CA IDMS™ Server r17

CA IDMS Server helps enable secure, open access to CA IDMS™ mainframe data and applications from the Web, Web services, PCs and other distributed platforms. It implements an optimized, high-performance architecture with JDBC and ODBC database drivers that provide industry-standard application programming interfaces. CA IDMS Server is an ideal tool for modernization, Web access, and integration of critical CA IDMS assets into a service-oriented architecture (SOA).

### Business Value

Organizations with significant investments in business-critical mainframe systems are deploying Web-based solutions for improved customer service and to provide a competitive edge. CA IDMS Server provides a cost-effective and flexible approach to modernization because there is no need to move or copy CA IDMS data to other environments, to redefine the data, or to rewrite existing systems in order to open these mainframe assets for use with Web applications and SOA participation.

### Product Overview

CA IDMS Server couples standards-based Web access with proven technologies to help your organization achieve the objectives of secure, optimized, high-performance access to CA IDMS mainframe resources without the high costs and risks associated with platform migrations. With CA IDMS Server, new systems can access existing CA IDMS enterprise resources and seamlessly deliver them to modern applications.

### Delivery Approach

CA Services can help you realize the benefits from your CA IDMS investments by using CA IDMS Server to modernize your CA IDMS mainframe assets without the expensive costs and high risks of a conversion to another environment. CA also offers technical webcasts and webcast replays on CA IDMS topics including CA IDMS Server for modernization. See the CA IDMS pages on ca.com and support.ca.com.
What’s New, What’s Compelling in CA IDMS Server r17

CA IDMS ODBC Driver Enhancements

WIRE PROTOCOL A commonly requested feature, the ODBC driver in CA IDMS Server r17 supports wire protocol communications directly from the Windows operating system to the CA IDMS Central Version address space. The wire protocol uses TCP/IP and improves performance, scalability, and stability when used with CA IDMS on z/OS and z/VSE. This feature also enables use of the ODBC driver with CA IDMS on z/VM.

ODBC 3.5 SUPPORT The CA IDMS Server ODBC driver now supports the current ODBC 3.5 specification standard for improved compatibility with customer applications and third-party tools.

ODBC 64-BIT SUPPORT The ODBC driver includes native support for x64-compatible 64-bit processors, eliminating the need to use the Windows on Windows component (WOW) on a 64-bit machine.

NAMED PARAMETERS Similar to JDBC, the ODBC driver supports the use of named parameters in CALL statements that are included in the standard escape syntax, allowing applications to set and get parameters by name.

JDBC Driver Enhancements

JDBC 4.0 SUPPORT The JDBC driver now supports the current JDBC 4.0 specification for improved connection management and compatibility with customer applications and third-party tools.

SCROLLABLE RESULT SETS The JDBC driver supports scrollable result sets by caching rows as they are fetched from the database. This feature enables the use of standard Java RowSet implementations that use disconnected row sets, updateable row sets, and other advanced features.

NAMED PARAMETERS Similar to ODBC, the JDBC driver supports the use of named parameters in CALL statements that are included in the standard escape syntax, allowing applications to set and get parameters by name rather than by using ordinal numbers.

HIBERNATE DIALECT CA IDMS Server r17 includes a Hibernate dialect specifically designed for CA IDMS SQL that enables easier Java application development. Hibernate provides object-relational mapping for Java objects persistence in relational databases, so Java developers are free to concentrate on the business logic of their application.

WIRE PROTOCOL AND 64-BIT SUPPORT Please note that CA IDMS Server type 4 JDBC driver already provides wire protocol support and 64-bit support.

CA IDMS r17 Support

ENHANCED ANSI/ISO JOIN SUPPORT The ODBC and JDBC drivers support CA IDMS r17 ANSI/ISO JOIN enhancements in the ODBC and JDBC metadata functions when connected to a CA IDMS r17 or later system.

RESULT SETS The ODBC and JDBC drivers support the use of standard APIs to return multiple result sets from a procedure call when connected to a CA IDMS r17 system.

SYSCA.SINGLETION_NULL The ODBC and JDBC drivers now use the SYSCA.SINGLETION_NULL pseudo table for metadata APIs that issue SELECT commands on a dummy table to invoke scalar functions or check the status of the connection, preventing authorization errors.
Other New Features

**HTML HELP SYSTEM** CA IDMS Server r17 supports online HTML help, replacing WinHelp.

**LOG FILE ROLLOVER** The CA IDMS Server Windows- and Java-logging facilities now support a user-specified maximum log file size. When the log file reaches this size, a new log file is started, a user-specified number of previous files are saved, and older files deleted to prevent consumption of too much disk space.

**PSEUDO CONVERSE OPTIONS** The CA IDMS Server ODBC Administrator allows setting options that control pseudo-conversational processing for ODBC driver for an individual data source or all data sources. Both the ODBC and JDBC drivers support a new option that disables pseudo-conversational processing completely to reduce CA IDMS task overhead for high volume batch-type applications.

Features

**MODERNIZATION WITH OFF-THE-SHELF TOOLS** CA IDMS Server enables modernization with popular SQL-based Web and GUI tools for query, reporting, and development environments. Modernization using CA IDMS Server allows developers to work with CA IDMS just like any enterprise resource, using modern languages and tools that they already know. The standards-based approach of CA IDMS Server and CA IDMS SQL enable data warehousing, composite applications, enterprise reporting, Web services, and user-written applications to access and update CA IDMS mainframe resources in order to fulfill business needs that require availability of up-to-date operational data.

**WEB SERVER, APPLICATION SERVER, AND SOA FLEXIBILITY** CA IDMS Server is designed to work with popular Web servers and application servers to provide XML and Web services access to CA IDMS mainframe resources to meet changing business requirements. Applications using CA IDMS Server can be deployed on industry-standard J2EE or .NET environments, including IBM WebSphere, BEA WebLogic, JBoss Application Server, Apache Tomcat, and Microsoft IIS for scalable Web access to CA IDMS data and business logic.

**OPEN ACCESS** CA IDMS Server enables applications written in Java to access CA IDMS resources using the JDBC API. CA IDMS Server provides access to CA IDMS resources from Windows-based applications using the ODBC API. Microsoft .NET applications can access CA IDMS through the CA IDMS Server ODBC driver and the ODBC.NET provider. The JDBC and ODBC drivers implement access to CA IDMS using dynamic SQL requests from the client application. Since many Web and GUI-based tools produce dynamic SQL statements to drive database access, you can take advantage of this standards-based API without requiring intimate SQL knowledge. You can also use the dynamic SQL interface in CA IDMS Server to call remote stored procedures that provide complex database navigation or execute application business logic on the mainframe.

- **ODBC** The ODBC driver conforms to ODBC 3.5 and is a wire protocol driver that communicates directly from an ODBC Windows-based client to the CA IDMS address space via TCP/IP.

- **JDBC** CA IDMS Server implements all required interfaces to meet the JDBC 4.0 API specification and is also compatible with earlier JDBC versions. Type 2, Type 3, and Type 4 JDBC driver support is provided. The Type 2 driver supports Java application access from the Windows and z/OS environments. The Type 3 driver supports Java access from platforms including the Windows, UNIX, and Linux operating systems through communication with a JDBC server component running in a Windows or z/OS environment. The Type 4 JDBC driver is a wire protocol driver that allows you
to tailor options for individual connections and provides communications directly from a pure Java client to the CA IDMS address space with no intermediate server or middleware required.

• J2EE For J2EE application servers, the JDBC driver supports container-managed connection pooling to provide optimal performance for high-volume application server applications. The Java Naming and Directory Interface (JNDI) can be used with CA IDMS Server to allow applications to easily connect with a CA IDMS resource using only a logical data source name.

• Multiple Operating System Support CA IDMS Server provides access to CA IDMS on the z/OS, z/VSE, and z/VM operating systems from 32-bit and 64-bit Windows-based clients and Java client platforms.

SECURITY AND COMPLIANCE Security and encryption across network communications are key requirements for serious distributed access and application development. The CA IDMS Server JDBC and ODBC drivers can utilize Secure Sockets Layer (SSL) support for secure communications over TCP/IP. SSL support is available for the z/OS operating system only. CA IDMS Server also provides external identity auditing via CA SiteMinder™ Web Access Manager. This feature provides an end-to-end audit trail of external user identities that update data in CA IDMS databases, addressing internal security, auditing, and compliance requirements. The end user identities are recorded in the CA IDMS journals in the same way that the CA IDMS user id is recorded for mainframe transactions. This feature is optimized to support external identity auditing for J2EE application server-based systems managed by CA SiteMinder Web Access Manager and can also be used with other compatible identity managers and standalone Java applications.

SQL CALL PROCEDURE SUPPORT CA IDMS Server exploits the CA IDMS SQL Call Procedure feature to facilitate reuse of existing CA IDMS mainframe business logic. Call Procedure provides a standardized way of invoking remote procedures that run in the CA IDMS address space, including CA ADS™ mapless dialogs. Applications can also access the SQL catalog to discover and retrieve metadata information about procedures.

POSITIONED UPDATES CA IDMS Server provides efficient positioned update operations for top application performance. When an application specifies the FOR UPDATE clause on an SQL query statement and sets a cursor name, it can execute SQL INSERT, UPDATE, and DELETE statements with the WHERE CURRENT OF cursor-name clause.

DISTRIBUTED TRANSACTIONS AND COMPOSITE APPLICATIONS CA IDMS Server supports distributed transactions and composite applications that coordinate processing with other non-CA IDMS enterprise resources. Through the JDBC driver, a Java application can use the interfaces and methods defined by the Java Transaction API protocol (JTA) to commit or rollback CA IDMS updates in a distributed transaction. JTA is a Java mapping of the industry-standard X/Open XA protocol, and works with J2EE Compliant Transaction Managers. Alternatively, a Java application can be deployed in a J2EE application server, and the application server can manage participation in a distributed transaction.

MULTITHREADED DATABASE ACCESS CA IDMS Server delivers multiple thread support in the JDBC and ODBC drivers and the TCP/IP communication components of the product. This provides high performance for applications that use multiple threads for database access by supporting concurrent access to CA IDMS mainframe data for each thread.

HIGH PERFORMANCE The ODBC and JDBC drivers support data conversions on the client which can reduce CPU utilization on the server. Applications can specify the number of rows requested on each internal BULK FETCH in order to tune performance based on the characteristics of the expected
result set. CA IDMS Server supports DESCRIBE INPUT for prepared statements and delayed parameter binding, allowing input parameters to be changed when a prepared statement is re-executed. The batch update feature provides J2EE compatibility by enabling an application to submit multiple SQL DML or DDL commands for execution in a single request. The JDBC driver caches these commands and processes them in the order they were received. This feature also supports the CA IDMS bulk INSERT feature for improved performance.

**IPV6** The next generation of the Internet protocol, IPv6, provides 128-bit address support for greatly expanded addressing and other improved capabilities. CA IDMS Server transparently supports the use of either IPv4 or IPv6 protocols in both JDBC and ODBC drivers. IPv6 support is available only for the z/OS operating system.

**PRIMARY AND FOREIGN KEY METADATA SUPPORT** Primary and foreign key metadata support enables CA IDMS Server to use standard APIs that discover the relationships between CA IDMS SQL tables and the relationships between CA IDMS network records based on their primary and foreign key definitions. It offers the opportunity for using more complete and standard SQL functionality with CA IDMS network databases and reduces or eliminates reliance on table procedures. This feature is useful for software tools that need to know CA IDMS database definitions, such as development tools, business intelligence (BI), and data warehousing (DW) tools.

**EASE OF USE** The JDBC driver and server support options and tracing on all platforms without requiring the installation of any native libraries. The JDBC driver also supports delayed parameter binding, which allows parameters to be set at any time before a statement is executed to improve compatibility with third-party applications.

**UNICODE AND CUSTOMIZED DATA TRANSLATION OPTIONS** CA IDMS Server translates character data transferred between the client platform and the host. The JDBC driver converts the data from the native platform encoding to Unicode for use in the Java application. By default, the ODBC driver translates EBCDIC character data on the host to ASCII on the PC using U.S. English code pages. You can override the default language by creating a customized translation table that uses the Country Extended Code Pages for alternate languages. You can also enable double-byte character set processing.

**Benefits**

**CA IDMS Server Enables Modernization While Protecting Investments**

Your core CA IDMS systems have probably been in place for years and now you may be evaluating how to maximize the business value in these systems going forward. How do you modernize your CA IDMS assets to enable Web-based applications and Web services that tie into the CA IDMS data and logic?

There are many approaches to these challenges. The smart alternative is to leverage your CA IDMS databases and applications as enterprise resources and using CA IDMS Server to modernize without the risks of application or platform migrations. CA IDMS enterprise resources can be accessed and used directly by Web applications and can also be encapsulated as services and effectively reused in new multi-tier applications.
Supported Environments

MAINFRAME REQUIREMENTS CA IDMS Server r17 requires CA IDMS r16 or r17 and CA IDMS SQL r16 or r17, except as noted below.

CLIENT REQUIREMENTS Windows Server 2003, 32- or 64-bit edition; Windows Server 2008, 32- or 64-bit edition; Windows XP SP2 or later, 32- or 64-bit edition; Windows Vista, 32- or 64-bit edition; Windows 7, 32- or 64-bit edition. Platforms must support Java 1.6 or later.

SPECIFIC REQUIREMENTS

• The type 4 JDBC driver requires CA IDMS r16 SP2 or later and is available for the z/OS, z/VSE, and z/VM operating systems.
• The wire protocol ODBC driver requires CA IDMS r17.
• SSL support is available on the z/OS operating system only.
• SSL for the type 4 JDBC driver and wire protocol ODBC driver requires z/OS V1R7 or later with the Application Transparent Transport Layer Security (AT-TLS) feature enabled.
• IPv6 support is available on the z/OS operating system only.
• External Identity Auditing is supported for the type 4 JDBC driver with CA IDMS r16 SP4 and later and the wire protocol ODBC driver and CA IDMS r17.

Why CA

Making CA IDMS data and applications accessible to Web applications is a job that CA IDMS Server has been doing for almost as long as Web and application servers have been available. Today CA IDMS sites around the world are using CA IDMS Server in conjunction with popular application servers such as IBM WebSphere, Oracle BEA Weblogic, and JBoss Application Server to deliver CA IDMS data and applications to modern Web-based solutions. A platform-neutral solution based on industry-standard APIs, CA IDMS Server enables access to your mainframe CA IDMS systems in place and at a fraction of the cost compared to projects that require expensive and risky migration to other platforms. CA IDMS Server simplifies integration of CA IDMS assets with modern IT environments, enabling you to add new Web and distributed applications without any disruption to existing databases and applications and without significant new mainframe resources.

CA Modernization—Economical and Without Risk

To learn more and see how CA IDMS solutions enable other organizations to unify and simplify their modernization projects for better business results, visit ca.com/idms.