RAPID DEVELOPMENT OF DATABASE INTERFACES WITH ORACLE APEX, USED FOR THE CONTROLS SYSTEMS AT CERN

Z. Zaharieva, R. Billen, CERN, Geneva, Switzerland

Abstract

The need to rapidly prototype, build and deploy applications and to be able to react immediately to the changing user requirements is a challenge facing every enterprise. CERN – the largest particle research centre in the world – has tremendous data storage requirements, encompassing many different databases and has to quickly provide interfaces to visualize the data. This article will cover how Oracle APEX has been used to build several different database-centric interfaces related to the accelerator complex. Real-world applications will be discussed and it will be shown how Oracle APEX has met the preliminary requirements of the application developers and the user community at CERN. The article will address the question of when APEX could be a suitable choice of application development technology, and will share a developer's first-hand experience of both the good and bad points.

INTRODUCTION

The amount of technical data, necessary for the control of the CERN accelerator complex is enormous. In order to better manage this data, CERN was one of the first European companies to purchase the Oracle® Relational Database Management System (RDBMS) in 1983. Nowadays it is the standard RDBMS used for data management needs in the inter-related domains of the CERN accelerator environment. In a set of distributed databases, Controls Configuration, Operational, Layout and Assets data are managed coherently [1].

THE NEED FOR USER INTERFACES

From the central control room, the accelerators are operated by means of high level graphical User Interfaces (UI). These software applications are directly driven by data from the mission-critical or 'on-line' databases, such as the Controls Configuration database. Two types of interactive man-machine interfaces to the technical data are usually necessary. The first one allows control room operators or equipment experts, having the appropriate data access rights, to read and modify the accelerators parameters. In order to provide feature rich interfaces, the data manipulation interfaces are primarily written in the Java programming language and following the J2EE standards.

Data Browsing Interfaces

The second type of interfaces comprises of read-only applications (reporting tools), whereby the data only needs to be consulted. The complexity of these interfaces is reduced, as the applications are only 'window-on-data' and do not have to implement elaborate business logic to modify the data. It has to be pointed out, however, that sometimes more complex business logic is necessary in order to create advanced reports.

The reporting tools must also provide excellent navigational capabilities and access to data from disparate locations.

Another challenge for the reporting tools is that they are utilised by a very large user community – operators, equipment specialists, controls systems specialists, etc. and must meet the diverse needs of these various groups.

EMBRACING NEW TECHNOLOGY

The data browsing tools must be available from virtually anywhere at CERN due to the dispersed location of their users. Historically web-deployed pages using Oracle Web Access – OWA (PL/SQL web-extension) have been used extensively in order to meet this need. Despite the fact that this technology is not yet obsolete, there is a clear move within CERN towards the state-of-the-art technologies and more productive development environments.

The choice of the most appropriate tool has to be evaluated taking into account the business needs, the development team's skills and the cost of ownership [2].

Oracle Application Express (APEX) was chosen in 2004 for the implementation of data browsing interfaces related to the controls systems mainly due to the ease of use of the framework and rapid development cycle. The technology was exploited for new developments as well as for re-engineering of existing PL/SQL OWA interfaces.

APEX is a web-based 4GL graphical application development environment. Today, it is a standard feature of every Oracle database.

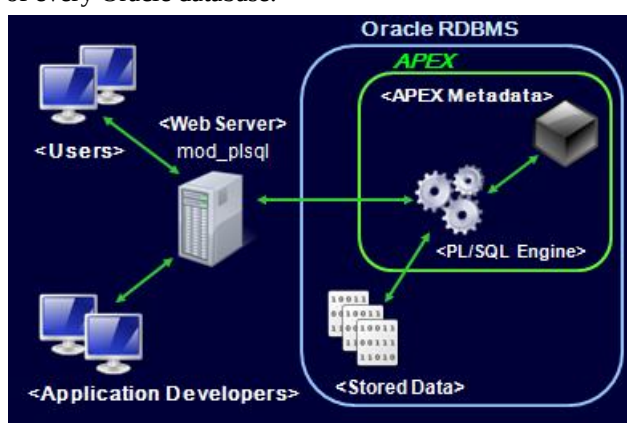

Figure 1: APEX architecture.

The application is stored as metadata in a repository in the database. To access the application the user needs only a web-browser, which contacts a web-server via

HTTP(S) and the APEX engine renders the application in real time from the metadata and the data stored in the database [3]. It is important to note that in order to build and deploy an APEX application the developer does not need any other piece of client software but a web-browser.

ORACLE APPLICATION EXPRESS – WHY

The control and operation of CERN's accelerators is a very dynamic environment with new and changing user requirements coming regularly, directly impacting the data browsing tools. The best software development style to suit such an environment is one of agile programming, facilitating fast prototyping, short time to production deployment, coupled with iteration to achieve a feature complete solution. The APEX environment allows for such agile development through its focus on simplifying the development cycle. The framework provides ready to use widgets and page components that serve to speed the development process, as well as templates and libraries to ensure the professional look-and-feel of the applications.

It is worth noting that the UI requirements of the reporting tools in the Controls environment could be achieved by using an HTML-based interface. APEX UI derives its presentation power from the capabilities of modern web browsers and is only able to achieve what is possible with HTML itself and extension of those features through Dynamic HTML (DHTML) and JavaScript. Developers can easily incorporate Web 2.0 functionality in the APEX-based applications, since the framework provides built-in declarative Web 2.0 capabilities [4].

All data exposed through the reporting tools is already stored in Oracle databases. The use of APEX allows developers to fully utilise all database objects, complex queries, to call PL/SQL stored packages and procedures thus exposing the power contained within the database.

Finally, reliability of the reporting UI is a major consideration since they must be available whenever the accelerators are in operation. APEX has proven itself to be a very stable and reliable technology. APEX is itself a service of the database servers and adds no additional point of potential failure.

START QUICKLY AND GAIN CONFIDENCE

The transition to the use of APEX in 2004 by the developers of the data browsing tools in the Controls environment was rather smooth. This was mainly due to the availability of skills in the areas of SQL and PL/SQL, as well as prior experience of development in another 4GL environment – Oracle Forms. The knowledge of web techniques such as HTML, CSS, JavaScript and AJAX contributed as well to the quick start in using APEX.

Finally, as a developer starting to work with the framework for the first time it is very easy to begin due to its use of wizards and declarative logic.

DEPLOYING APEX APPLICATIONS IN THE ACCELERATORS COMPLEX

During the last five years all newly developed reporting tools and renovation of existing ones have been implemented using APEX. More than 10 applications have been developed as data browsing interfaces for the Accelerators Entities and Signals Naming database, Power Converters Controls database, Open Analogue Signal System (OASIS) and others.

One of the largest applications that have been reengineered is the Controls Configuration DataBase (CCDB) Browser.

CONTROLS CONFIGURATION DATA REPORTING TOOLS

The CCDB is the heart of the Controls Systems providing data for all controls devices and the access to them for all accelerators at CERN. It also provides data for the hardware and software configuration of all front-end-computers used by the controls systems, as well as configuration of the accelerators' Timing system and several other systems. [5].

The CCDB database model has been developed and extended over the last 20 years during which time the reporting tools also evolved. Originally they were implemented using the PL/SQL OWA technology. This implementation however presented several shortcomings most notably the mixing of business logic with the presentation layer, leading to problems of maintainability and code reuse. This architecture also made it difficult to quickly respond to urgent new request. Last but not least the look and feel of the reporting pages was somewhat antiquated.

With this in mind a new approach was sought. The CCDB Data Browsing tools, originally comprising of around 80 different reports, were reengineered.

In order to ensure a uniformity of application look-and-feel and ease the maintenance workload, a system of coding standards and established best practices, such as naming conventions and common components (for example, web templates and navigation tools) were implemented within the team, improving performance and speeding the development process.

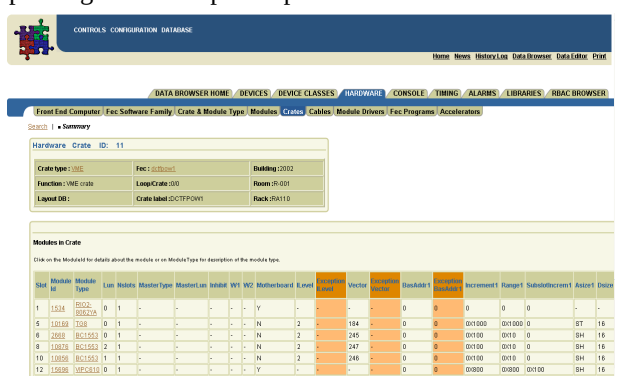

Figure 2: The new look of the CCDB Data Browsing tools.

Organising the APEX workspace is an important consideration, since there are very strict availability requirements for the browsing tools. The best quality of service for users and smooth migration effort are ensured by establishing development, test and production versions of the applications with separate workspaces, each accessing their own schema, in addition to distinct version control for each workspace and clear migration procedures. Essential to reducing the migration effort from one environment to another is the use of APEX facilities such as framework-provided or custom substitution strings and packaging the applications for a single-file, single-step deployment.

The use of APEX-provided widgets such as trees, calendars, etc. has greatly reduced the development time of the CCDB Data Browser. PL/SQL packages stored in the database were directly used as part of the APEX applications, allowing the reuse of existing business logic. JavaScript code was used to improve the user experience or to provide new features on the client side such as caching of results to avoid additional queries to the database and the use of AJAX to refresh only the necessary parts of a page.

Access requirements of the CCDB Data Browser are quite open, allowing access to any CERN user. APEX allowed very easy integration of the applications with a CERN centrally managed authentication web-service.

Securing the reporting tools against malicious users is comparatively not that difficult, since the interfaces do not allow data manipulation to start with and the only data entry fields of the interface serve as search criteria to establish the queries. The security checks for the CCDB Browsing tools are to protect them against SQL injection, XSS (cross site scripting), and URL tampering [6].

In addition to re-implementing the existing functionality, a number of new reports and enhancements to existing ones were quickly provided due to the scalability of the framework. Nowadays the CCDB Data Browser comprises of close to 160 different reports used by a community of approximately 200 people.

PROS AND CONS

CERN already uses Oracle's RDBMS to host its databases and APEX comes bundled as a free tool as part of the Oracle database server. This also means that APEX sits as close to the data it works with as possible, allowing it to take full advantage of all of the facilities provided by the database for data manipulations.

APEX has a richly defined set of APIs, allowing for complex logic, able to meet the most advanced reporting needs. Oracle SQL Developer, already in use at CERN, provides special extensions and additional support for working with native APEX components.

APEX is operating system and internet browser agnostic. This was a major consideration for this project since it was necessary to support Windows for desktop computing and Linux for technical purposes within the control centre.

The first APEX deployment in the Controls area at CERN in 2004 was using version 1.6 of the framework. Today version 3.2 is in use and so far version upgrades have been a flawless experience with deployments never having presented any difficulties and backwards compatibility of features has always been ensured. There is also a very active user community, sharing tips and solutions and utilities through APEX support forums.

One potential inconvenience of APEX, however, is its lack of native version control integration. It is possible to perform a manual export of an application, and certain tools have been created to assist this process. However such exports must either export a whole application file or else individual pages. Exporting individual pages quickly becomes tedious as an application grows in size, whilst exporting a whole application makes it impossible to adequately control individual modifications to the application. Of course if the changes in question concern not the APEX interface components, but PL/SQL code stored in the database, then this problem disappears.

Another point to consider when using APEX applications is the potential increase in the database connections established between the client machines and the database servers due to the two-tier architecture.

CONCLUSION

Oracle Application Express is a mature development tool optimized for building web-enabled window-on-data type applications. Oracle development expertise is a prerequisite for rapid and effective results, especially for more complex or relatively large user interfaces. Good programming policy and practices remain important for overall maintainability. For the data browsing tools of the CERN accelerators complex, APEX fulfills the stringent application needs. The APEX user interfaces are continuously used by a wide user community of accelerator operators and equipment specialists.

REFERENCES

[1] R. Billen *et al*., "Accelerator Data Foundation: How It All Fits Together", ICALEPCS'09, Kobe, Japan, October 2009, TUB001.

[2] M. Riley, "Choosing the Right Tool", Oracle Magazine, July-August 2009.

[3] http://apex.oracle.com.

[4] D. Peake, "Express Web 2.0", Oracle Magazine, Sep-October 2007.

[5] J. Cuperus, R. Billen and M. Lelaizant, "The Configuration Database for the CERN Accelerator Control System", ICALEPCS'03, Gyeongju, Korea, October 2003, WE114, p. 309 (2004)*.

[6] D. Peake, "Developing Secure Applications", Oracle Magazine, July-August 2009.

* Published at the Joint Accelerator Conferences Website (JACoW) http://www.JACoW.org