Chapter 1

A Pragmatic Introduction to Oracle

In This Chapter

- Getting familiar with Oracle
- Implementing grid computing
- Incorporating Oracle into everyday life

Oracle 11g is by far the most robust database software on the market today. It's also the leading database software used and sold all over the world. It has become an enterprise architecture standard for managing data, regardless of the data's size or complexity.

This chapter highlights the reasons to use Oracle 11g.

Introducing a New Kind of Database Management

Oracle is software that efficiently organizes data in a relational manner. Before Oracle, other database software ran on mainframes and used a *hierarchical data model* where data is stored in a tree-like structure as flat files — those crazy COBOL programmers!

The *relational model* is a concept where data is logically stored. These design elements are in the form of tables. Tables have columns, and the columns have attributes (character or number, for example). The tables are organized to store specific data. The tables relate to one another through primary keys.

For more clarity, Oracle, the company, was founded on the database software that transformed the industry into what it is today. Oracle, the company, owns many software products and applications that it has written or acquired, but the database software is still Oracle's core product.

TIP

This book focuses more on database administration rather than Oracle applications administration.

Decoding the *g* in Oracle 11g

Oracle has always had some creative marketing techniques. In the late 1990s, the Internet was booming, and everyone wanted Internet technology. Oracle released an upgraded version of Oracle 8 and labeled it 8i. *i* represents the Internet. This addition was a popular move because businesses realized the advantages of providing access via the Internet. Use of the Internet also reduced the labor and cost requirements for client server applications in which the client was installed onto the end user's PC.

As popular as the Internet boom was, grid computing is now the evolution of enterprise architecture management. (Hence the *g,* which stands for grid.)

Pooling Resources with Grid Computing

Grid computing offers a pool of distributed resources for computing services. It's simply described as computing as a service, similar to a utility-type model.

Oracle supports grid computing with its Real Application Clusters (RAC) capability and its Oracle Enterprise Manager (OEM):

- **RAC** uses Oracle's clustering software to manage a highly available environment. If you need additional hardware resources (such as memory or CPU), or experience hardware failure, you simply add another *node* (server) to the grid. (Truthfully, it's more complicated than that, but you get the point.)
- **EM** manages the databases and hosts, which are also called *targets.* It has a web interface that gives you a comprehensive view of each target's state. It handles all the monitoring requirements and provides other web-based tools to interact or perform maintenance with.

Together, RAC and EM make up the components to support true grid computing. RAC is a complex architecture that requires a fair amount of systems and database administrator knowledge, which is unfortunately beyond the scope of this book.

Chapter 13 covers the capabilities and configuration for EM and its lighter single database version, DB Console. You can find additional information about Oracle RAC at www.oracle.com/database/rac_home.html.

Anticipating Technology and Development Trends

Oracle's success is partially due to anticipating, adapting, and establishing database technology trends. You can choose from numerous designing tools and Integrated Development Environment (IDE) technologies, such as Service Oriented Architecture (SOA), Java, and Extensible Markup Language (XML).

These technologies are portable, which reduces hardware or software dependencies and suits standard *business-to-business (B2B)* processing and communication:

- **SOA** is a style of IT architecture that utilizes a build-once/deploy-many concept. Its root definition includes webcentric services that work together to sustain business processes. SOA separates the application function from the underlying software and hardware to allow better use (or reuse) of application processing. These functions or service units are written to be flexible by design and capable of service-to-service communication.

  SOA concepts eliminate hard coding and stove piping of applications for better use with other applications. Generally, SOA is engineered for large enterprise architectures that require a scalable, cost-effective approach to application development and maintenance.
- **Java** is a free programming language that standardizes applications across hardware platforms. This write-once/run-anywhere programming language supports *object-oriented programming (oop)* methodologies. Java is widely used for enterprise-level applications on the web and is very popular because it can run on any operating system without much tweaking. Oracle supported Java shortly after its creation.
- **XML** is an all-purpose language that helps share data across systems via the Internet. It standardizes the programming methods or calls, which allow for B2B communication. XML supports the SOA framework as well.

Meeting Oracle in the Real World

The Oracle 11g database can support any requirement you have for using and storing data. From financial institutions, such as banks, to human resources or manufacturing applications, Oracle can handle it. Its strengths lie in its vast number of software components and its ability to recover to any point in time.

General Oracle use supports a variety of applications that are labeled by type. The following list outlines the majority of database types:

- **Online Transactional Processing (OLTP):** Used for transaction-oriented applications where the response is immediate and records are modified or inserted regularly.
- **Decision Support System (DSS):** Used for processing data and making judgments on data for making decisions. A DSS database usually involves many ad hoc queries.
- **Online Analytical Processing (OLAP):** Used for analyzing data. Typically, OLAP is used for business intelligence or *data mining,* such as budgeting or forecasting.
- **Hybrid:** Acts as a multifunctional database. Most hybrid databases contain transactional, processing ad hoc querying, and batch processing. Larger databases that have service-level requirements are generally isolated to their own databases for performance and manageability reasons.

Uses for Oracle center around data and information. Industries leaders are particularly interested in information. Have you heard the motto "Information Drives Performance"? That motto basically suggests that the performance of a company is relative to the information it has and uses. This information assists in making more competitive and educated decisions.

A good example of this process is how Amazon and eBay use their information. They track user interaction on their Web sites to help define a user's shopping tendencies and interests. They then make programmatic recommendations based on that information to promote purchases, which in turn creates revenue. Information usage in this manner is known as *Business Intelligence* (BI) and is a common practice among many businesses today. Instead of saying, "Build it and they will come," Oracle can say, "Get their information and build them something they can't refuse."

Making the Oracle Decision

The decision to use Oracle over other technologies or database software can be a difficult one. Several things can influence your decision:

- Cost
- Available expertise
- Project scope
- Scale

Most of our clients decided to use Oracle based on available expertise because pricing is fairly competitive across database companies. In one case, Microsoft SQL Server was almost chosen because the developers had ASP/VB.NET experience. If the developers were Java eccentric, the database software would have never been discussed. Management, however, realized that it could use the pre-existing Oracle database infrastructure and still develop with Microsoft products.

Microsoft Access and even Microsoft Excel have their place, but if you want functionality, scalability, recoverability, and security, Oracle is the best choice. Linux gurus also use MySQL or PostgreSQL. Both are free for public use. The difficulty in using or managing MySQL or PostgreSQL is finding qualified expertise. You also need to consider the software support capability of the product. Oracle support provides a deep, mature group and a knowledge base for issues, such as bugs or general guidance.

In comparison to other database software products, Oracle has a similar level of complexity in installing, configuring, and maintaining it. Senior expert-level professionals are sometimes necessary for particular issues, but most novices to Oracle can achieve success without much training or guidance. We've trained many DBAs in our day, and they all had very little knowledge of Oracle but were eager to get their hands dirty. A good understanding of information technology and computers in general definitely helps with the learning curve.

REMEMBER

Oracle runs on all the common and latest operating system versions of Linux, UNIX, Microsoft Windows, Mainframes, and Mac. It provides the same functionality and utilities regardless of the operating system or hardware. It also supports 64-bit architecture to add additional memory space for large applications. You can purchase licensing per CPU or per named user.

TIP

Additionally, Oracle provides lower-cost licensing for its standard editions. Oracle licensing information is available at www.oracle.com/corporate/pricing/technology-price-list.pdf.