support iimonitor commands. This feature provides additional capability to the Ingres system administrator to monitor, diagnose, and control the Name Server, Communications Server, and Data Access Server.

New iimonitor commands that are specific to GCF servers are as follows:

- set trace
- register server
- remove tickets
- remove pooled sessions

For more information, see the *Command Reference Guide*.

Connectivity Enhancements

The connectivity enhancements ease migration of applications from non-Ingres database architectures and improve interoperability.

ODBC Enhancements

The Ingres ODBC 3.5 driver is enhanced to support most of the ODBC 3.x specifications. The enhancements offer these benefits:

- Better integration with Microsoft products such as Access and Excel, and with other third-party products that use ODBC
- Easier migration from other databases, such as Oracle and SQL Server
- Improved performance

The new Ingres ODBC driver includes several new features, including the following ODBC 3.x functions:

- SQLBrowseConnect()
- SQKGetInfo()
- SQLGetTypeInfo()
- SQLDescribeParam()
- SQLColumnPrivileges()
- **Note:** Not supported on gateways except RMS.
- SQLTablePrivileges()

Note: Not supported on gateways except IMS, VSAM, and RMS.

The following functions are supported through the ODBC escape sequence syntax:

- CONVERT
- INTERVAL scalar

The following features in Ingres 2006 Release 2 are supported:

- The DESCRIBE INPUT query
- New data types for ISO dates and time intervals

For more information, see the *Connectivity Guide*.

JDBC Enhancements

The following JDBC enhancements ease application migration and performance:

- The Ingres JDBC driver supports the Java 2 Platform Standard Edition 5.0 (J2SE 5.0) specification, excluding RowSets.
- The JDBC cursor default is changed to `CURSOR=READONLY`. This cursor mode setting typically generates improved JDBC performance, especially for selects returning many rows. A configuration parameter lets you retain the old default, if preferred.
- The JDBC `ParameterMetaData` interface, which is related to the `DESCRIBE INPUT` functionality in the DBMS, is supported.
- Enhanced support is provided for XA transactions in J2EE environments using the `IngresXADataSource`.

For more information on JDBC, see the *Connectivity Guide*.

Ingres .NET Data Provider 2.0

The Ingres .NET Data Provider is enhanced to support the Microsoft .NET 2.0 Framework and MS Visual Studio 2005.

The Ingres .NET Data Provider 2.0 takes advantage of the .NET 2.0 features for greater usability, interoperability, and flexibility of Ingres data access in a Microsoft .NET environment. It supports the new base classes introduced into the data provider class hierarchy by Microsoft. For Visual Studio 2005, the data provider supports the new Server Explorer window, making it easier to manage data source definitions.

For more information, see the *Connectivity Guide*.

Note: The new `Ingres.Client` assembly of Ingres2006 Release 2 replaces the old `Ca.Ingres.Client` assembly of Ingres2006. Old applications that want to use the new `Ingres.Client` assembly require source application changes.

PHP 5 Support

The Ingres PHP driver is updated to support the features of PHP 5.

For more information, see the download page of the Ingres web site.

Support for IPv6 Networks

Ingres supports Internet Protocol version 6 (IPv6) networks. IPv6 provides more addresses for networked devices and more efficient processing than does IPv4.

Use of IPv6 by Ingres is transparent. Ingres immediately begins using IPv6, wherever possible, as the network is migrated to IPv6.

Ingres 2006 Release 2 supports both IPv6 and IPv4 addresses. IPv4 must be used to access versions of Ingres prior to Ingres 2006 Release 2.

For details, see the *Connectivity Guide*. and IPv6 Configuration (see page 24).

IPv6 Configuration

In most cases, no configuration in Ingres is required to implement IPv6 support. It is automatically enabled in the `tcp_ip` (`tcp_dec` on VMS) network protocol starting with Ingres 2006 Release 2.

Some versions of operating systems, however, provide no or limited support for IPv6. In some cases, IPv6 support is available, but must be enabled or configured in the operating system before it can be used. On systems with partial or no IPv6 functionality, Ingres will adjust automatically to the level of IPv6 support available. On some systems, however, Ingres may have difficulty, particularly when starting up or establishing connections.

For those rare situations, configuration parameters can be used to "back out" the IPv6 support, if required. The converse is also possible: the TCP support can be restricted to IPv6 addresses only. These parameters, which are not available in the configuration utilities, must be set with `set`, `ingsetenv` or `iisetres` commands.

TCP/IP and Ingres Communications

On all platforms, Ingres Net typically uses TCP/IP to communicate between Ingres installations. A typical scenario is where applications in the client installation communicate through Ingres Net with the DBMS Server in the server installation. If using JDBC or .NET applications with Ingres, then TCP/IP is used to communicate between the Ingres JDBC or .NET driver running under the application and the Data Access Server (process `iigcd`).

On Linux and UNIX only, TCP/IP is typically used to communicate between Ingres processes—that is, as the local IPC. The `tcp_ip` protocol driver used for network communications is used for local communications also. Therefore, if you experience trouble with IPv6 across the network, local communications are likely to have trouble too. If so, basic Ingres server processes such as the name server (`iigcn`) and the DBMS Server (`iidbms`) may not even start.

Parameters for Controlling IPv6 Support

The parameters for controlling IPv6 support are as follows:

- II_TCPIP_VERSION environment variable
- II_GC_PROT environment variable (Linux and UNIX only)
- ii.hostname.gcX.*.protocol.status (and port) in config.dat, where the gcX server can be gcc, gcd, or jdbc.

VMS: The only configuration options are II_TCPIP_VERSION and ii.hostname.gcX.*.tcp_ip.version.

II_TCPIP_VERSION Environment Variable—Specify Version of TCP/IP to Use

The II_TCPIP_VERSION environment variable determines the version of IP addresses that the tcp_ip (or tcp_dec on VMS) protocol driver uses. It can be set using the ingsetenv command.

Note: The equivalent configuration parameter in config.dat is tcp_ip.version.

This variable has the following format:

II_TCPIP_VERSION = *value*

value

Controls which tcp_ip protocol driver or which version of IP addresses to use, as one of the following:

ALL

(Default) Uses both IPv4 and IPv6 addresses.

VMS: The default behavior is to use both IPv4 and IPv6 addresses, and to map the IPv4 addresses as IPv6.

6

Uses IPv6 addresses only.

VMS: The IPv6 versions of the listen, accept, and connect are used.

4

Windows: Uses only IPv4 addresses with IPv6-capable functions.

Linux, UNIX, VMS: Uses the IPv4-only version of the protocol driver.

46

Linux and UNIX: Uses only IPv4 addresses with IPv6-capable functions.

II_GC_PROT Environment Variable—Set IPC Communications Protocol (Linux and UNIX Only)

On Linux and UNIX, the II_GC_PROT environment variable sets the local IPC communications protocol. This variable can be set using the `ingsetenv` command.

This variable has the following format:

II_GC_PROT = *protocol*

protocol

Specifies the local IPC communications protocol as one of the following:

TCP_IP

(Default) Uses the current TCP_IP protocol driver.

TCP_IPV4

Uses the previous version of the TCP_IP protocol driver, which supports IPv4 addresses only.

ii.hostname.gcX.*.protocol.status Resource—Set Network Communications Protocol

This resource in config.dat sets the network communications protocol for the designated server (gcX) to the appropriate Ingres network protocol driver. The gcX server can be gcc, gcd, or jdbc. The resource can be set by using the `iiisetres` command.

The format is as follows:

```
ii.hostname.gcX.*.protocol.status
```

and

```
ii.hostname.gcX.*.protocol.port
```

protocol

Specifies the Ingres network protocol driver, which can be one of the following:

tcp_ip

(All environments except VMS) (Default) Uses the current TCP_IP protocol driver.

tcp_ipv4

(Linux and UNIX only) Uses the previous version of the TCP_IP protocol driver, which supports IPv4 addresses only.

wintcp

(Windows only) Uses the previous version of the TCP_IP protocol driver, which supports IPv4 addresses only.

tcp_dec

(VMS only) Uses the current TCP_IP protocol driver.

Options for Disabling IPv6 Support

Two approaches can be used to disable IPv6 support:

- Restrict the `tcp_ip` protocol driver to only use IPv4 addresses
- Completely back out the enhanced `tcp_ip` protocol driver and use the old version of the driver.

Use IPv4 Addresses Only

The `tcp_ip` protocol driver can be restricted to listen and connect only with IPv4 style addresses. (IPv4 is the standard IP version that was used prior to IPv6).

To restrict the `tcp_ip` protocol driver to use IPv4 addresses only:

Use any one of the following options, which are functionally equivalent:

- Set the `II_TCP_VERSION` operating system variable as follows:

Windows:

```
set II_TCPIP_VERSION=4
```

Linux and UNIX:

```
set II_TCPIP_VERSION=46
```

- Set the Ingres `II_TCPIP_VERSION` environment variable as follows:

Windows:

```
ingsetenv II_TCPIP_VERSION 4
```

Linux and UNIX:

```
ingsetenv II_TCPIP_VERSION 46
```

- Set the Ingres resource as follows:

If using Ingres Net:

```
iisetres ii.machine.gcc.*.tcp_ip.version 4
```

If using Data Access Server:

```
iisetres ii.machine.gcd.*.tcp_ip.version 4
```

If using JDBC Server:

```
iisetres ii.machine.jdbc.*.tcp_ip.version 4
```

- (VMS only) (Optional) Set the "lnm" (logical name) attribute to cause `II_TCPIP_VERSION` to be defined when the servers start up:

```
iisetres ii.machine.lnm.ii_tcpip_version 4
```

Back Out IPv6 Support

To completely back out IPv6 support, you must use the old driver, which supports IPv4 only. The old driver is renamed to `tcp_ipv4` on UNIX and Linux, and is `wintcp` on Windows. The old driver is expected to be made obsolete in a future release of Ingres.

To back out the enhanced `tcp_ip` Ingres protocol driver

1. Back out network protocol for servers with `tcp_ip.status = ON`

Linux, UNIX, Windows:

```
iisetres ii.machine.gcc.*.tcp_ip.status OFF      (All platforms)
iisetres ii.machine.gcc.*.tcp_ipv4.status ON      (Unix/Linux)
iisetres ii.machine.gcc.*.tcp_ipv4.port II        (Unix/Linux)
iisetres ii.machine.gcc.*.wintcp.status ON        (Windows)
iisetres ii.machine.gcd.*.tcp_ip.status OFF      (All platforms)
iisetres ii.machine.gcd.*.tcp_ipv4.status ON      (Unix/Linux)
iisetres ii.machine.gcd.*.tcp_ipv4.port II7       (Unix/Linux)
iisetres ii.machine.gcd.*.wintcp.status ON        (Windows)
```

Set the port value to same as that used by the same server for `tcp_ip`. If using JDBC Server instead of GCD Server, replace "gcd" with "jdbc" in the above examples.

VMS:

```
define/group II_TCPIP_VERSION 4
or
define/system II_TCPIP_VERSION 4
```

2. Back out local IPC protocol (Linux and UNIX only):

```
ingsetenv II_GC_PROT tcp_ipv4
```

3. Back out network and local IPC protocol (Linux and UNIX only):

```
set II_TCPIP_VERSION=4
or
ingsetenv II_TCPIP_VERSION 4
```

Note: This step is equivalent to steps 1 and 2 above, and is the simplest way to back out to the IPv4-only driver on Linux and UNIX.

IPv6 in the JDBC Driver and Ingres .NET Data Provider

No Ingres parameters control or restrict IPv6 in the JDBC driver or the .NET Data Provider. The DAS (server side of the connection) can be configured as documented in Options for Disabling IPv6 Support (see page 27). However, there are some Java-specific networking system properties that can be set to control the IPv6 behavior in the Ingres JDBC driver (and other Java applications).

To return IPv6 addresses before IPv4 addresses:

```
java.net.preferIPv6Addresses=true
```

To restrict driver to IPv4 only:

```
java.net.preferIPv4Stack=true
```

Examples of Disabling IPv6 Support

In the following examples, assume: machine=host1, Ingres installation id=AA, startup count is 1 for Ingres Net and Data Access Server.

1. Revert to the old IPv4-only driver

Linux and UNIX:

```
ingsetenv II_GC_PROT tcp_ipv4
iisetres ii.host1.gcc.*.tcp_ip.status OFF
iisetres ii.host1.gcc.*.tcp_ipv4.status ON
iisetres ii.host1.gcc.*.tcp_ipv4.port AA
iisetres ii.host1.gcd.*.tcp_ip.status OFF
iisetres ii.host1.gcd.*.tcp_ipv4.status ON
iisetres ii.host1.gcd.*.tcp_ipv4.port AA7
```

Windows:

```
iisetres ii.host1.gcc.*.tcp_ip.status OFF
iisetres ii.host1.gcc.*.wintcp.status ON
iisetres ii.host1.gcd.*.tcp_ip.status OFF
iisetres ii.host1.gcd.*.wintcp.status ON
```

Note: wintcp port is typically already set correctly on Windows.

VMS:

```
define/group II_TCPIP_VERSION 4
```

or

```
define/sys II_TCPIP_VERSION 4
```

Optionally, set the "lnm" (logical name) attribute to cause II_TCPIP_VERSION to be defined when the servers start up:

```
iisetres ii.machine.lnm.ii_tcpip_version 4
```

2. Restrict all remote communications to IPv4 addresses only on Windows, Unix or Linux:

```
ingsetenv II_TCPIP_VERSION 4
```

Supportability Enhancement

This release includes enhanced support for debugging database procedures and for dealing with runtime errors. New trace point QE131 enables the display of the line number within a procedure at which an error condition is detected. In addition, the HELP PROCEDURE command has been enhanced for user written procedures; the procedure text is displayed with line numbers to the left of the procedure source code.

Usability Enhancements

This release contains the following usability enhancements:

- Redesigned installation wizards on Linux and Windows for easier installation.
- Installation wizard for Ingres .NET Data Provider.
- Demonstration database that can be created and populated at the end of the installation process. This database supports the demonstration application, which shows you how to code Ingres applications.
- *Quick Start Guide* describes how to begin using Ingres and connect to Ingres from various application development environments.

Removed or Deprecated Features

The JDBC Server is removed and replaced by the Data Access Server.

The Ingres Web Deployment Option is deprecated and will be removed in a future release.

Chapter 3: New Features and Enhancements in Ingres 2006

This section contains the following topics:

[What Is Ingres 2006?](#) (see page 33)
[Features Included in Open Source](#) (see page 33)
[New Features for Database Administrators](#) (see page 34)
[New Features for Application Developers](#) (see page 37)
[Additions to the Visual DBA Suite](#) (see page 38)
[Connectivity Enhancements](#) (see page 41)
[New Features for Linux](#) (see page 47)
[Changes to Existing Features](#) (see page 48)

What Is Ingres 2006?

Ingres 2006 is an enterprise-class open source database. Ingres 2006 was previously released by Computer Associates as Ingres r3.

Features Included in Open Source

The following Ingres components are contributed to the open source community:

- Ingres DBMS and associated database administration tools
- Embedded SQL precompilers
- Character-based querying, reporting, and application development tools
- Connectivity components, including ODBC, JDBC, and the .Net Data Provider
- Ingres Distributed Option (also known as Ingres Star)
- Ingres Replicator Option
- Ingres Web Deployment Option
- TP monitors, including CICS, Tuxedo, and Encina

Features Not Included in Open Source

Features not included in the open source edition are as follows:

- Support for spatial objects
- B1 security

The spatial object library is available for download from www.ingres.com if you have a valid technical support contract with Ingres Corporation.

While the source for the Visual DBA suite is not contributed to the open source community, the suite is included in the Ingres for Windows download.

The following members of the Ingres product family are not contributed to open source and continue to be available for purchase from Ingres Corporation:

- OpenROAD
- Enterprise Access
- EDBC products

New Features for Database Administrators

The new features for database administrators make it easier to administer an Ingres database and allow database administrators to deploy Ingres in a more scalable environment.

Parallel Query

As a multi-threaded server, Ingres has long supported symmetric multi-processing (SMP) systems by creating individual threads to handle user queries, and executing these queries in parallel across all available CPUs. Ingres 2006 introduces the ability to execute individual queries in parallel across all available CPUs in the system, which greatly improves performance. For more information, see the *SQL Reference Guide*.

Key Range Table Partitioning

With the functionality of key range table partitioning, the data in a database can be partitioned based upon the value of a given key, which significantly improves the performance of queries that require a full table scan. For more information, see the *SQL Reference Guide*.

Online Modify

The online modify functionality enables users to modify tables while working online. The DBMS performs the modify processing while allowing concurrent updates to the table. For more information, see the *SQL Reference Guide*.

Ingres Cluster Solution for Linux

Ingres Cluster Solution is an extension to the base Ingres product that allows it to operate on a cluster as a distributed application, providing transparent access to databases residing on shared storage devices. If there is an Ingres or hardware failure on one of the nodes, only those transactions that are in process on the failing node are aborted and ACID properties are maintained.

Ingres High Availability Option

Ingres 2006 provides automatic failover support for Ingres clusters on Sun Solaris and Windows. For more information, see the *System Administrator Guide*.

Unextenddb Utility

Ingres 2006 introduces an unextenddb utility, which provides the ability to unextend a database location. For more information, see the *Database Administrator Guide* and *Command Reference Guide*.

Killing Queries

Ingres 2006 provides the ability to kill a query in another session while leaving the session in place. For more information, see the *Command Reference Guide*.

Numeric Overflow Support in Report-Writer

Ingres 2006 provides support for numeric overflow in Report-Writer. For more information, see the *Command Reference Guide*.

Collation Specification at the Column Level

This feature allows the specification of a collation sequence at the column level that differs from the database default collation sequence. A new optional `COLLATE` clause is added to the column specification on the `CREATE TABLE` statement. The `COLLATE` clause lets you specify a case-insensitive collation for columns that contain Unicode data.

In previous releases of Ingres, the DBA had the option when creating a database of defining a collation sequence to be used for non-Unicode text columns. The Unicode standard default collation sequence was provided for Unicode text columns. These collation sequences were in effect for all columns in the database and could not be changed without recreating the database.

Note: As part of this feature, the data descriptor used throughout the Ingres system changed. This data descriptor is also compiled into imaged ABF applications. After upgrading to Ingres 2006, all ABF applications should be re-imaged. Delete the contents of the ABF object directory, `$ING_ABFDIR/database-name/app-name`, and then re-image.

For details on this new feature, see the *SQL Reference Guide*.

System-wide Setting for Default Lock Level

This feature adds a system configuration parameter that allows the Ingres administrator to define the default lock level for the entire Ingres instance.

In previous releases of Ingres, to alter the default lock level the application programmer had to use the `SET LOCKMODE` statement in the application, which affected only the current session.

The new parameter, `system_lock_level`, is available to the administrator through Configuration-By-Forms. Valid values are `DEFAULT`, `ROW`, `PAGE`, and `TABLE`. The `DEFAULT` value is the default and allows the system to decide the lock level. `DEFAULT` is the assumed value if the parameter is not present.

Note: Each of the default lock levels is subject to escalation, as in previous releases.

For more information on the default locking level, see the *Database Administrator Guide*.

New Features for Application Developers

The new features for application developers provide additional flexibility in application design and make it easier to migrate existing applications that run against non-Ingres database architectures.

Automatic Sequence Number Generation

Ingres has a sequence facility that provides the ability, through SQL, to create a column in a table that contains a sequentially incremented number for each row.

Users can define a named sequence generator by using a CREATE SEQUENCE statement. The sequence generator can produce values in any context that requires a scalar value using the phrase next value for <sequence name> or <sequence name>.nextval. The sequence generator can be used in the values list of an INSERT statement, in the select list of a query, or anywhere that a scalar numeric value is required.

The sequence facility allows sequences to be defined as integer or decimal values. Ingres permits sequences to be decimal(31), which supports a range of +/- 10**32.

Use the ALTER SEQUENCE statement to change the parameter settings for a sequence generator and the DROP SEQUENCE statement to delete a sequence generator.

For more information, see the *SQL Reference Guide*.

No Wait for Lock Requests

Ingres uses a value of NO WAIT for the timeout parameter in the SET LOCKMODE statement to indicate that when a lock request that is made that cannot be granted without incurring a wait, control is immediately returned to the application that issued the request. NO WAIT applies to any lock in a transaction or a lock on one or more specific tables. For more information, see the *SQL Reference Guide* and *Database Administrator Guide*.

Support for New Data Types

Ingres 2006 supports the bigint and tinyint data types.

bigint

The bigint numeric data type stores 64-bit integers. The bigint data type is an implementation of the ANSI standard bigint. Integer8, Int8, and i8 are synonyms for this data type.

The int8 function converts the specified expression, which can be a c, char, varchar, nchar, nvarchar, text, float, money, decimal, integer1, smallint, to a 64-bit integer. Decimal and floating-point values are truncated. Numeric overflow occurs if the integer portion of a floating-point or decimal value is too large to be returned in the requested format.

tinyint

The tinyint numeric data type is a synonym for i1. Supported values are from -128 to +127.

For more information on new data types, see the *SQL Reference Guide*.

Additions to the Visual DBA Suite

Three new Visual DBA (VDBA) Suite tools have been added:

- Visual Database Objects Differences Analyzer
- Visual Configuration Differences Analyzer
- Export Assistant

Visual Database Objects Differences Analyzer

The new Visual Database Objects Differences Analyzer (VDDA) tool allows you to compare groups or individual Ingres database objects, either in the current installation or saved into a snapshot file. It also allows you to visualize the differences in the database objects.

The following options are supported:

- Performing the comparison either at the installation level (that is, comparing any database objects present in an installation) or at the schema level (that is, comparing objects owned by a given user within a database). In both cases, you can limit the comparison to certain object types.
- Saving such groups of database object definitions into snapshot files for later comparison with current or other saved database object definitions. This allows comparisons within an installation over time.

The list of differences generated by VDDA includes one line for each difference in a property of the given database object. The bottom status line indicates the number of objects with differences and the number of differences found in the comparison.

For more information, see online help for VDDA.

Visual Configuration Differences Analyzer

The new Visual Configuration Differences Analyzer (VCDA) tool allows you to compare the configuration information for two Ingres installations. VCDA enables you to take snapshots of the installation and compare these snapshots with either a snapshot taken at some point in the past or with a configuration snapshot taken on another machine.

For example, after you install Ingres and have tuned the configuration to meet your needs, take a snapshot of the configuration. If you encounter problems with the installation later on, take another snapshot of the configuration and compare it to the earlier snapshot to determine if any configuration changes have contributed to the problem. Keep an on-going record of configuration changes by taking a snapshot of the installation each time you change its configuration.

VCDA snapshots contain information taken from the config.dat file, the symbol table, and the vnode database, as well as environment variables set at the system and user level.

Specifically, VCDA allows you to:

- Save a snapshot of the current installation configuration into a file
- Compare two snapshot files, or the current installation with a saved snapshot
- Restore selective groups of configuration parameters from a saved snapshot

VCDA lists the differences with an associated icon that distinguishes between those parameters that are different and those that exist in one snapshot but not in the other.

VCDA uses information from the config.dat file, which typically contains parameters that apply to the local host name only. However, if you want to concatenate the contents of all your config.dat files into a single config.dat file to distribute across multiple environments, VCDA manages this situation as follows:

- The host name under which a snapshot is saved becomes part of the snapshot information.
- If VCDA detects that host names other than the snapshot host name are managed within the config.dat information of the snapshot, VCDA displays them.
- A host name mapping option is available in that situation, so that VCDA can compare the additional host names' configuration parameters. If this option is not used, the parameters are compared, including their host names (that is, only parameters that are identical for the same host name in the two snapshots are considered identical).

For more information, see online help for VCDA.

Export Assistant

The new Export Assistant complements the existing Import Assistant in both design and function. The Export Assistant is a wizard designed to simplify the task of exporting Ingres (or Enterprise Access) data into external files. Specifically, you can export data into the following file formats:

- .csv (and other delimiter formats)
- .xml
- .dbf
- fixed widths

The Export Assistant is accessible from the Start menu and from the Ingres Visual Manager and Visual DBA tools. You can also invoke the Export Assistant from the command line.

Connectivity Enhancements

The following new features provide connectivity enhancements.

Support for JDBC 3.0 API

Support for the JDBC 3.0 API includes three components: an installation server, a Java client driver, and an information utility. For more information, see the *Connectivity Guide*.

Data Access Server

The Data Access Server (DAS) runs as part of a standard Ingres installation. The DAS translates JDBC requests from the Ingres JDBC Driver into Ingres internal format and forwards the request to the appropriate DBMS Server. The DAS supports the same network protocols and port designations as the Communications Server.

The DAS also supports the new Ingres .NET Data Provider component that enables high-performance native .NET access to Ingres data sources and delivers Ingres data to the Microsoft .NET Framework.

Through the DAS, a JDBC client and Ingres .NET Data Provider have full access to Ingres, Enterprise Access, and EDBC databases. The DAS can also access database servers on remote machines using Ingres Net.

For more information, see the *Connectivity Guide*.

JDBC Driver

The Ingres JDBC Driver is a pure Java implementation of the JDBC 3.0 API released with the Sun Java 2 SDK, version 1.4. The driver supports application, applet, and servlet access to Ingres data sources through the Data Access Server.

The JDBC driver provided in Ingres 2.6 continues to be supported in Ingres 2006. For migration instructions related to the JDBC driver, see the *Migration Guide*.

The Ingres JDBC Driver with the DAS supports the following JDBC 3.0 features:

- Boolean data type (similar to Bit)
- Savepoints
- Named procedure parameters
- Auto-generated keys
- Connection Pool Configuration

The Ingres JDBC Driver is delivered as a single Java archive file, `ijjdbc.jar`, located in the library directory (`lib`) of the Ingres installation. Access to the driver can require, depending on the Java environment used, adding the Java archive to the `CLASSPATH` environment setting or as a resource in the appropriate utility. For browser/applet access, the Java archive must be copied to the Web Server directories.

JDBC Information Utility

The JDBC 3.0 API support includes a JDBC information utility, `JdbcInfo`. This utility displays the Ingres JDBC Driver internal release information. The class files for the `JdbcInfo` utility are located in the library directory (`lib`) of the Ingres installation.

For more information, see the *Connectivity Guide*.

Updateable Result Sets in JDBC

The Ingres JDBC driver supports updateable result set features of the JDBC 2.1 API. Updateable result sets permit an application to update or delete the current row of the result set, or insert rows into the associated table using methods provided by the JDBC `ResultSet` class. A new class, `RsltUpdt`, has been added as an extension to the cursor result set class, `RsltCurs`, to support updateable result sets. The result set methods associated with the `RsltUpdt` class are listed below.

The ability to update a result set is determined by calling the following method:

```
ResultSet.getConcurrency()
```

The current row of a result set can be deleted using the following method:

```
ResultSet.deleteRow()
```

Columns values of the current row can be set using the following methods:

```
ResultSet.updateAsciiStream()
```

```
ResultSet.updateBigDecimal()
```

```
ResultSet.updateBinaryStream()
```

```
ResultSet.updateBoolean()
```

```
ResultSet.updateByte()
```

```
ResultSet.updateBytes()
```

```
ResultSet.updateCharacterStream()
```

```
ResultSet.updateDate()
```

```
ResultSet.updateDouble()
```

```
ResultSet.updateFloat()
```

```
ResultSet.updateInt()
```

```
ResultSet.updateLong()  
ResultSet.updateNull()  
ResultSet.updateObject()  
ResultSet.updateShort()  
ResultSet.updateString()  
ResultSet.updateTime()  
ResultSet.updateTimestamp()
```

Once column values have been set, the changes can be saved or dropped using the following methods:

```
ResultSet.updateRow()  
ResultSet.cancelRowUpdates()
```

To insert a row, the result set current position must be moved to a special reserved row. The following methods control the positioning of the result set and the insertion of rows:

```
ResultSet.moveToInsertRow()  
ResultSet.moveToCurrentRow()  
ResultSet.insertRow()
```

The following methods can be used to determine the status of a result set row:

```
ResultSet.rowDeleted()  
ResultSet.rowInserted()  
ResultSet.rowUpdated()
```

For more information, see the *Connectivity Guide*.

.NET Data Provider and Visual Studio .NET Integration

This release of Ingres introduces support for the Microsoft .NET Framework and Visual Studio .NET application development tools.

Ingres .NET Data Provider

The Ingres .NET Data Provider is a .NET component that enables high-performance native .NET access to Ingres data sources and delivers Ingres data to the Microsoft .NET Framework.

The Ingres .NET Data Provider offers a series of .NET types to describe the user's data, .NET provider classes to manipulate the data, and connection pooling to efficiently manage data connections.

The design and naming conventions of the Ingres .NET Data Provider's data types, classes, properties, and methods follow the same pattern as the Microsoft .NET Data Providers. Consequently, developers who are familiar with the Microsoft providers can easily develop or convert existing code from Microsoft databases to Ingres databases.

All Ingres .NET Data Provider modules are written in C#, a managed .NET language with full access to every .NET Framework capability. Even though the data provider is written in C#, any managed language such as VB.NET or J# can use the data provider because of .NET's language interoperability feature.

For more information, see the *Connectivity Guide*.

Visual Studio .NET Integration

The .NET Framework was written with design-time support. Integration with the Visual Studio .NET visual tools allows programmers to drag-and-drop the Ingres .NET Data Provider design component onto a design surface such as a the Windows Form Control (WinForm). Integration also includes the following design components:

- Data Adapter Configuration Wizard: Enables programmers to specify the design properties of the Ingres DataAdapter object.
- Query Builder: Enables programmers to build SQL statements that the Ingres .NET Data Provider uses to retrieve and modify database information.
- Parameter Collection Editor: Enables programmers to add parameters to the Command component.

For more information, see the *Connectivity Guide*.

Ingres ODBC Administrator

Prior to this release, ODBC users on non-Windows platforms were required to manually edit the `odbc.ini` configuration file to define a data source. This release introduces an Ingres ODBC Administrator utility for non-Windows platforms. Supported platforms include UNIX, Linux, and VMS. The new ODBC Administrator enables users to:

- Create, edit, and delete data source definitions
- View configuration details about a particular data source
- Display a list of installed drivers and view configuration details for a selected driver
- Define an alternate path for accessing driver definitions
- Define an alternate path for accessing data source definitions
- Turn ODBC tracing on or off
- Test a data source connection

For more information, see the *Connectivity Guide*.

WinSock 2.2 API TCP/IP Protocol Driver for Windows

This release provides a new Windows TCP/IP protocol driver that takes advantage of the latest Windows Winsock 2.2 API. The new driver (`tcp_ip`) removes an architectural limitation of the previous implementation (`wintcp`), which resulted in a performance problem when a client application used many INSERT statements or a large number of single SELECT statements. Also, connection attempts sometimes failed when multiple connects were attempted simultaneously.

To allow a smooth migration from the existing protocol driver to the new one, the existing `wintcp` protocol driver is included in this release, but will be removed in the future. We recommend that you use the `tcp_ip` protocol driver.

For more information, see the *Connectivity Guide*.

ODBC Call-level Interface

The Ingres ODBC Call-level Interface (CLI) provides access to the ODBC application environment without the need to use third-party software. It is installed when you install the Ingres ODBC Driver and is supported on all platforms on which Ingres runs.

The Ingres ODBC CLI performs the following functions:

- Optionally determines driver characteristics from ODBC configuration files
- Loads and unloads the ODBC driver into and from application memory
- Maps the driver manager API to the driver API
- Performs basic error checking
- Provides thread safety
- Provides ODBC tracing
- Provides function templates, type definitions, and constant definitions for ODBC applications

For more information, see the *Connectivity Guide* and the *System Administrator Guide*.

New Features for Linux

The following features are new for the Linux platform.

RPM Packaging

Ingres 2006 for Linux is packaged using the Red Hat Package Manager.

KDE/GNOME Desktop Integration

When Ingres 2006 is installed on Linux, it installs the Ingres for Linux *Getting Started* guide on the desktop. It also creates two program groups, one for the Visual DBA suite and the second for the complete set of Ingres documentation. The user has the information needed to start and use Ingres within minutes.

Linux Cluster Support

For information about the Ingres Cluster Solution on Linux, see Ingres Cluster Solution for Linux (see page 35).

Changes to Existing Features

This section describes changes made in Ingres 2006 to existing features.

Enhanced Unicode Support

Unicode is a standard method of storing character data for multinational situations. Many applications require the ability to store and retrieve Unicode data. Ingres 2006 supports the second phase of a phased implementation of full Unicode support and extends support to the UTF-16 encoding scheme.

For details on this feature, see the *SQL Reference Guide*, the *OpenSQL Reference Guide*, and the *Command Reference Guide*.

Unicode Coercion

Ingres coerces between Unicode data types and non-Unicode data types, including nchar/nvarchar and char/varchar types. Unicode coercion includes the following features:

- Unicode and non-Unicode columns can be joined together in a query.
- The Ingres COPY statement (and by inference, COPYDB and UNLOADDB) can be executed to generate or load non-binary data files where a table contains at least one Unicode column.
- Query strings containing non-Unicode literals that reference Unicode columns are coerced to Unicode before the DBMS executes the query.

Collation Sensitive Support for Wildcard Searching

Ingres performs wildcard searches involving Unicode strings by making use of a case insensitive collation table.

Complex Query Optimization

The Ingres query optimizer has been enhanced to handle queries that reference large numbers of tables and indexes with shorter response time. For more information, see the *SQL Reference Guide*.

Increased Range Table Limit

The limit for the number of table references in a query has increased from 30 to 126.

This limit refers to the sum of explicit table references (even if the same table appears several times in a query), explicit view references, and tables and views included by the expansion of a view definition. It also refers to the sum of such references in all subselects and unioned selects of a query.

For more information, see the *SQL Reference Guide*.

JDBC User ID Enhancements

In previous releases of Ingres, the JDBC driver required a user ID and password that was valid in the context of the DBMS platform, even for local connections where the local user ID is sufficient. The JDBC driver also failed to use virtual node (vnode) login information when making remote connections, and was not able to access private vnode definitions.

In Ingres 2006, the JDBC driver no longer requires a user ID and password when the Data Access Server is running on the same platform as the Java client. Instead, for local connections, the local user ID is sufficient to establish the DBMS Server connection.

When the Data Access Server and DBMS Server are on separate platforms, a vnode is required in the target database specification that defines how the connection between the Data Access Server and DBMS Server is to be established. The vnode provides the connection information while the JDBC user ID and password are used to access the remote DBMS Server.

A new driver connection property/URL attribute allows the JDBC application to control the context (Data Access Server (local) or DBMS Server (remote)) in which the user ID is used.

When used in the Data Access Server context, the user ID and password allow access to the private vnode information for the user ID provided, and both the login and connection information from the vnode is used to access the remote DBMS Server.

When used in the DBMS Server context, global vnode definitions are used for (nonsensitive) connection information and the provided user ID and password are used to access the remote DBMS Server.

For more information, see the *Connectivity Guide*.

Increased Column Limit

The limit on the number of columns per table has increased from 300 to 1024. This feature is upwardly compatible; however, programs written to take advantage of the new limit cannot be used with earlier Ingres releases that included the 300-column limit. For more information, see the *SQL Reference Guide*.

VDBA Enhancements

The following VDBA enhancements have been implemented for Ingres 2006. These enhancements are available only on Windows, Solaris, HP-UX, and AIX platforms.

VDBA Architecture Split

Two VDBA utilities, SQL/Test and Performance Monitor, have been split from the VDBA architecture and are now called Visual SQL and Visual Performance Monitor. By isolating these components as smaller, stand-alone executables, users benefit from increased response time because they no longer need to launch the whole VDBA executable.

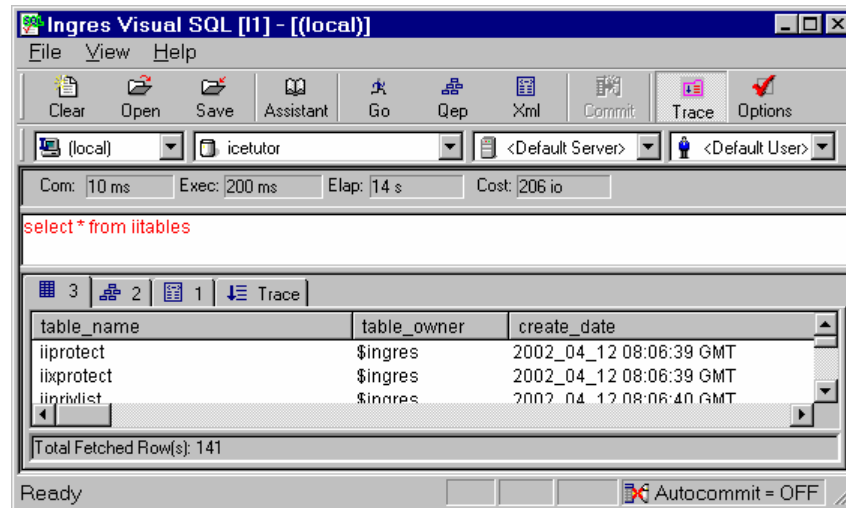
When launched in VDBA, these new components also help avoid VDBA locking itself. This can happen, for example, when there is an uncommitted query on a table in an SQL Test window, and an operation is performed on the same table in a DOM window opened in the same VDBA instance.

The SQL Assistant has also been split into a stand-alone DOM component. The SQL Assistant can now be accessed from the Ingres Export Assistant, in addition to Ingres Network Utility and VDBA, to help build SQL queries.

Visual SQL and Visual Performance Monitor are accessible from the Start menu, Ingres Visual Manager, Network Utility, VDBA, and the command line.

Visual SQL

The new stand-alone executable (vdbasql) for Visual SQL displays the following window.



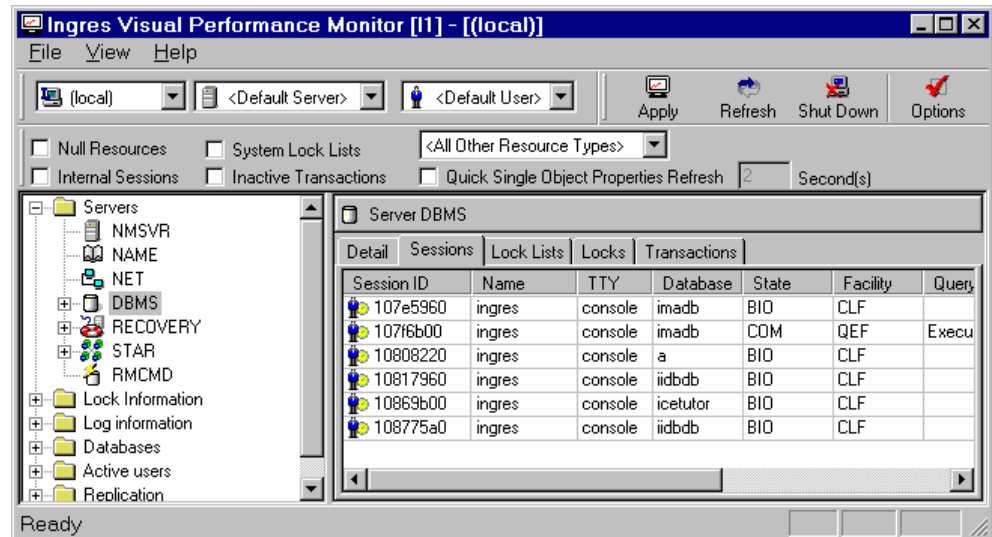
This window allows all the functionality of the previous SQL/Test window in VDBA, but has the following additional toolbar controls:

- New combo-boxes for choosing a node, server class, and impersonated user. (The equivalent function is provided in VDBA through the branch chosen in the nodes tree when launching an SQL/Test window.)
- A Commit button that allows you to directly commit the current transaction without the need to type commit in the query area.
- An Options button that provides access to the properties of the SQL Test control. (This is equivalent to the SQL/Test Preferences in previous VDBA releases, plus the session timeout parameter that was global to VDBA.)

In addition, the status bar now provides the autocommit state of the transaction.

Visual Performance Monitor

The new stand-alone executable (vdbamon) for Visual Performance Monitor displays the following window.



This window allows all the functionality of the previous Performance Monitor window in VDBA, but has the following additional toolbar controls:

- New combo-boxes for choosing a node, server class, and impersonated user. (The equivalent function is provided in VDBA through the branch chosen in the nodes tree when launching a Performance Monitor window.)
- An Apply button that displays the monitor information corresponding to the chosen node (and, if applicable, server class and impersonated user).
- An Options button that provides access to the properties of the Performance Monitor control. (This is equivalent to the Monitor Preferences in previous VDBA releases, plus the background refresh preference, the Grid In List option, and the session timeout and max number of sessions parameters that were global to VDBA.)

Additional Properties

The Properties dialog is accessible from the new Visual SQL and Visual Performance Monitor executables, as well as from VDBA. (This was previously called the Preferences dialog.)

The SQL/Test Priorities dialog now includes a session timeout parameter that was previously global to all VDBA sessions.

The Performance Monitor Properties dialog includes the following additional parameters:

- Background refresh for monitor windows (includes the choice of having background refresh active or not)
- Session timeout and the maximum number of sessions used within the Performance Monitor control's internal session cache
- Grid In List option that displays a grid in the Detail Information pane that allows for easier viewing of data

In addition, Visual SQL and Visual Performance Monitor provide a new menu option, Save Preferences As Default. If this option is selected, which is the installation default, the properties are permanently stored and used every time the utility is executed. If this option is unselected, the selected properties are only used for the utility's current instance.

Notification of Metadata Changes

Notification of changes to Ingres metadata includes the following features in VDBA:

- VDBA is notified by the DBMS of any metadata changes and automatically refreshes itself.
- The refresh preferences are no longer available.
- A new preference (called other servers) provides background refresh for those servers that do not provide metadata change notification (for example, Ingres DBMS with releases older than Ingres 2006 and gateway servers).
- The background refresh of monitor data and node definitions are placed in the Monitor and Nodes preferences.

For more information, see the online help for VDBA.

Ingres Visual Manager Enhancements

The Ingres Visual Manager contains the following enhancements:

- Direct access to all Ingres tools
- Can register alerts in the OS event log
- Message explanations

Note: These enhancements are only available on Windows, Solaris, HP-UX, and AIX platforms.

Direct Access to All Ingres Tools

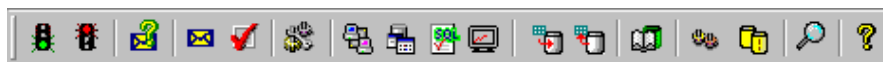
Ingres Visual Manager provides direct access to all Ingres visual tools. The only exception is the Ingres Service Manager; its full functionality is already provided by Ingres Visual Manager. You can access these tools in the following ways:

- Right-click the tray toolbar to display a menu:



The Ingres 2006 documentation is accessible from the menu.

- Select a tool from the IVM toolbar.



Only the most frequently used tools are accessible from this toolbar. Use the above menu to access all other tools.

Registering Alerts in the OS Event Log

When an Alert message is written to the Ingres errlog.log file, IVM indicates the alert through a special icon change in the tray toolbar and in the IVM window tree. IVM also lets you set additional preferences for alert notification using the Preferences dialog. These additional preferences include:

- **Sound** – When selected, you are alerted of an event by a sound (beep).
- **Message Box** – When selected, you are alerted of an event by a message box if there are unread Alert messages.

A new preference has been added to the Preferences dialog:

- **OS Event** (for new Events only) – When selected, the full text of all Alert messages resulting from new events are logged in the Operating System [Application] Event log in addition to the Ingres errlog.log file. This new preference allows third-party tools to be signaled by Ingres Alert messages.

In addition, if the operating system supports it (as with Windows), the message category (class) and number is stored as information belonging to the event. External applications that are monitoring error messages for the Ingres installation are given the error category and number without having to parse the text of the message.

If the OS Event option is selected, you can also set preferences for generating the following specific operating system events:

- One event is generated whenever the OS Event option checkbox has been selected or deselected and the Preferences dialog has been validated.
- One or more events are generated when the Alert States have been changed (for example, from “alert” to “non-alert”) in the Messages Categories and Notification Levels dialog.
- One event is generated if IVM detects that the errlog.log file has been manually changed.

Viewing Message Explanations

To help you respond to Ingres messages, IVM now displays an explanation for any message in the errlog.log file.

Select the desired message on any page or window in IVM that allows message selections and view its explanation by clicking the Message Explanation toolbar button.

Configuration Rules System Enhancements

The Configuration Rules File System has been modified to handle negative values and decimal values.

Shadow Copy of the Symbol Table

When setting, unsetting, or changing environment variables, a backup of the symbol.tbl file (symbol.bak) is maintained. If the original symbol.tbl file becomes corrupted, you can use the backup symbol table file to restore it. A history of updates to the symbol.tbl file is maintained in the symbol.log file. The symbol.bak and symbol.log files are located in the same location as the original symbol.tbl file. For more information, see the *System Administrator Guide*.

Additional Join Functionality

In addition to inner, left, right, and full joins, users can request cross joins (effectively inner joins without an clause) and natural joins. Also, the ON clause can be replaced by a USING clause that contains a list of columns, each of which appears in both tables being joined. Instead of the explicitly coded join qualification of the ON clause, the USING clause applies one equijoin predicate for each column pair in the list of columns. For example, "... a left join b using c1, c2 ..." is identical to coding "... a left join b on a.c1 = b.c1 and a.c2 = b.c2" For more information, see the *SQL Reference Guide*.

Improved Out-of-the-Box Configuration Defaults

Ingres configuration defaults have been updated to reflect the current hardware environments.

Improved IMA Support

Ingres 2006 introduces an additional Ingres Management Architecture (IMA) component, IMP.

IMA provides the framework for accessing system data for monitoring and managing installations through SQL without affecting the underlying operation of the product. For more information, see the *System Administrator Guide*.

CREATEDB Enhancements

Ingres 2006 adds a flag to CREATEDB, which allows the specification of a non-default page size for catalogs. For more information, see the *Command Reference Guide*.

ALTERDB Enhancements

Ingres 2006 adds three new flags to the ALTERDB utility, which allow for the deletion of invalid checkpoints, the deletion of a specific checkpoint, and a non-Unicode enabled database to be Unicode enabled. For more information, see the *Command Reference Guide* and the *Database Administrator Guide*.

Terminal Monitor Enhancements

Ingres 2006 adds support for the `-p` flag to the terminal monitor (that allows a password to be specified) and support for the `-r` flag (that allows a role name and optional role password to be added).

Also, command completion and command history recall are supported for the Linux operating system.

For more information, see the *SQL Reference Guide* and the *Command Reference Guide*.

Enhancements for Log Full

Ingres 2006 allows the user to dictate behavior when a log full situation occurs. The options provided are COMMIT, ABORT, or CONTINUE. For more information, see the *SQL Reference Guide*.

Extended B-tree Limits

Ingres 2006 extends the 440-byte limit on the maximum width of a B-tree key.

Installer Enhancements

Ingres on Windows environments is delivered in Microsoft Windows Installer format. Microsoft Windows Installer version 2.0 is required on the machine on which you are installing Ingres components.

The installer on Windows presents a setup wizard to guide you through the installation process. The setup wizard lets you select a Complete or Custom install. The Custom installation lets you choose individual components for installation.

Installation as a User Other Than ingres

With Ingres 2006, the product can be installed as a user other than ingres. Ingres security is retained. When installing Ingres, you can specify a user ID (and its associated group ID) that owns the Ingres installation. During installation, this system administrator ID is automatically created and given the required permissions. If they are not previously defined, the user ID and group ID are added to your UNIX or Windows systems.

Supportability Enhancements

Product enhancements that improve the supportability of Ingres include dumping queries to the error log file upon certain error conditions (for example, an optimizer time-out or an exhaustion of resources), and modifying the ingstart process so that the exact version and patch information is written to the error log file on startup.

Help System Enhancements

Each Ingres visual tool now has its own Help system that is independent of other visual tools. This separation of Help systems makes it easier and quicker for users to navigate through the list of Help system topics. In addition, search results now display only those topics that are applicable to the visual tool in use. Standalone Help systems are now provided for the following visual tools:

- VDBA
- Configuration Manager
- Ingres Visual Manager
- Import Assistant
- Export Assistant
- Journal Analyzer
- Network Utility
- Visual SQL
- Visual Performance Monitor
- Visual Database Objects Differences Analyzer
- Visual Configuration Differences Analyzer

Index

A

Alterdb • 57
ANSI date and time • 18

B

BEFORE triggers • 18
b-tree • 57

C

cluster • 35
collation • 36, 48
columns • 50
configuration • 9, 56
configuration rules • 56
connection pooling • 14
createdb • 57
cursor definition text • 12

D

Data Access Server (DAS) • 32, 42
data types • 18, 38
date functions • 20
decimal data type • 11
derived tables • 17
describe input statement • 19

E

export assistant • 41

H

help system • 59

I

incremental rollforwarddb • 11
Ingres .NET Data Provider • 15, 23, 32, 44, 45
Ingres Management Architecture (IMA) • 56
installer • 32, 58
integer/string coercion • 20
IPv6 • 24

J

JDBC • 23, 32, 41, 42, 43, 49
joins • 56

L

LOB locators • 7, 13
lock
 level, default • 36
 requests • 37
 Terminal Monitor • 57
 unextenddb • 35
log full • 57

M

message explanations • 55
metadata changes • 53
modify, online • 35

N

numeric overflow • 35

O

ODBC • 22, 46, 47
ODBC Call-level Interface • 47
open source • 33, 34
optimization • 48

P

parallel query • 34

Q

query
 alerts • 55
 killing • 35
 parallel • 34
 PHP • 23
 table references • 49

R

RPM • 47

S

scrollable cursors • 7, 14
sequence defaults • 19
sequence number generator • 37
server management • 21
SQL functions • 10, 20

supportability • 31, 58
symbol table • 56

T

table
 default storage structure • 9
 references • 49
 storage structure • 9
 temporary, indexes • 19
 temporary, referencing • 19
TCP/IP • 24, 46

U

Unicode • 36, 48
user • 58
UTF-8 character set • 8

V

Visual DBA • 38, 39, 40, 50, 51
Visual Studio • 45
visual tools • 39, 40, 52, 54