Building Database Applications for the Web

The fervor generated by the World Wide Web (WWW) is prompting many organizations to drastically rethink their application implementation strategies. Its potential to make information more accessible and dynamically customizable is very appealing. To support the rapidly evolving web requirements, a new breed of application development tool is required - tools that can build and deploy applications extremely quickly, are flexible enough meet a wide range of demands, and are highly cost and resource efficient. This paper describes the types of web applications that users are trying to build, reviews the benefits and challenges that building web applications will likely bring, and explains how Oracle's tools meets the challenges and addresses the potential that the web provides.

The Evolution of Web Applications

Several types of web-based applications have evolved with user and organizational demands:

1. *Static web pages*

These web applications consist of static web pages. They are the most prevalent applications on the internet today as they are easy to create and manage for even a low-level user. The applications integrate different kinds of media (such as text, images and video) and typically allow users to move from one page to another through a series of links. Since these applications are static, read-only and generally not tailorable to specific users, they focus on visual appeal to maintain user attention.

2. *Dynamic web publishing applications*

These web applications evolved from the desire to make information access within an organization more timely. To make the applications dynamic to reflect constantly changing information, the content of the application is usually stored in a database. The latest information is presented to the user without needing the intervention of a web master to bring it up to date.

3. *Transaction-oriented applications*

These web applications are geared to replace or coexist with internal client/server or legacy applications. They perform the same function as traditional applications, but contribute a consistent (and richer) interface across different platforms, and potentially lower resource requirements and maintenance costs to the equation.

4. *Electronic commerce*

The web enables a much closer electronic interaction between organizations and their customers and suppliers. Users can interact online with an organization without needing to install a specific application or setup up special hardware. Customers can execute transactions without the intervention of sales representatives and suppliers can monitor inventory levels without assistance. In addition, applications can be customized according to the preferences and needs of each individual user.
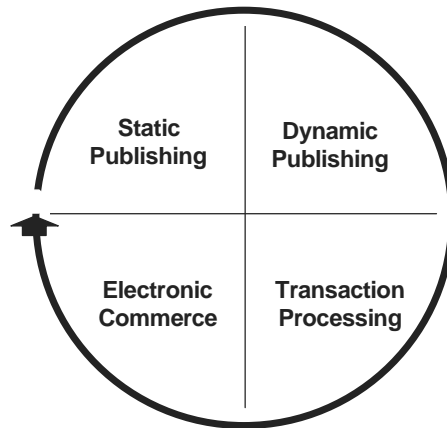
Fig 1: The Evolution of Web Applications

Benefits of the Web

Some of the benefits that users are looking to web applications to provide are as follows:

1. *A reduction in application administration costs*

Increasing administration costs are a side-effect of the move to client/server. Program executables and files must be installed on each client to run client/server applications, and that software must not only coexist with other applications, but must also be upgraded and maintained on every client. As more applications are deployed to more desktops, the cost of maintaining those applications can increase rapidly.

Web applications offer the potential to significantly reduce maintenance costs by reducing the requirement to install applications on desktops, and instead allowing them to be set up on centralized application servers and downloaded to the desktop when needed.

2. *A reduction in resource requirements*

Client/server applications typically require meaningful amounts of RAM over and above the requirements of the windowing environment under which they are running. In addition, they can be greedy consumers of often limited disk space.

Web applications generally have lower resource requirements. Since the "body" of web applications executes on an application server, less RAM is required on the desktop to execute the application. Also, the application is installed on an application server, reducing client disk space requirements.

3. *Application portability*

Client/server developers have struggled with the complexities of deploying applications that run on more than one windowing platform. The difficulties arise from visual as well as operational differences between these environments. In many cases, applications have to modified for each platform, making the task of maintaining them even more challenging.

Web applications can execute within the confines of a portable web browser using machine-independent languages such as HTML and Java. This enables developers to focus on building the best application, rather than the intricacies of building portable GUIs.

Challenges Posed by Web Applications

As with any application architecture, the web poses its own set of challenges:

1. *Web "languages"*

A number of new languages have evolved with the web. Some languages were built specifically for networked applications (such as Java) and some have been extended to support the web (such as Basic). Java is designed to be portable whereas Basic is restricted to Intel-compatible platforms. The problem with many of the languages is that they are still in their infancy and are lacking in many areas. These areas include limited control over visual appearance, performance and, most importantly in terms of transactional systems, support for database interaction. Although it is expected that the major languages will quickly evolve to reach the robustness needed for the majority of applications, there is also a programmer learning curve associated with that growth.

Given the high publishing content on the web, publishing formats have evolved as well. By far the most popular format is HTML which is used as the basis for presenting textual information on the majority of web sites. Also popular, is Adobe's PDF format, offering fidelity and control that is currently absent in HTML, but requiring the downloading and installation of the Adobe Acrobat reader. Both formats are portable and work with all the major web browsers. Tools that support both these standards as well as the languages above, give developers the flexibility to deploy applications with the technology that is most appropriate for the job.

2. *Immediate Deployment of Web Applications*

Once organizations recognize the benefits of the web, they typically want to start deploying applications on it as soon as possible. Unfortunately, the restrictions currently imposed by web languages and the ramp-up time and short supply of skilled web programmers, make this difficult to achieve. Tools that make web developers more productive and enables them to leverage their existing knowledge, are expected to be in high demand.

3. *Web Architecture*

Client/server applications required developers to understand a new way of building applications – rather than programs executing on a single host machine, they are componentized and distributed across the network. Similarly, web applications are placing new demands on the development community. Developers need to combine their experience in host-based programming with the visual development techniques of client/server to create applications that are both easy-to-use and scalable to support potentially millions of users. It follows that to achieve these goals, developers are more likely to be successful with tools which have historically supported host-based as well as client/server applications.

4. *Transaction Processing*

Web browsers typically communicate with web servers using the http protocol. Unfortunately, http is not able to maintain state across connections. This has a potentially severe impact on database applications as it is critical that, for example, orders are not committed to the database unless all line items on the order are valid. To circumvent this problem, tools that are database-aware and can interact directly with a database provide distinct advantages to developers building transaction processing applications on the web.

Tools for Information Publishing

Oracle offers two development tools for building and deploying publishing applications on the web: Designer/2000 and Developer/2000. Developer/2000 is a visual development environment for building database-oriented applications. Designer/2000 is a business modeling and design tool from which applications can be generated, including Developer/2000 applications.

Designer/2000

Designer/2000 enables developers to create visual representations of their applications -- database and application module diagrammers are used to define both server-side and client-side application definitions and logic. These definitions, together with templates and user-defined standards, form the basis for the generation of complete applications.

Designer/2000 makes use of the Oracle WebServer, an HTTP server tightly integrated with Oracle7, to create web applications. The Oracle WebServer retrieves data from the database and formats it into HTML pages using the PL/SQL language, Oracle's standard extension to SQL.

The process of creating a web application using Designer/2000 is as follows (assuming a database has already been defined):

1. A module is created, defining which database objects to use, how to use them, and how different objects are related to one another.   Links are created between multiple modules to enable navigation through multiple screens (for example, from master to detail information)
2. The module definitions are used to generate PL/SQL packages which are then installed in the Oracle WebServer.
3. The PL/SQL packages, are used to generate dynamic HTML pages which incorporate the database access set up for that module and the layout and format defined using the templates and standards in Designer/2000. The HTML pages can be viewed using any web browser.
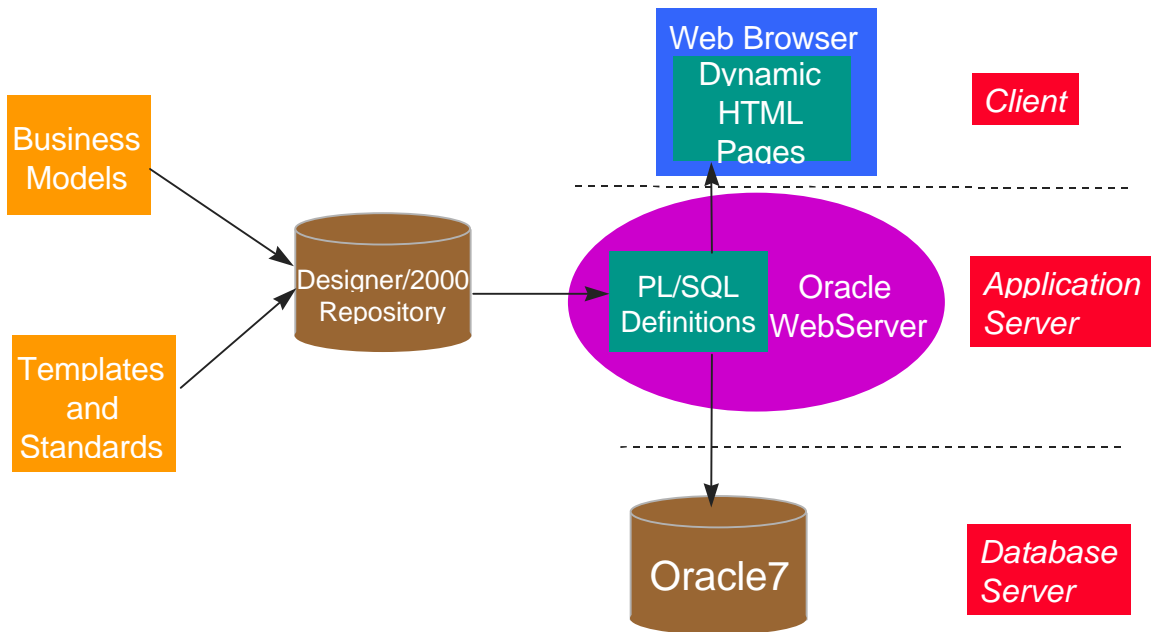
Business Models

Templates and Standards

Designer/2000 Repository

Web Browser
Dynamic HTML Pages

Client

PL/SQL Definitions Oracle WebServer

Application Server

Oracle7

Database Server

Fig 2: Designer/2000 WebServer Generator

Web applications can also be generated from previously-defined modules in the Designer/2000 repository, or even from existing Developer/2000 or Oracle Forms applications that have been reverse engineered into Designer/2000.

Developer/2000

Developer/2000 is used by developers around the world to create portable, multi-lingual enterprise-class client/server and character mode applications. Developer/2000 is uniquely able to leverage an Oracle7 database by enabling scripting of application logic in PL/SQL (the same language used for server logic in Oracle7). It also includes components for building complete applications including the screens, reports and charts. Developer/2000 applications run on multiple platforms including Windows, Macintosh, Motif, OS/2 and character mode. Given the product's portability, and the shift towards server-side processing that the web entails, Developer/2000 is well-positioned to enable the development and deployment of web applications.

Historically, reports built using Developer/2000 have been generated as postscript or text files. These reports can now also be generated for the web in HTML or Adobe's pdf format. As mentioned above, HTML is more commonly used on the web, but the pdf format provides a degree of presentation quality and precision that is critical for publishing certain types of information.

Reports for the web are created in Developer/2000 in the same graphical point-and-click way as reports for any other environment. Developers already familiar with the product can thus publish reports on the web without any additional training. In fact, even existing reports can be turned into web reports by simply re-running them specifying HTML or pdf as the target output.

Developer/2000's web publishing capabilities include features that enable an intuitive and visually impressive interface:

- a URL can be associated with a field in a report enabling a user to jump to another web location by simply clicking on the field
- reports have drill-down capabilities such that a user can navigate through a series of generated reports by, again, simply clicking on the field. In Developer/2000 Release 2, it will be possible to generate and run the reports on a server, so that a reports can be dynamically generated and uploaded to the browser as they are needed.
- charts created by Developer/2000 can be exported into supported web formats (such as GIF or PICT) and embedded within reports that are published
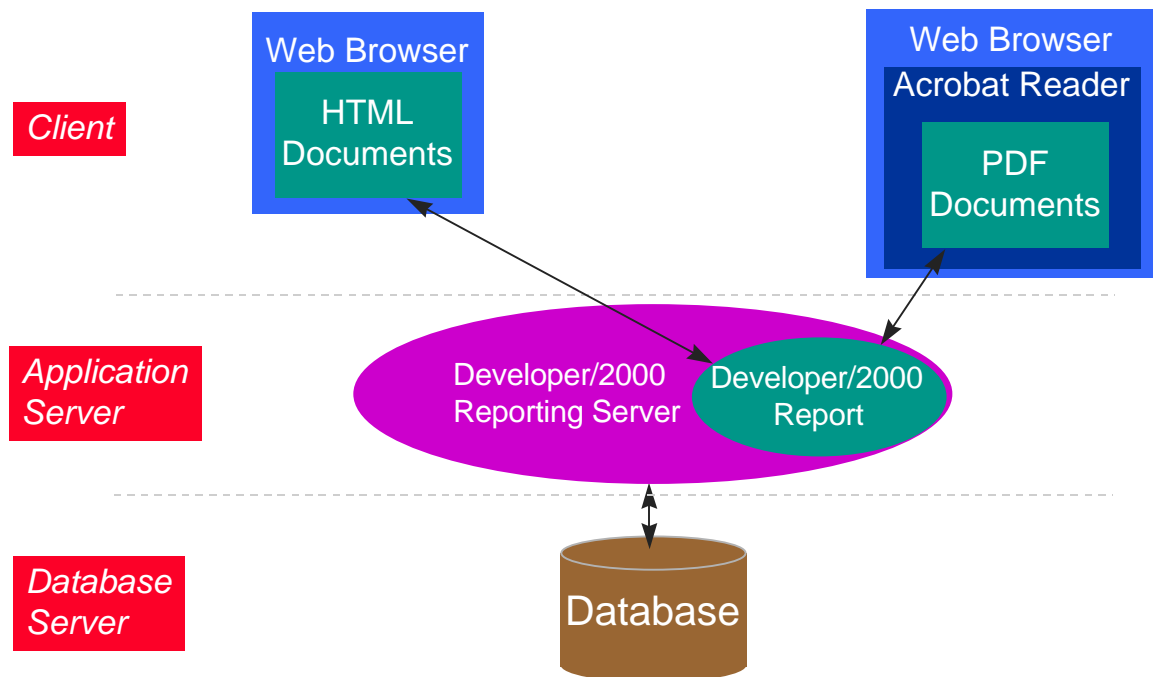

Fig 3: Developer/2000 Web Publishing

Tools for Transaction Processing and Electronic Commerce on the Web

As mentioned earlier, the majority of web applications today are publishing applications and any transaction processing is usually limited to simple, record-at-a-time processing. Oracle will provide three tools for building web applications containing complex transaction processing, starting in the second half of 1996: Designer/2000, Developer/2000 and Power Objects. These tools all have a high degree of built-in database processing intelligence, and creating data centric transactional web applications is a natural fit for their ability.

Although electronic commerce on the web involves considerations over and above those of standard transaction processing applications, in principle their functioning is very similar. Oracle is committed to supporting security standards and meeting any other requirements that make the functionality described below effective for electronic commerce applications.

Designer/2000

In addition to generating web publishing applications, the Designer/2000 WebServer generator described above will also be able to create transaction processing applications. The generation process is the same, except that the PL/SQL definitions created in the Oracle WebServer are now capable of performing a complete set of data manipulation functions. These functions enable the dynamic HTML documents to query as well as insert, update and delete data from the database.

JavaScript is also generated as part of the application to help drive a user's interaction with it. For example, a set of buttons could have JavaScript code associated with them to allow a user to scroll through records retrieved and to perform database operations such as inserting or deleting records. Designer/2000 is the only tool on the market that not only delivers complete web applications, but does this without the developer having to write a single line of code or learn a new programming language.

Developer/2000

There are two driving forces behind the direction of Developer/2000's web support: the fact that users want to leverage their existing investment in Developer/2000 applications and client/server expertise, and the emergence of Java as the de facto language for deploying applications on the web. Building on its ability to run on a wide range of server platforms such as Unix and Windows NT, Developer/2000 will be able to create web applications whilst preserving current investments and utilizing Java.

The process of creating a web application is, once again, as simple as building an application in Developer/2000's GUI development environment and simply generating it for the web. Existing character mode or client/server applications can be re-deployed on the web by simply regenerating them. When a web application is generated, it will create a Java application in addition to the Developer/2000 executable that is created today – the executable runs on a server and the Java application is downloaded on request by a web browser.
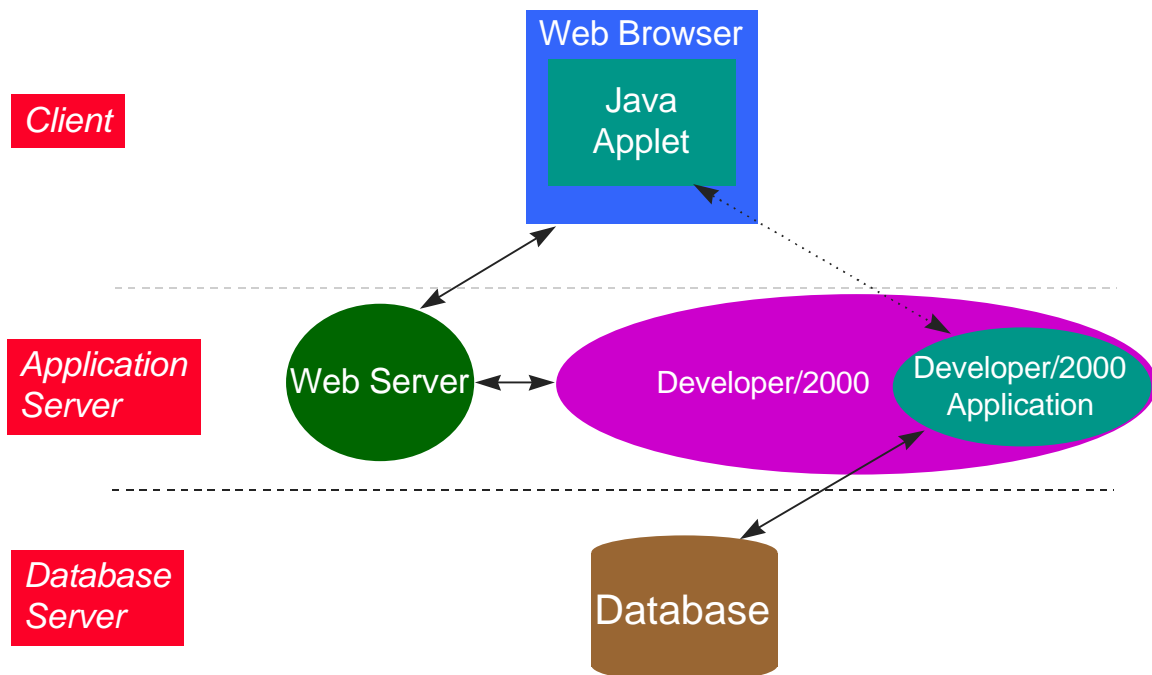

Fig 4: Developer/2000 Web Applications

Developer/2000 applications can be deployed on the web without coding,

Apart from the coding-free transition of applications to the web, a key advantage of this approach is that it leverages the advanced database technology in the Developer/2000 processing engine. Web applications automatically inherit the same scalability attributes of existing Developer/2000 applications. As with the functionality described above, the Developer/2000 application does not need to be installed on any client machine, but is located portably on major server platforms such as UNIX and Windows NT.

Power Objects

Power Objects, Oracle's workgroup client/server development tool, has two new features for developing intranet and internet application. Firstly, it takes advantage of browser plug-in technology to make applications available on the web. Power Objects' ability to operate as a plug-in enables developers to easily build portable applications and move existing applications to the web. Applications running inside a web browser have the same look-and-feel as before, eliminating the need to retrain end users. Since Power Objects uses BASIC as a scripting language, the many developers already familiar with BASIC can easily take advantage of its web support.

Through Power Objects' support for OCX controls, a web browser OCX (such as the Oracle PowerBrowser OCX) can be embedded within a Power Objects application. This enables any web site to be accessed directly from within a Power Objects application with minimal effort. Developer/2000 also provides this functionality.
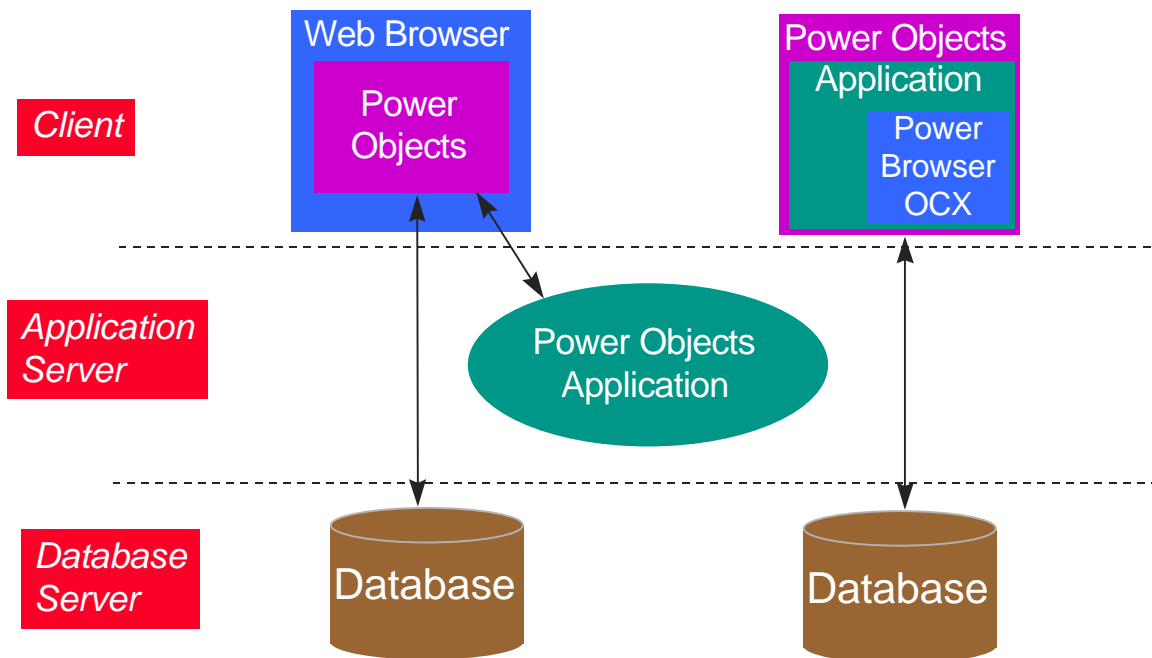
Regardless of whether Power Objects is used as a plug-in or whether other web functionality is included inside a Power Objects application, the product leverages its strongly integrated database support to execute transactions dependably and efficiently. Building web applications with Power Objects is thus quick, easy and does not require a steep learning curve.

Summary of Key Deliverables

Oracle's tool set delivers on the benefits of the web and overcomes its shortcomings in a number of different ways:

- **Reduce maintenance costs.** Neither WebServer applications generated by Designer/2000 nor Developer/2000 web applications require the installation of applications on client machines in order to operate. They enable centralized management and easy administration. Both products are also perfectly positioned to run on network computers.
- **Reduce resource requirements.** Since applications are executed on servers, hard disk space and RAM requirements are lowered on each client machine, significantly reducing the cost of hardware required to support an organization's applications.
- **Run on all major platforms.** Developer/2000 applications run on Windows, Windows 95/NT, Motif, Macintosh, character mode terminals and now, under web browsers as well, adopting the native look-and-feel of each platform. WebServer applications utilize HTML and Javascript and run in a browser on any platform which supports these standards. Power Objects runs on Windows, Windows 95/NT, Macintosh and as a web plug-in, providing support for customers who prefer a consistent look-and-feel on each specific platform.
- **Support for standards.** All of Oracle's tools support and take advantage of web standards. Designer/2000 generates HTML applications. Developer/2000 provides for client/server applications to be converted to Java, HTML, and pdf, and along with Power Objects, supports the embedding of OCX and ActiveX controls inside applications.
- **Eliminate the learning curve.** Oracle's tools are built for productivity. Designer/2000's unique ability to generate web applications without writing a single line of code is unmatched in the industry. Developer/2000 and Power Objects use a familiar drag-and-drop development approach coupled with exceptional database integration to deliver highly functional data processing applications that can scale to meet even the rigorous demands of the internet. Regardless of the tool selected, developers are assured of getting web applications up and running extremely quickly without incurring the overhead of learning new languages and development approaches, while still leveraging everything the web has to offer.
- **Support transaction processing.** Developer/2000 is an accomplished database development environment enabling the processing of complex enterprise transactions. Power Objects makes database interaction extremely simple for tactical transaction processing. Both tools bring this processing power to the web without being constrained by its current limitations. Regardless of whether users need transaction processing on intranets, or whether support for electronic commerce is required, Oracle has the tools to deliver the goods.

Conclusion

The support that Oracle provides for developing web applications, is another feather in the leadership cap it has shown for many years. Oracle was the first to move users from 3GL legacy applications to character mode 4GLs, and from 4GLs to portable client/server systems. Now Oracle is the first to offer users the ability to transition investments in older technology to web state-of-the-art technology. Designer/2000 executes this transformation more productively than any other tool. Developer/2000 does it portably and with the scalability and robustness users have come to expect from Oracle. And Power Objects does it with minimum effort and maximum efficiency. Regardless of user needs or demands, Oracle is and will continue to be the leader in application development.