

Managing Multiple Databases

Cheryl Walton

Accounting data, personnel data, customer data—whether companies are old or new, big or small, they must collect and store data. Companies have used computers to this end almost since computers were invented. However, as computers and network computing have changed over the years—from mainframes and dumb terminals to distributed processing and thin clients—so have the methods by which companies use computers and networks to store and manipulate data.

As a network administrator, you may work for a company that is big enough and old enough to have experienced these changes. If so, chances are that you are managing several separate databases running on several separate systems within your company's network. For example, one of the major airline companies has a number of databases for storing employee information. According to Ammen Harper, a product manager at Novell, when a new employee is hired, the airline com-

pany enters user account information into a minimum of three databases. After the employee has worked for the airline a while, he or she may appear in a dozen or more databases and directories throughout the company.

Even if the company you work for is relatively new and small, your company's network may be in the process of outgrowing the database that originally met its needs. For example, if your company is using Microsoft Access 97, a database included in Microsoft Office 97, your company may soon require a database that can accommodate more data fields per table. (Access 97 currently supports 255 fields per table. In contrast, Oracle8 supports 1,000 fields per table.)

Whether you work for a large company with several legacy databases or a small company in the process of outgrowing its current database, you can solve database problems with the following Novell products:



- NetWare 3.12 and above and the Novell Java Virtual Machine (JVM) for NetWare are reliable platforms for running many of today's commonly used database applications. (The Novell JVM runs on NetWare 5 and NetWare 4.)
- NetWare for SAA 4 and Novell's SQL Integrator 1.0 make it easier to manage databases running on a variety of platforms—even mainframes.

THE DIRECT APPROACH

Many popular databases run on NetWare. Choosing a database that runs on the version of NetWare your company uses has two advantages: Users can easily access the information stored in those databases, and you can easily manage users' access. For example, if your company is running Oracle8 on NetWare 4.11 or above, Novell Directory Services (NDS) provides authorized users with a single sign-on to all of your company's network resources, including Oracle8. In addition to enabling this single sign-on, NDS ensures that user sessions are secure by authenticating users through Rivest, Shamir, and Adleman Algorithm (RSA) public-key/private-key encryption technology. (For a definition of RSA encryption, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com>.)

The following database applications run on the NetWare platform:

- Oracle8 and 8i
- Pervasive.SQL 2000

Inside This Article

<i>The Direct Approach—Your Company's Database Options</i>	p. 16
<i>Data by the Cup—Java-Based Databases</i>	p. 18
<i>Cheers for Tiers—The Benefits of N-tier Architecture</i>	p. 20
<i>It's All in the Wrist—Tools for Accessing Databases</i>	p. 20
<i>A Well-Connected Host—NetWare for SAA 4</i>	p. 20
<i>Java for Data—Databases That Run on Novell's Java Virtual Machine for NetWare</i>	p. 21
<i>Multiplicity—SQL Integrator 1.0</i>	p. 22
<i>Four Faces, One Dictionary Manager—Specifying Your User Interface</i>	p. 24
<i>The Oracle Sees All—The Oracle8 and 8i Databases</i>	p. 28
<i>P's and Q's—Communicating Through SQL Integrator 1.0</i>	p. 30
<i>Buying In—Manipulating Database Information</i>	p. 30

Illustration: Jessica Chen

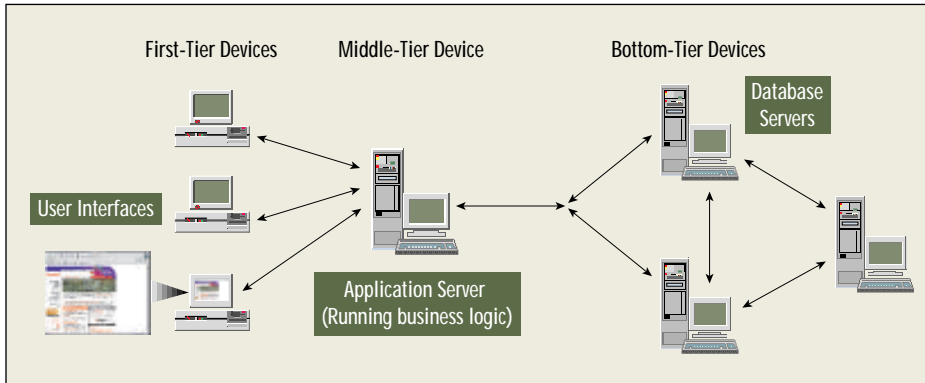


Figure 1. This n-tier architecture uses three tiers: browsers as first-tier devices, application servers as middle-tier devices, and database servers as bottom-tier devices.

- SQLBase 7.5
- Advantage Database Server 5.5
- InterBase 4.2.2
- SQL Anywhere 6.0.2

In addition, any database written in Java will run on the Novell JVM for NetWare.

Oracle8 and 8i

Oracle8 and 8i from Oracle Corp. are object-oriented relational databases. In addition to providing complete integration with NDS, these databases allow you to create tables that contain up to 1,000 fields and several petabytes of information. (For a definition of object-oriented relational databases and petabyte, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com>.)

Oracle8 runs on NetWare 4 and above, and a five-user version of Oracle8 ships with NetWare 5, NetWare 4.2, and Novell Small Business Suite 5. Oracle8i, which is Internet enabled, runs on NetWare 5. (For more information about how Oracle8 or 8i and NetWare can help you solve your company's database problems, see "The Oracle Sees All" on p. 28. For more information about Oracle databases, visit the Oracle web site at <http://www.oracle.com>.)

Pervasive.SQL 2000

Pervasive.SQL 2000 version 7.5 from Pervasive Software Inc. is an embedded relational database for small- to large-sized companies. Pervasive.SQL 2000 7.5 supports NDS and runs on NetWare 5, NetWare 4.11, and NetWare 3.2. This database also supports Binary Large Objects (BLOBs) and an unlimited number of fields and rows per table. (For an explanation of embedded relational data-

bases and BLOBs, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com>.) However, the total table size is limited to 64 GB of data. (For more information about Pervasive.SQL 2000 7.5, visit <http://www.pervasive.com/products/pervasive2000/psql.html>.)

Like databases written in any other language, Java databases can store and manipulate relational data and object-oriented data. Java databases can also be embedded in other applications.

SQLBase 7.5

SQLBase 7.5 from Centura Software Corp. is an embedded relational database for small- to medium-sized companies. This database runs on NetWare 3 and above and includes a variety of features, such as support for NDS, database page encryption, and database page alteration protection. SQLBase 7.5 also supports 245 fields per table and up to 512 GB of data. (For more information about SQLBase 7.5, visit <http://www.centurasoft.com/products/sqlbase>.)

Advantage Database Server 5.5

Advantage Database Server 5.5 from Extended Systems Inc. is an embedded

relational database for small- to medium-sized companies. This database runs on NetWare 3.11 and above and supports a number of features, including StreamlineSQL and BLOBs. (StreamlineSQL is a technology for speeding up SQL requests. For an explanation of SQL, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com>.) In addition, Advantage Database Server 5.5 supports tables that contain up to 65,534 records and 4 GB of data. (For more information about Advantage Database Server 5.5, visit <http://www.advantagedatabase.com>.)

InterBase 4.2.2

InterBase 4.2.2 from Inprise Corp. is an embedded relational database for medium-sized companies. InterBase 4.2.2 runs on NetWare 4.11 and can support up to 32 TB of information per table. This database supports concurrent access to data by multiple users. (For more information about InterBase 4.2.2, visit <http://www.interbase.com>.)

SQL Anywhere 6.0.2

SQL Anywhere 6.0.2 from Sybase Inc. is an embedded relational database for companies of all sizes. Through a technology called *SQL Remote*, SQL Anywhere 6.0.2 enables remote workgroups and mobile users to receive information from and send updates to a corporate database.

SQL Anywhere 6.0.2 runs on NetWare 3.12 and above. This database supports up to 1,024 GB of data and up to 999 fields per table. (For more information about SQL Anywhere 6.0.2, visit <http://www.sybase.com/products/anywhere>.)

DATA BY THE CUP

You can also run any database written in Java on the Novell JVM for NetWare. (For more information about databases that are written in 100 percent Java, see "Java for Data" on p. 21.) Although Java is an object-oriented programming language, Java databases are not necessarily object-oriented databases. (For an explanation of object-oriented databases, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com>.) Like databases written in any other language, Java databases can store and manipulate relational data and object-oriented data. Java databases can also be embedded in other applications.

It's All in the Wrist

You can use the following developer tools to create your own applications for accessing and manipulating data.

Product	Description	Company Information
Visual Basic 6.0 CE Toolkit	Development software for creating Visual Basic first-tier applications	Microsoft Corp. Voice: 1-425-882-8080 or 1-800-426-9400 Fax: 1-425-936-7329 http://www.microsoft.com
Visual C++ 6.0 CE Toolkit	Development software for creating Visual C++ first-tier applications	
Front Page 2000	Development software for creating web-based first-tier applications with Visual Basic script or Novell Script for NetWare	
WebDB	Development software for creating web-based first-tier applications	Oracle Corp. Voice: 1-800-672-2531 or 1-650-506-7000 http://www.oracle.com
Visual Café Database Edition 3.0	Java development software for creating first-tier applications	Symantec Corp. Voice: 1-800-441-7234 or 1-541-334-6054 Fax: 1-800-554-4403 http://www.symantec.com ●

For example, you might embed a Java database, such as Cloudscape 2.0, in the middle-tier of a Java e-commerce application. You might use Cloudscape 2.0, an embedded relational database, to store product and pricing information that the middle-tier application housing the database could then access. Furthermore, you could supply browser access to this middle-tier application via a servlet, making the entire e-commerce system platform independent. (For more information about Cloudscape 2.0, see "Java for Data" on p. 21.)

Java databases will meet the requirements of most companies. However, large-sized companies will probably need to use a database that handles more records and more data than Java databases can handle.

CHEERS FOR TIERS

Because Java applications are platform independent, they are particularly well-suited for the Internet. In fact, many Java databases are specifically designed to function as a tier in n-tier network architectures.

Commonly used to provide Internet or intranet services, n-tier networks implement three or more separate processes, each of which executes on a separate platform. For example, these processes might include a Java servlet (process one) that exchanges information with a

Java application (process two) that in turn exchanges information with a database (process three).

In n-tier networks, bottom-tier devices are back-end servers that contain corporate database applications and data stores. First-tier devices (clients) are typically workstations but can also be laptops or other mobile devices, such as personal digital assistants (PDAs). Middle-tier devices are application servers. The applications running on these servers supply the logic through which first-tier devices can access information and services running on bottom-tier servers.

First-tier devices often access middle-tier applications through browsers. For example, in a network that provides e-commerce services, a user might use a browser (a first-tier device) to access a servlet running on a corporate web application server (a middle-tier device). In turn, this servlet might access information from a database server (a bottom-tier device) and then return this information to the user's browser. (See Figure 1 on p. 18.)

Although commonly used as the basis for networks that supply Internet services, n-tier network architectures are not limited to providing user access to database information via the Internet. IT professionals such as Jim Abbott, a technology strategist for Novell, suggest that n-tier

networks are equally valuable within a company's intranet for providing users with access to the information and services they need to do their jobs. Abbott offers the following example to highlight the benefits of an n-tier architecture:

Suppose you needed to provide 12,000 users with access to database information. In an n-tier network, the application that supplies these users with database access runs on a middle-tier server, and users access this application through their browsers. In a traditional client-server network, on the other hand, the application that supplies users with database access runs on all 12,000 of the users' workstations.

Now suppose you had to upgrade the application that is supplying access to the database. In an n-tier network, you could simply upgrade the application running on the middle-tier server. In a traditional client-server network, however, you would have to upgrade all of the 12,000 applications running on the users' workstations. "Nobody in their right mind wants to do that," Abbott concludes.

A WELL-CONNECTED HOST

If you plan to implement an n-tier network, two Novell products can help you manage that network: In an n-tier network, NetWare for SAA 4 and SQL Integrator 1.0 run on middle-tier servers

and help you manage your company's databases, whatever those databases may be. That is, whether your company has one database running on a NetWare server or several databases running on different operating systems, you can use servers running NetWare 4.11 or above in the middle tier to manage the databases.

In the September 1998 issue of *NetWare Connection*, James E. Gaskin reported that "more than three-quarters of all corporate data still reside on mainframes." ("Making the Mainframe Connection With IntraNetWare for SAA 3," p. 24. You can download this article at <http://www.nwconnection.com/past/>.) If your company still stores its data on a mainframe, you can access and manage that data via NetWare for SAA 4.

NetWare for SAA 4 is a Systems Network Architecture (SNA) gateway that enables servers running NetWare 4.11 and above to access mainframe and AS/400 database management applications. SNA gateways also provide access to systems running software based on IBM's Systems Application Architecture (SAA). (For a definition of SNA and SAA, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com/>.)

IBM and Novell developed NetWare for SAA 4 to support native SNA features and native NetWare features, such as NDS and the Novell JVM for NetWare. NetWare for SAA 4 also supports applications that run on the following first-tier operating systems:

- Windows NT, 98, 95, and 3.11
- OS/2
- DOS
- Macintosh

With NetWare for SAA 4, you can use applications running on any of these operating systems to manage an n-tier network. For example, you can use the NetWare Administrator (NWADMIN) utility running on Windows NT.

You can also use development tools that run on these operating systems to develop applications that enable users to access bottom-tier databases running on IBM mainframes or AS/400 minicomputers. For example, you can use a Java development tool, such as Oracle Corp.'s JDeveloper, to create a Java application that enables users to access data on IBM mainframes or AS/400 minicomputers

Java for Data

The following databases run on the Novell Java Virtual Machine (JVM) for NetWare:

Product Name	Description	Company Information
PointBase Server Edition 2.1	Embedded object-oriented relational database	PointBase Inc. Voice: 1-650-570-6560 or 1-877-238-8798 (toll free) Fax: 1-650-570-6526 http://www.pointbase.com
Cloudscape 2.0	Embedded object-oriented relational database	Cloudscape Inc. Voice: 1-510-239-1900 or 1-888-595-2821 Fax: 1-510-239-1909 http://www.cloudscape.com
Jeevan 2.0	Embedded or distributed object-oriented database	W3apps Inc. Voice: 1-954-385-5739 Fax: 1-954-385-6856 http://www.w3apps.com ●

via a browser. This application would run on the middle tier—such as the Novell JVM for NetWare—of an n-tier

network. (For more information about NetWare for SAA 4, visit <http://www.novell.com/products/host/saa4.html>.)

J.A. Korsmeyer
1/3 Page
AD

4 7/8" x 4 7/8"
(4.875" x 4.875")

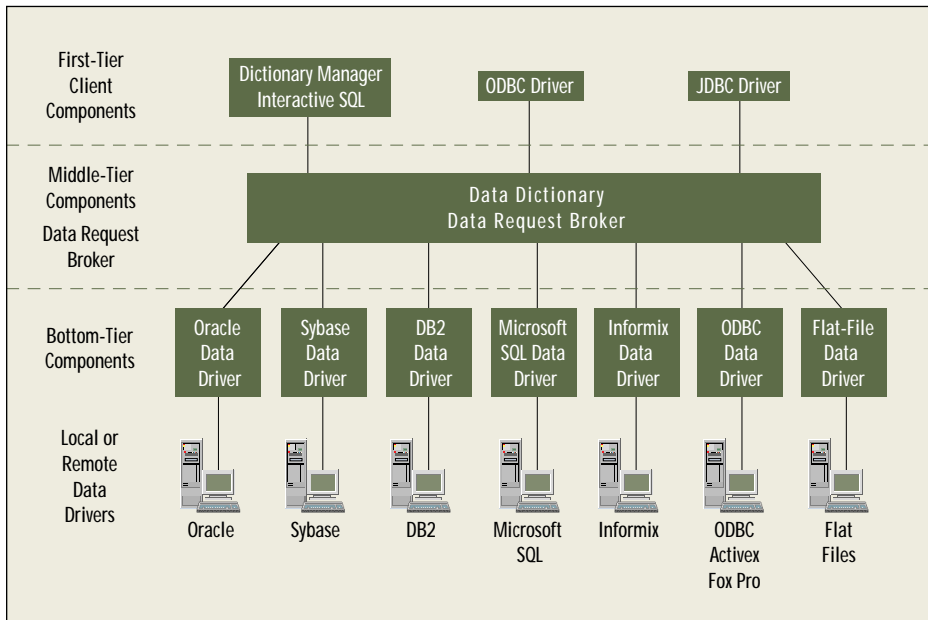


Figure 2. Novell's SQL Integrator has a component architecture that is well suited for n-tier networks.

MULTIPLICITY

If more than three-quarters of corporate data reside on mainframes and mini-computers, then nearly all of the remaining data reside on network servers. How can you access and manage databases running on these servers? According to Harper, one way to access and manage these databases is by using SQL Integrator 1.0. (Novell developed SQL Integrator in collaboration with B2Systems Inc. For more information about B2Systems, visit <http://www.b2systems.com>.)

SQL Integrator enables servers running NetWare 4.11 and above to connect to databases running on bottom-tier servers. SQL Integrator uses SQL commands to access and integrate data residing in many of the databases that run on the following server operating systems:

- NetWare
- Windows NT
- Solaris
- HP/UX
- AIX

"SQL Integrator converges all of the major-player databases into a single view," says Harper, adding that SQL Integrator also uses NDS to authenticate connections to those databases. In other words, SQL Integrator can access data from multiple tables that reside in multiple databases that are running on multiple platforms. SQL Integrator can then

present that data to the user as a single set of tables.

You can also use NDS to store the usernames and encrypted passwords that are required to access databases that are running on bottom-tier servers. Storing database-specific usernames and passwords in NDS eliminates the need for users to log in to each database they want to access via SQL Integrator. When users enter their NDS username and password to log in to a first-tier application, SQL Integrator passes database-specific login information from NDS to each connected database.

SQL Integrator has a component architecture that lends itself especially well to n-tier networks. (See Figure 2. A *component* is an independent program that has a specific function, such as accessing a database.) SQL Integrator's components fall into one of the following categories:

- Client
- Data Broker
- Data Drivers

CLIENT

The client category includes the following components:

- Open Database Connectivity (ODBC) driver
- Java Database Connectivity (JDBC) driver
- Dictionary Manager

The ODBC driver allows first-tier applications and development tools that support the Microsoft ODBC application programming interface (API) to communicate with SQL Integrator's Data Broker components. These Data Broker components run on the middle tier of n-tier networks and act as intermediaries between Client (first-tier) and Data Driver (bottom-tier) components. (Data Broker components are discussed in more depth in the next section.)

The ODBC driver supports first-tier applications that use Microsoft's ODBC API 2.5 and above. The ODBC driver enables these first-tier applications to request information from middle-tier SQL Integrator Data Broker components. These Data Broker components then access bottom-tier databases to supply the requested information. (For more information about the ODBC API, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com>.)

The JDBC driver uses the JDBC API to allow first-tier Java applications to request information from middle-tier SQL Integrator Data Broker components. These middle-tier components then access bottom-tier databases for the requested information. (For more information about the JDBC API, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com>.)

The Dictionary Manager is a first-tier Java application that enables you to create and maintain data dictionaries through a variety of user interfaces. Data dictionaries are middle-tier files that store links between the logical database tables that first-tier applications use and the physical database tables that are stored in bottom-tier databases. (For more information about data dictionaries, see the "Look It Up" section on p. 24.)

To use the Dictionary Manager, you must install it on your workstation. You can also specify startup command options when you install Dictionary Manager. For example, you can designate the user interface you prefer. (For more information about user interfaces for the Dictionary Manager, see "Four Faces, One Dictionary Manager" on p. 24.)

Whichever interface you choose, you will be able to access Interactive SQL, a tool that enables you to execute SQL statements that query any of the databases to which SQL Integrator provides a connection. For example, if you were

connected to a database containing employee information, you might execute the following SQL statement:

```
SELECT empname FROM emp
```

Interactive SQL would then retrieve all of the employee names in the selected table.

The Dictionary Manager—and therefore Interactive SQL—supports American National Standards Institute (ANSI) SQL-92, as do the ODBC and JDBC drivers. (SQL-92 is the latest version of SQL.) Furthermore, Novell's implementation of SQL-92 is compatible with all database vendor implementations of SQL. As a result, you can use uniform syntax regardless of the database you are accessing. (Although SQL is a standard language for relational databases, most database vendors use their own dialects of SQL. For example, because ANSI does not define a standard SQL command for creating an index of the fields within a database table, each database vendor creates its own SQL command to perform this function.)

In addition, the Dictionary Manager allows you to create and maintain flat-file tables. (A *flat-file table* is an ordinary file that contains tabular data formatted with fixed-length fields.) You can create and maintain these tables on the same server that is running SQL Integrator middle-tier components. Among other things, you can use flat-file tables to develop applications when you don't know the structure of the actual database the application will be accessing.

Four Faces, One Dictionary Manager

SQL Integrator's Dictionary Manager allows you to specify the user interface you prefer at startup. You can designate the user interface you prefer by including the following option at the end of the Java command that starts Dictionary Manager:

```
-ui <user interface>
```

You can replace user interface with one of the options listed below:

- Windows (the Microsoft Windows look and feel)

DATA BROKER

When first-tier applications request information that resides in bottom-tier databases, these requests go to the SQL Integrator's Data Broker components. The following Data Broker components work together to supply that information:

- Data Dictionary
- Data Request Broker

Look It Up

To help you create and maintain data dictionaries, the Dictionary Manager accesses the Data Dictionary, which contains information about all of the data types supported by SQL Integrator's implementation of SQL-92. For example, SQL Integrator supports time stamp, character, and integer data types.

First-tier applications can ask for any data type supported by SQL Integrator, whether or not the actual bottom-tier database supports that data type. If a first-tier application requests information from a bottom-tier database that does not support the requested data type, the Data Dictionary emulates the requested data type and then maps the requested data type to a similar data type that the bottom-tier application does support.

The entries in data dictionaries point either to tables residing in bottom-tier databases, to flat file tables, or to database views. (A *database view* is a table that contains a subset of the fields and rows that exist in a parent table or a selection of fields and rows from several tables that have been joined together.) Entries in data dictionaries do not actually contain data stored in database tables, but instead

- Motif (the motif look and feel)
- System (the user interface of the operating system on which Dictionary Manager is running)
- Metal (the Java look and feel)

If you do not include the `-ui` option, the default user interface is metal.

If you install Dictionary Manager on a Windows operating system, the installation program automatically creates a startup shortcut. This shortcut will include the startup command options—including any `-ui` options—you specified when you installed Dictionary Manager. ●

point to that data. As a result, you can change data contained in bottom-tier databases without affecting SQL Integrator's ability to provide first-tier services.

SQL Integrator allows you to create multiple data dictionaries and to include separate tables in any data dictionary you create. For example, you might create one data dictionary that contains human resources tables and another dictionary that contains product development tables.

The Dictionary Manager stores information about the data dictionaries you create in a global data dictionary called a *dictionary catalog*. The dictionary catalog contains the following information:

- The names of individual data dictionaries.
- The name of the server on which the Data Request Broker is running. (The Data Request Broker runs on the same server on which the data dictionaries reside.)
- Connection strings and environment variables through which the Data Request Broker relays requests from first-tier applications to bottom-tier databases.

Data Request Broker

Like the Data Dictionary, the Data Request Broker is a middle-tier component. The Data Request Broker uses the entries contained in data dictionaries to relay requests from first-tier applications to the bottom-tier databases that hold the requested information.

When a first-tier application sends a request to the Data Request Broker, it analyzes that request to determine whether the request involves tables from a single database or tables from multiple databases. Requests that involve a single database are called *homogeneous requests*. Requests that involve multiple databases are called *heterogeneous requests*.

If the request is a homogeneous request, the Data Broker sends the entire request to the bottom-tier database to which that request is directed. If the request is a heterogeneous request, the Data Request Broker decomposes the request—that is, the Data Request broker breaks down the request into its constituent subrequests. The Data Request Broker then sends these subrequests to each of the databases from which information is requested.

For example, suppose an original request included the following: `update emp`

The Oracle Sees All

If you want to implement a database that runs on NetWare, particularly if you want that database to be part of an n-tier network architecture, you may want to consider implementing an Oracle8 or 8i database. (For an example of the n-tier network architecture, see Figure 1 on p. 18.) According to Larry Robinson, director of the Novell products division at Oracle, Oracle designed Oracle8 and 8i to be compatible with the n-tier network architectures that are designed to support Internet and intranet services. Oracle8 and 8i run on NetWare 5 and are fully integrated with some of the technologies that make NetWare 5 especially well suited for n-tier architectures.

Specifically, Oracle8 and 8i are fully integrated with the following NetWare 5 technologies:

- Novell Directory Services (NDS)
- Novell Java Virtual Machine (JVM) for NetWare
- Netscape Enterprise Server for NetWare

A DIRECTORY FOR ALL REASONS

Whether your company's Oracle8 or 8i database is running on an n-tier network or a traditional client-server network, full integration with NDS allows you to manage that database just as you manage other network resources—through Novell's NetWare Administrator (NWADMIN) utility.

NDS integration also allows authorized users to access both the network and the Oracle8 or 8i database with a single login. In a traditional client-server architecture, authorized users can log in to Oracle when they log in to the network via NDS. In an n-tier architecture, authorized users can use their browsers to log in to NDS and Oracle either over the Internet or your company's intranet.

SEE JAVA RUN

Integration with the Novell JVM for NetWare makes it easy for Java applications running on the JVM to access information stored in Oracle8 and 8i databases. For example, an application running on the Novell JVM for NetWare may use the Java Database Connectivity (JDBC) application programming interface (API) to access relational data stored in your company's Oracle database. (For more information about JDBC, see the glossary on the NetWare Connection web site at <http://www.nwconnection.com>.)

In addition, Oracle8 and 8i are object-oriented relational databases. As a result, an application running on the Novell JVM for NetWare may use an object-oriented interface to access Java processes (such as Java Beans) that are stored as objects in the same Oracle database that is storing relational data. (For more information about object-oriented relational databases, see the glossary on the NetWare Connection web site at <http://www.nwconnection.com>.)

In fact, you can not only store but also run these Java processes in the Oracle8 and 8i databases. Each Oracle8 or 8i database includes a specialized JVM, which has the ability to execute (within the database itself) Java code that is stored in the database. In other words, an Oracle8 or 8i database is a repository for both application logic and the data upon which that logic operates.

"If you think about it, Java being a component environment, or a NetWare component environment, you can have a Java application that's running on the Novell JVM actually executing Java inside Oracle, on the Oracle JVM, and the two processes communicating," Robinson explains.

The ability to both store and execute data-driven processes in the database can improve the overall efficiency of middle-tier applications. These applications do not need to call data out of the database and then manipulate that data on the application server. Instead, the middle-tier application can call a process that executes within the database, and that process will then return the results to the middle-tier application.

This feature also allows you to put business rules near the data to which the rules apply. For example, if you were running a middle-tier application that processes customers' orders, this application might access customer information in your company's database. Upon accessing this customer information, the middle-tier application could automatically invoke a process that runs inside your company's database and checks the customer's credit history.

"The nice thing about putting business rules in the database is that they're near the data. It doesn't matter how people access the data; the rules are always defined," Robinson concludes.

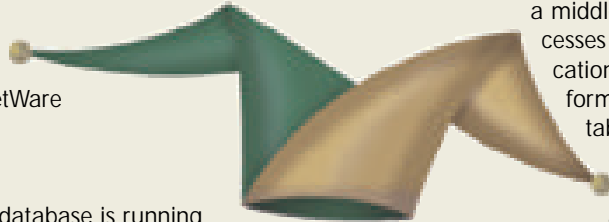
WEB SERVICE

If you have implemented an n-tier network architecture, users probably access database information via applications running on a middle-tier web server. Since Oracle8 and 8i are integrated with Netscape Enterprise Server for NetWare, you can easily develop web-based applications that run on Netscape Enterprise Server and access these databases.

Netscape Enterprise Server for NetWare includes a number of APIs, such as the Java servlet API and the Common Gateway Interface (CGI) API, and widely used scripting languages, such as Perl script and JavaScript. These APIs and scripting languages make it possible to develop a variety of applications that users can access via their browsers and that can, in turn, access Oracle8 and 8i databases. (You can download Netscape Enterprise Server for NetWare for free by visiting http://www.novell.com/download/#Netscape_Servers_for_NetWare.)

For example, you could create a human resources form that enables users to request—through their browsers—information about their benefits. This form might consist of an HTML document that contains embedded JavaScript processes which retrieve the requested information from the database via the JavaScript API.

You could also create a servlet that retrieves information via the Java servlet API, or you could create an HTML form with embedded Perl script. In other words, Oracle's integration with NetWare technologies such as the Netscape Enterprise Server and the Novell JVM for NetWare enables you to implement almost any middle-tier application imaginable. As Robinson points out, above all else, Oracle's integration with NetWare 5 technologies "is about choice." ●



set birthday = (select birthday from personnel where empno = 1234). Upon receiving this request, the Data Request Broker would break down the request into an update statement (update emp set birthday) and a select statement (select birthday from personnel where empno = 1234). The Data Request Broker would then process the select statement first, since the broker needs this information to perform the update request.

The Data Request Broker can also decide what requests or subrequests can be pushed down to the databases themselves. A request that is pushed down to a database is processed within that database rather than within the Data Request Broker. Without the ability to push requests down to databases, the Data Request Broker would have to retrieve the data with which to process those requests—obviously a bandwidth-intensive process.

In addition, the Data Request Broker can translate the SQL dialect of the request to the SQL dialect of the database storing the requested information.

DATA DRIVER

After the Data Request Broker translates SQL requests from first-tier applications to the SQL dialect of a particular database, the broker sends the translated requests to SQL Integrator's Data Driver components. These components run on the same servers that are running bottom-tier databases and interface directly with the connectivity libraries supplied by the vendors of these databases. Each database's connectivity library processes SQL requests in the SQL dialect recognized by that particular database.

SQL Integrator provides two types of Data Driver components:

- Local Data Driver components
- Remote Data Driver components

Some corporate databases are part of traditional client-server networks rather than bottom-tier processes in n-tier networks. To provide access to databases on client-server networks, SQL Integrator includes Local Data Driver components. Local Data Driver components allow the Data Request Broker to connect to databases running on the same NetWare middle-tier server upon which the Data Request Broker is running.

Local Data Driver components for NetWare 5 servers are available for flat-file

tables, Oracle 7.3.2, and Oracle 8.0.3.0.6 databases. Local Data Drivers for NetWare 4.2 servers are available for flat-file tables, Oracle 7.3.2, and Sybase 11.0 databases. Local Data Drivers for NetWare 4.11 servers are available for flat-file tables, Oracle 7.3.2, and Sybase 11.0 databases.

To provide access to bottom-tier databases in an n-tier network, SQL Integrator includes Remote Data Driver com-

ponents. These components enable the Data Request Broker to connect to databases running on bottom-tier servers. Remote drivers are available for the following databases:

- Oracle running on Windows NT, HP/UX, AIX, and Solaris
- Sybase running on Windows NT, HP/UX, AIX, and Solaris

LearnKey 1/2 Page Island AD

4 7/8" x 7 3/8"
(4.875" x 7.375")

Buying In

The table below contains a few third-party products that can help you query and manipulate database information.

Product	Description	Company
Enterprise Reports 3.0	Java report writing software that enables you to extract, analyze, and format data	EnterpriseSoft Voice: 1-510-742-6700 Fax: 1-510-742-6800 http://www.EnterpriseSoft.com
Jano	Java engineering and product data management (EDM/PDM) software for manufacturing companies	Methus s.r.l. Voice: + 39 39 2304425 Fax: + 39 39 2304426 http://www.methus.com
R&R Report Writer 8.0	Data access and reporting software that uses plain English commands	Information Interface Systems Inc. Voice: 1-800-726-5816 or 1-510-527-5822 Fax: 1-510-527-5816 http://www.infois.com
Crystal Reports 7	Report writing software that includes a web-based Java query tool	Seagate Software Voice: 1-800-877-2340 or 1-604-681-3435 Fax: 1-604-682-2934 http://www.seagatesoftware.com ●

- Microsoft SQL Server running on Windows NT
- IBM DB2-6000 running on AIX
- Informix running on Windows NT, HP/UX, AIX, and Solaris
- Microsoft ODBC running on Windows NT

P'S AND Q'S

To help you design SQL requests using Interactive SQL, the documentation that comes with SQL Integrator includes the SQL Grammar Manual. However, in many cases neither you nor your users will want to spend time minding the p's and q's of SQL grammar. Fortunately, you don't have to speak fluent SQL to communicate with databases through SQL Integrator. You can let first-tier applications communicate for you instead.

If you want to implement first-tier applications that communicate through SQL Integrator rather than directly with bottom-tier databases, you have the following options:

- You can purchase third-party application development tools that help you develop customized applications.
- You can use scripting languages and other development tools available through Novell to develop a customized application.

Store-Bought

If you want a first-tier application that accesses bottom-tier databases but you are not a programmer, you can probably find third-party software that meets your company's needs. (For more information about third-party applications designed to query databases, see "Buying In.") Most third-party applications designed to access information from a single database can access SQL Integrator instead.

Although SQL Integrator is a middle-tier application rather than a database, it "looks" like a database to the first-tier applications that connect to it. When you configure third-party applications to access SQL Integrator, you actually enable those applications to query multiple databases and, therefore, to access more data with a single query.

Typically, when you install such applications, you must supply connection information that tells the applications

what data source you want them to draw from. In most cases, directing the applications to SQL Integrator rather than to a bottom-tier database is fairly simple. For example, suppose you want to enable Seagate Software's Crystal Reports to access database information through SQL Integrator. (Crystal Reports allows you to query a database and then create reports based on the results of that query. For more information about Crystal Reports, visit Seagate's web site at <http://www.seagatesoftware.com>.)

To direct Crystal Reports to use SQL Integrator as its data source, you complete the following steps:

1. Install the SQL Integrator ODBC client driver on the first-tier Windows NT, 98, or 95 workstation.
2. Use the ODBC Administrator to create a Data Source Name (DSN) for using SQL Integrator on the middle-tier NetWare server.
3. Install Crystal Reports on the same workstation.
4. When the Crystal Reports wizard prompts you to name a data source, click ODBC. The ODBC driver lists all the DSNs of the databases it can connect to, including SQL Integrator.

5. Click the DSN you created for SQL Integrator.

Homemade

If you want to write a first-tier application but are not a full-time programmer, you can purchase third-party development software to help you create that application. For example, you can use BulletProof Corp.'s JDesigner Pro to create a server-based Java application that enables users to query your company's databases via their browser and SQL Integrator. JDesigner Pro is Java development software that uses a graphical interface through which you can access managers and wizards that help you design Java applications. (For more information about JDesigner Pro, visit <http://www.bulletproof.com>.)

The Right Tools

Even if you are a programmer, application development software can save you time and energy when creating applications that provide first-tier access to bottom-tier information. For example, application development software that supports scripting languages is useful for writing web-based applications that give users access to information via their browsers. (For an explanation of scripting languages, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com>. For more information about software that enables you to write web-based applications to connect with bottom-tier databases via SQL Integrator, see "It's All in the Wrist" on p. 20.)

If you would rather not buy third-party software, you can download the Novell Developer Kit (NDK), which contains several scripting engines. For example, Novell has scripting engines for Perl script, Visual Basic script, and JavaScript. (To download the NDK, visit <http://developer.novell.com/ndk>.)

You can also download application development software that supports Novell Script for NetWare (NSN), which is scripting software that supports Novell Script Pages (NSP) and NetWare Data Objects (NDO). NSP is application development software that enables you to embed NSN script directly into HTML web pages, which in turn allows you to create forms-based applications that users can access through their browsers. NDO is a Universal Component Extension (UCX) that extracts data from SQL-compliant

databases. In other words, NDO is a component that enables forms-based and other web-based applications to access bottom-tier databases and SQL Integrator. (For more information about UCX, see the glossary on the *NetWare Connection* web site at <http://www.nwconnection.com>.)

Unlike third-party application development tools, NSN, NSP, and NDO are available free to Novell customers. You can download NSN, NSP, and NDO at <http://developer.novell.com>. (For an example of how you can use NSN to develop first-tier applications, download "Oracle Expense Account," a demo application that uses NSN, at <http://developer.novell.com/ndk/demo.htm>.)

CONCLUSION

The way companies store, access, and manage data will continue to change as computing technologies change. For example, companies have begun to use Enterprise Information Portals (EIPs), such as MyEureka from Information Advantage Inc., to search corporate data stores. EIPs take advantage of n-tier net-

work architectures to provide users with a single point of entry into all of the resources on a network.

Mark Potter, database architect for IMS Health in Canada, likens EIPs to Yahoo's portal. "The EIP is meant to give you a common view, like a web interface, that insulates the user from the fact that there are scores of different systems under the hood," Potter explains. "You can search for information by starting with a topic and drilling down."

This trend indicates that Internet technologies and n-tier architectures will continue to inspire new approaches to old problems. With SQL Integrator, Novell not only puts NetWare in the middle of n-tier architectures but also supplies one of the newest approaches to the old problem of accessing multiple databases that are running on multiple platforms.

Cheryl Walton works for Niche Associates, an agency that specializes in writing and editing technical documents. Niche Associates is located in Sandy, Utah. ●

Biscom
1/3 Page
AD

4 7/8" x 4 7/8"
(4.875" x 4.875")