A Short Overview of DB2 Connect

By Craig S. Mullins

Applications that run on a non-zSeries platform require a gateway to access DB2 for z/OS data. IBM’s DB2 Connect product can be used to enable applications that run on a Linux, UNIX, or Windows environment to access mainframe DB2 data (or iSeries data).

DB2 Connect is an add-on product from IBM that implements DRDA Application Requester functionality for DB2. Using DB2 Connect, applications executing on non–
mainframe platforms can transparently access DB2 for z/OS databases. For example, an application running on a Windows workstation can use DB2 Connect to access DB2 data from the mainframe.

Of course, DB2 Connect is not the only gateway product on the market. Other popular gateway products exist for connecting DB2 for z/OS to 3-tier distributed applications, such as DataDirect’s Shadow Direct gateway. But today we're going to offer up an overview of IBM's DB2 Connect because it is the most popular of the available options for DB2 for z/OS.

IBM DB2 Connect

DB2 Connect provides fast, transparent read/write access to all DB2 family servers, not just z/OS. You can use DB2 Connect in conjunction with IBM DB2 Relational Connect to obtain transparent, consolidated access to Oracle, Sybase, Microsoft SQL Server, and Informix databases, too.

DB2 Connect forwards SQL statements submitted by application programs to host database servers. DB2 Connect can forward almost any valid SQL statement and fully supports the common IBM SQL, as well as the DB2 for z/OS, DB2 for VSE and VM (formerly SQL/DS), and DB2 for iSeries implementations of SQL.

As already mentioned, DB2 Connect implements the DRDA architecture. DB2 Connect provides support for distributed requests across databases and DBMS platforms. For example, you can join tables across different DB2 for z/OS subsystems or even code a UNION between a DB2 table and an Oracle table. Distributed request provides location transparency for database objects.

DB2 Connect supports multi-site update. Also known as distributed unit of work (DUW) and two-phase commit, multi-site update enables applications to update data in multiple remote database servers with guaranteed integrity. You can update any mix of supported database servers. For example, your application can update several tables in DB2 on Windows, a DB2 for z/OS table, and a DB2 for iSeries database, all within a single transaction.

DB2 Connect can be used to bring web applications and host data together. DB2 Connect provides broad support for developing and deploying applications using the latest web technologies, helping you in many cases to reduce development time and the cost of ownership for applications and enhancements. Whether you are developing web applications using Java, .NET, or ASP, DB2 Connect can be used to supply enterprise data to your application server.

Other beneficial features of DB2 Connect include:

- Support for both static and dynamic SQL
- Static SQL profiling for converting ADO, ODBC, DB2 CLI, and JDBC applications to static SQL
- Special optimized catalog for ADO, ODBC, DB2 CLI, and JDBC applications to make access even faster
- Resource management functions, such as connection pooling and connection concentrator, for building large-volume, enterprise web applications
- Federated database support for transparent access to DB2 and other data sources
- The ability to prioritize SQL requests submitted by DB2 Connect applications using the dynamic load balancing and fail-over capabilities of Workload Manager
- Support for authentication schemes and single sign-on capability using Kerberos and Microsoft Active Directory
- Encryption of user IDs and passwords to protect them from becoming compromised
- Integration with IPSec and SSL security protocols

Packaging and Editions of DB2 Connect

IBM offers several connection solutions under the DB2 Connect brand. If you decide to use DB2 Connect to hook up your applications to DB2 for z/OS, you will need to choose how you will connect them. The actual package or edition of DB2 Connect to use depends on your environment and needs.

I do not intend to replicate information about the various editions of DB2 Connect and what is implied by each. For that I direct you to the fantastic overview of the topic provided by Dean Compher titled DB2 Connect® Usage and Editions.

One thing worth calling out is the Type-4 JDBC driver, which enables users to connect to DB2 on zSeries and iSeries servers directly. Even though you do not need a DB2 Connect gateway in this case, you will still need to have a DB2 Connect license.

DB2 Connect Thread Pooling

The most resource-intensive prospect of using DB2 Connect to access mainframe data from a workstation is establishing connections. In an active distributed environment hundreds, or even thousands of client applications are connecting to and disconnecting from the host through the DB2 Connect server. Doing so consumes overhead as processing time is spent establishing and dropping connections.

To reduce this overhead, DB2 Connect Enterprise Edition (EE) uses thread pooling techniques that minimize the resources required on the DB2 for z/OS database servers. Efficiency is gained by concentrating the workload from all applications into a much smaller set of connections. This is known as thread pooling, and there are two techniques at the disposal of DB2 Connect: connection pooling and connection concentrator.

The default technique used by DB2 Connect EE is connection pooling. It allows reuse of an established connection infrastructure for subsequent connections. When a DB2 Connect instance is started, a pool of coordinating agents is created. When a connection request comes in, an agent is assigned to this request. The agent will connect to the DB2 server and a thread will be created. When the application issues a disconnect request, the agent will not pass this request along to the DB2 server. Instead, the agent is put back in to the pool.
The agent in the pool still owns its connection to the DB2 server and the corresponding DB2 thread. When another application issues a connect request, this agent is assigned to this new application.

Connection concentrator extends the features of connection pooling by improving load balancing in Parallel Sysplex configurations. With connection concentrator, DB2 uses a more sophisticated approach to reducing resource consumption for very high volume OLTP applications. This function can dramatically increase the scalability of your DB2 for z/OS and DB2 Connect solution, while also providing for transaction-level load balancing in DB2 for z/OS data sharing environments.

With connection pooling one application has to disconnect before another one can reuse a pooled connection. Rather than having the connection become free for use by another client at client disconnect, connection concentrator allows re-use of a server task when an application performs a COMMIT or ROLLBACK (as long as certain conditions are met).

Be sure that there are enough agents available in the connection pool to contain all active clients. Doing so enables connection concentrator to avoid unnecessary connection reuse outside of the load balancing process.Whereas connection pooling helps reduce the overhead of database connections and handle connection volume, connection concentrator helps increase scalability. With connection pooling, a connection is only available for reuse after the application owning the connection issues a disconnect. Connection pooling can improve overall performance for Web and client/server implementations having a large number of connections and disconnections.

Connection concentrator allocates host database resources only for the duration of each SQL transaction while keeping user applications active. This allows for configurations where the number of DB2 threads and the resources they consume can be much smaller than if every application connection had its own thread.

To summarize: keep in mind that the main difference between connection pooling and connection concentration is when the connection breaks. Connection pooling breaks the connection at disconnect, whereas connection concentration breaks the connection at the COMMIT point (after maximum logical agents is met).

Connection concentration has been available through the JDBC Type 4 driver for quite some time for connections to DB2 for z/OS. You do not need to have a running DB2 Connect instance to get Connection Concentration. Instead, you can just call the JDBC Type 4 driver or the DB2 Connect Driver as an API and still concentrate your connections.

Summary

This short post offers up a short little overview of IBM's DB2 Connect. It is not intended to be an exhaustive treatment of the subject but should offer sufficient information to introduce the DB2 professional to the capabilities of IBM’s distributed gateway solution.