

IBM WebSphere Information Integrator Classic Federation for z/OS in an IBM Information Management System (IMS) Environment



Overview

WebSphere Information Integrator Classic Federation for z/OS is designed to provide Windows, Unix and Linux tools and applications with direct, real-time SQL access to mainframe databases, files and some mainframe programs. Tools and applications issue JDBC or ODBC SQL commands to read and write data stored in VSAM and sequential files, as well as IMS DB, CA-IDMS, CA-Datcom, Adabas and DB2 UDB for z/OS databases. SELECT, INSERT, UPDATE, DELETE and procedure CALL are supported.

Processing includes:

- accept and validate SQL statements from a server, client or desktop tool or application
- communicate SQL and result sets between distributed tools and applications and mainframe data platform(s)
- access the appropriate data using all available native file and database access aids such as indexes and keys
- translate results into a consistent relational format regardless of source data type

WebSphere II Classic Federation's ODBC and JDBC drivers interact with the distributed desktop tools and applications that are issuing SQL commands. These could be a simple query tool like MS Query, a complex business intelligence solution like Ascential or Business Objects or an e-business solution like WebSphere.

The SQL is delivered by the drivers to WebSphere II Classic Federation's z/OS-based data server whose sub components read-from and write-to the legacy databases and files using native database I/O commands. TCP/IP or MQ Series communication protocols connect the two.

Metadata

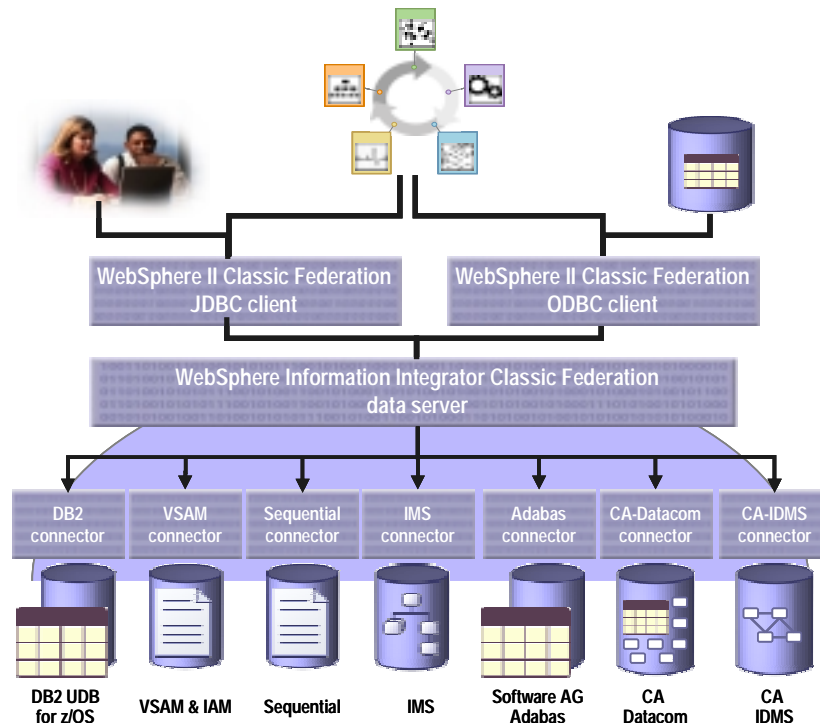
To process an SQL data access request for a prerelational data source such as IMS DB requires a mapping between the physical segment layouts and one or more logical relational tables. These logical tables also contain information on the underlying database hierarchical structures so that the operational

components can efficiently navigate the database.

IMS DB database definitions (DBDS) along with COBOL copybooks are utilized by the WebSphere II Classic Federation metadata management tools and utilities to create these required "logical relational table definitions. Logical table definitions as well as their underlying IMS DB database information are stored on the mainframe in the WebSphere II Classic Federation metadata catalog. The metadata catalog simulates a relational database catalog's tables for the pre-relational data.

The WebSphere II Classic Federation data server responds to SQL catalog queries using the metadata catalog's definitions. The end-user or developer has no idea that the data and the catalog information define logical tables and columns rather than "real" relational database tables and columns. Similarly, the user has no idea that the data accessed is stored in multiple, potentially non-relational data sources. Data appears to come from a single relational database.

WebSphere Information Integrator Classic Federation for z/OS Operational Environment



IBM WebSphere Information Integrator Classic Federation for z/OS in an IBM Information Management System (IMS) Environment



Data Server

At the heart of the operational environment is the WebSphere II Classic Federation data server that services SQL statements sent from tools and applications through the JDBC and ODBC clients. The data server runs in its own address space and typically listens on a TCP/IP socket for communication from the clients. All data access is done by platform and database specific "data connector" sub-components using native database I/O commands.

The data server always uses the documented API interfaces for the particular file system or DBMS to be accessed. It never relies on internal database control block organization. This reduces the likelihood of returning invalid results as well as minimizes the potential for errors during processing. It also ensures the integrity and security of the underlying databases and files.

Data server functionality includes:

- ◆ Determine the type or types of files and/or databases to be accessed.
- ◆ Rewrite the SQL query into one or more native file or database commands. One SQL command could translate into multiple native data requests. For multi-table JOINS, the SQL may translate into multiple native data accesses to different types of files and databases. For example, a JOIN between logical tables representing VSAM file data and IMS DB data would be serviced using the VSAM connector and the IMS connector.
- ◆ Optimize native data access calls based on the SQL statements and file or database characteristics. Query optimization is built in at several levels to ensure that the least amount of data is read and that all database aids are utilized.
- ◆ Filter the data to satisfy the SQL qualification. Typically this is to satisfy "WHERE" clauses.
- ◆ Sort result sets as needed (e.g., ORDER BY or GROUP BY)
- ◆ Construct a standard relational result set. For pre-relational

data stores like IMS DB, this involves restructuring the data, as it is accessed, into columns and rows and translating data types. It may also result in normalization of the data when segments contain embedded recurring data items (OCCURS and OCCURS DEPENDING ON constructs.)

- ◆ Process catalog queries using the WEBSPPHERE II Classic Federation metadata catalog
- ◆ Process INSERT, UPDATE and DELETE with support for commit/rollback/autocommit and two phase commit for IMS DB
- ◆ Invoke stored procedures for mainframe algorithm reuse. Stored procedures are defined to run within the data server's address space. In addition, users can utilize the APPC bridge to access programs running in other regions such as CICS.
- ◆ Integrate IMS DC transactions utilizing the WebSphere II Classic Federation transaction services and the data servers stored procedure mechanisms.

The Connector Approach

Embedded in the data server is a query processor that acts as a relational engine. The query processor has no knowledge of the physical databases/files being referenced in a SELECT, INSERT, UPDATE or DELETE statement. For each table referenced in an SQL statement, the query processor invokes a database or file type specific data connector.

The data connectors (as well as the query processor) are re-entrant so only a single copy is loaded even though multiple load requests may be issued based on the number of tables referenced in a statement and the number of concurrent users. This maximizes throughput while minimizing the operational footprint.

WebSphere II Classic Federation data connectors have been developed to use the most efficient, yet standard (i.e. comes with the database) multi-user environments available. WebSphere II Classic Federation never relies on internal database

control block organization. This ensures the integrity of the result set while leveraging the performance profile of the underlying database.

IMS Connector:

The IMS connector issues native DL/I commands to IMS DB. However, the IMS connector must be capable of servicing hundreds of simultaneous users with sub-second response time without interfering with existing transaction processing environments. This led to the use of the DRA interface to IMS DB.

DBCTL over DRA supports the dynamic scheduling of multiple PSBs that is needed for a multi-user environment. This is the same method used by CICS to interface with IMS DB databases. All IMS DB installations have this interface although they rarely code their own applications to use it.

The IMS connector acts as the coordination controller with IMS DB and is responsible for scheduling PSBs, issuing DL/I calls, un-scheduling PSBs and resolving in-doubt units-of-recovery. IMS DB connections and DL/I call completion are monitored using a DRA Control Exit running in the data server's address space.

Query optimization is built in at multiple levels and qualified Segment Search Arguments (SSAs) are used whenever possible. Detection of search fields is automatically performed during loading of the metadata catalogs by accessing the load module form of the DBD being referenced in the table mapping.

SSA generation is highly optimized based on the contents of the WHERE clause. Ideally, direct comparison SSAs will be generated when the WHERE clause contains equality or in-equality predicates. SSAs containing range comparisons are generated for SQL predicates such as BETWEEN and LIKE as well as SQL range comparisons.

Join optimization is performed in two ways:

- mapping paths in the IMS DB hierarchy
- using the appropriate JOIN predicates in the queries.

IBM WebSphere Information Integrator Classic Federation for z/OS in an IBM Information Management System (IMS) Environment



When accessing a path in the hierarchy, IMS DB "path calls" will be issued if the WHERE clause references a search field located in the bottom-most "leaf" segment in the hierarchy.

Fully qualified SSAs will be generated when multiple paths in a hierarchy (multiple tables) are referenced in the same query and join qualification is supplied on the columns that map to search fields in the segments that are common in the two hierarchic paths.

Insert-Update-Delete

All updates are done using standard SQL syntax. WebSphere II Classic Federation supports commit, rollback and autocommit enabling tools and applications to issue many updates into a database and commit or rollback all the changes.

In addition, WebSphere II Classic Federation supports 2-phase commit processing for updating that spans any combination of DB2 UDB for z/OS, IMS DB and CA-Datcom. However, full "XA" support is available only through the JDBC client.

To implement 2-Phase commit for IMS DB requires the data server's query processor task to go through an ODBA initialization service rather than the standard DRA initialization service. The query processor goes through MVS Resource Recovery Services (RRS) context switching services and 2-phase commit protocols as it processes the query. This mode is only used when 2-phase commit processing is required to minimize the impact of the additional overhead associated with the ODBA and RRS processes.

Using Stored Procedures

WebSphere II Classic Federation supports stored procedures that run in the data server's address space as well as IMS DC transactions. Stored procedures are user-written application programs and do not have to be database or file specific.

There are two "flavors" of stored procedures typically used in IMS DB/DC environments:

- non-transactional programs
- IMS DC transaction programs

Non-transactional Procedures:

This type of stored procedure is invoked using the standard JDBC or ODBC "CALL" syntax. The information passed to the data server by the ODBC or JDBC client is passed to the stored procedure application. Information returned to the data server from the stored procedure application will be returned to the client and then on to the requesting application or tool.

Data returned by the stored procedure can consist of a simple set of values (limited data) or a multi-row result set for larger amounts of data. The stored procedure application is responsible for making connections to databases or files that it accesses.

Stored procedure applications can connect to applications that run outside of the data server's address space using the supplied APPC "Bridge". This utilizes SNA LU6.2 scheduling features. In this type of implementation, the user must also build the "external" application that is being invoked.

Transactional Procedures:

WebSphere II Classic Federation's IMS transaction services is designed to invoke the majority of existing IMS DC 3270-based transactions. There are three basic types of 3270-based transactions supported:

- Independent Fast Path
- Non-conversational MPPs (message processing programs)
- Conversational MPPs (message processing programs)

IMS DC transaction programs use message segments or messages. From a design standpoint, the IMS DC transaction application program is expecting one or more input messages to provide the required input data. The IMS DC transaction application program then generates one or more output messages to report the outcome of processing the input data.

Input messages originate from a logical terminal. The output messages are by default sent back to that same logical terminal, though the IMS DC transaction program can also send output messages to both the original logical terminal and other logical terminal(s).

Non-conversational MPPs can accept multiple input messages and generate multiple output messages. Because dealing with multiple input messages is error-prone, a non-conversational MPP usually only uses a single input message. Also, for most transactions, a single output message is returned, although for something like a list screen, multiple output messages may be generated: one for the fixed portion of the screen and one or more additional messages for the repeating information.

Independent fast path transactions only send a single input message and receive a single output message when using MFS (Message Formatting Services). For fast path transactions that do not use MFS, multiple input and output message segments are supported.

Metadata Management

Logical tables mapped to IMS DB databases can define data from a single segment type or from multiple segments down a path in a hierarchy. To define a mapping, the WebSphere II Classic Federation data mapper loads an IMS DB data base definition (DBD) for reference. The DBD provides the database construct information including segment names, index definitions and hierarchy. COBOL copybooks are then used to define the contents of the various segments.

A Data Mapper user (typically a DBA) customizes the table mappings as needed for application requirements. Data from one IMS DB segment definition may be mapped to multiple, different logical tables where each logical table is designed to address the needs of a particular application. Conversely, data from multiple segments in a path can be mapped to a single table. A logical table can contain all or only some of the data from a segment definition.

SQL statements can be used to create views that reference the logical tables. Filtering logic (WHERE clauses) can also be attached to logical tables to simplify the use of the IMS DB data. This enables support for redefined record definitions as well as complex multi-segment joins. This simplifies the SQL that accesses the

IBM WebSphere Information Integrator Classic Federation for z/OS in an IBM Information Management System (IMS) Environment

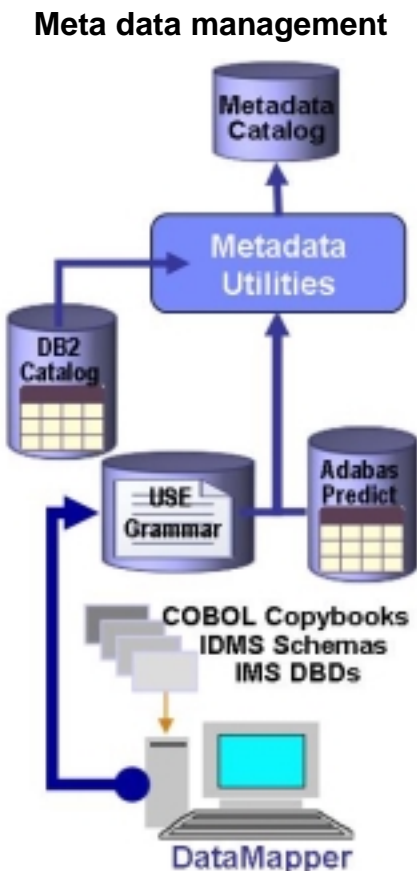


data as much of the data manipulation and filtering can be built in to the meta data.

The output from the Data Mapper is used as input to the z/OS-based metadata utilities that generate the metadata catalog used by the data server and the IMS connector. This input takes the form of a "USE" grammar file that acts like DDL for WebSphere II Classic Federation.

The Metadata utilities validate the IMS DBD information against the load version of the DBD. This ensures that invalid hierarchy and index data is not built into the metadata catalog.

Note that the metadata utilities use additional inputs for DB2 UDB for z/OS and/or Software AG Adabas data access. Similarly, the DataMapper uses IDMS schemas and sub-schemas to create mappings for CA-IDMS data access.



Wrap-up

WebSphere II Classic Federation for z/OS is superior because it:

- Supports complex SQL processing including cross-database joins
- Accesses IMS DB using a standard, efficient, scalable, multi-user interface - DRA
- Can JOIN data from multiple databases
- Provides a dynamic SQL interface
- Provides for complex metadata-driven filtering and cross-database logical table definitions via views
- Is managed and monitored like every other application with an installation's existing system tools
- Does not require mainframe programming for mainframe data access
- Is a read and write solution

- Can integrate both mainframe algorithms as well as IMS DC transactions using its embedded stored procedure capabilities
- Can be implemented stand-alone or in conjunction with DB2 Information Integrator for an enterprise-wide information integration infrastructure
- Does not require DB2 UDB for z/OS or any other underlying database
- Does integrate with existing security and accounting environments.
- Does support TCP/IP as well as MQ Series for cross-platform connectivity.
- Does include JDBC and ODBC application and tool connectivity as well as a call-level-interface (CLI) for distributed 3GL application connectivity.

In summary, DB2 Information Integrator Classic Federation for z/OS is a high performance, highly scalable distributed access and federation solution for z/OS mainframe installations committed to IBM's Information Management System