TechnologyPACER

No. 8 The Technology Newsletter of GMD FOKUS for the PACE Project December 2000

WEB INTEGRATION TECHNOLOGIES

Even if a mature distributed object-oriented (OO) infrastructure is in place and ready to be used, there are to date no clear ways to integrate back-end systems with the Web and provide dynamic content, i.e. content that is related to the data and processes stored in or provided by the back-end system. The Web currently operates with static web site skeletons where data is simply displayed; the provision of dynamic HTML pages and the separation of the presentation view (HTML) from the data/functional view (EJBs and Business Objects) is often not possible.

Dynamic Content

From the earliest days of the Web, developers have sought to add dynamic content by running logic on the server. The first important step was the Common Gateway Interface (CGI). When a web server receives a CGI request, it creates a new process for the program. A multiplicity of requests, though, leads to unacceptable performance.

To simplify the task of server-side programming, vendors have introduced other technniques for creating web applications, e.g. Active Server Pages (ASP) and server-side JavaScript (SSJS), which are proprietary solutions that work only with certain web servers. JavaScript is Netscape's cross-platform, object-oriented scripting language. The Core JavaScript - the base JavaScript language - contains a core set of objects, a core set of language elements, control structures, and statements, and it can be extended for a variety of purposes by supplementing it with additional objects.

Java Applets

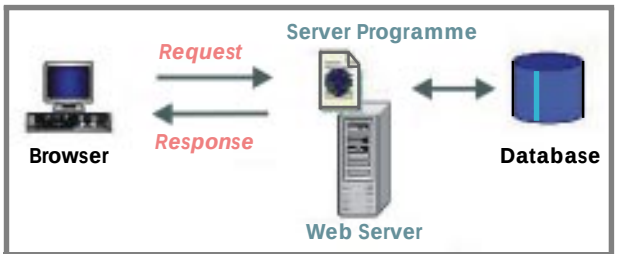
Java Applets, proposed by Sun Microsystems, is one of the most successful solutions. Applets enable the seamless integration of Java programmes into web pages, allowing for the development of innovative services and applications. The most significant problems of Applets are security limitations, distribution, and programming language dependability. As a solution to Applet problems, Java came up with the idea of Servlets.

Java Servlets

Servlets is a good integration technology for server-centric developments and seamless deployment of back-end systems and modules. Just as an Applet is a Java plug-in for a web browser, a Servlet is a Java plug-in for a web server. Servlets greatly improve portability. Being Java programmes, they do not have to be recompiled for different operating systems. Due to this fact, and because Java contains the network, security, and enterprise APIs, the Servlets are of great importance for the market.

The weakest points, however, are the limited interactivity, the unclear way of separating presentation and functional view, the dependency on Java, and the availability of commercial web-servers that have a Java Servlet engine incorporated. During the last year, significant developments towards Servlets have been achieved in comparison with existing traditional CGI approaches. Servlets and Applets are more flexible and dynamic than CGI and are starting to dominate the web centric developments.

A typical web application will collect data from the user (first tier), send a request to the web server, run the requested server program (second tier), package up the data to be presented in the web browser (third tier), and send it back to the browser for display (first tier).



WEB APPLICATION ARCHITECTURE

Continued on Page 2

Call Center Challenges

The term "workforce management" sounds simple, but in the call center world, putting management into action is complex. Managers must achieve accurate forecasting, scheduling, monitoring, training and other performance-related functions. And they've got to do it all to the satisfaction of call center agents.

The executive vice president and co-founder of California-based Hiho Technologies says workforce management failures result in disgruntled agents, severe bottom line consequences, and frustrated managers. Call center managers are under increasing pressure to retain good agents in an industry that is growing an estimated 20% annually, but encountering an annual agent turnover rate of 31%. At the same time, managers must meet higher productivity and customer service goals.

The vice president of CENTRIQ Inc. points out that managers must provide the right training to implement new IP-based technologies. Problems arise when call centers add e-mail, Web chat, VoIP, and other communications options to their lineup. Companies expect increased efficiencies, improved customer service, and financial savings from these investments, yet often find the opposite results. What had been a fairly simple operation for these companies becomes a complex Internet-based system requiring sophisticated scheduling, forecasting, and training techniques. This is where workforce management software can be applied effectively. Specifically, the next level of workforce management software involves skills-based scheduling. Just as the ACD can provide skills-based routing, evolving technology can do the same for scheduling. The goal and probable result: Getting the right agent talking with the right caller.

This is an excerpt from an article by Mike Shanin found at www.CallCenterBiz.com

Hiho Technologies, Inc is a leading provider of innovative workplace management solutions. www.hiho.com

CENTRIQ provides customer-centered service in e-commerce including consultation, systems integration services, and application hosting. www.centriq.com

Document Management and Workflow

An interview with Ken Orr, former Chief Information Officer (CIO) for the state of Kansas, USA.

Q: What can document management and workflow solutions accomplish for state and local governments?

A: Most of government is involved in moving paper from one side of the building to the other. Enormous amounts of what they do involves physically manipulating data. The degree to which we don't have to do that anymore means that we can change the nature of business.

Q: What's a good strategy for accomplishing that?

A: One of the things we're learning about workflow is that it's really about managing in-boxes. Everybody has two in-boxes, one for e-mail and one for production, or their daily work. If you can manage the production in-box, you can build workflow processes into the system. This can be done by implementing an underlying workflow management engine that keeps track of whether the right things get to the right people, and the appropriate actions are taken. Industry is working on a whole new generation of project management systems that will allow you to put documents into shared folders, complete with restricted access and notification capabilities. In the next decade or so, we're really going to change people's expectations of how they work, and how they work together electronically.

Q: What are some trends in the document management and workflow software industry?

A: The companies that have historically been involved with document management and groupware products are now intersecting with companies that provide web and e-mail technologies, as well as interfacing with database companies.

Q: Are the vendors supplying customers with the tools that they need?

A: We're beginning to see a new generation of tools coming up, which include capabilities for features such as visualization. We're going to see smarter documents with smarter attachments, which means we'll need smarter workflow systems.

Q: What are the major challenges facing state and local governments that want to implement document management and workflow solutions?

A: They're coming to it late typically, but that may work to their advantage because of the pace of change in technology. The disadvantage, of course, is that state governments are very traditional and have all sorts of legal constraints. The biggest underlying technological challenge is having sufficient bandwidth to implement the systems.

Source: www.gcn.com/research_results/

Web Integration Technologies

(Continued from Page 1) Although a Servlet can be a completely self-contained program, in some cases it is useful to split the task of generating dynamic content into two parts: the business rules that govern the relationship between input, processing, and output, and the graphic design rules that determine how information is presented to the user. The business logic could be handled by JavaBeans or VisualBasic and the presentation logic should be handled by the Servlet, Java Server Pages (JSP), or Active Server Pages (ASP).

The Presentation vs the Functional View

In order to solve the problem of presentation vs. functional view, the concept of active pages was proposed. Initially, Microsoft proposed the ASP technology which adequately enables the integration of back-end systems and business objects. Unfortunately, ASP is tightly coupled with DCOM and the Internet Information Server, the web browser of Microsoft. Sun, understanding the need for better technologies, proposed a similar technology, namely JSP. JSP modules are proprietary Java commands within the traditional HTML code, resulting in invocation of object method calls becoming an easy and straight forward process. Both ASP and JSP are good integration technologies for dynamic content and dynamic object method invocation. The objects can be distributed in different physical locations. The separation between the presentation and functional view, unfortunately, is still not the best possible solution.

Related URLs:

http://msdn.microsoft.com/workshop/server/

http://java.sun.com/products/jsp/

http://java.sun.com/products/servlet/

http://www.apache.org

http://www.w3.org/

Concerning the advantages of JSP and ASP in comparison to Servlet technology, it is easy to make a change and then let the JSP capability of the web server you are using deal with compiling it into a Servlet and running it. JSP technology is designed to simplify the process of creating pages by separating web presentation from web content. JSP provides a number of server-side tags that allow developers to perform most dynamic content operations without ever writing a single line of Java code. So, developers who are only familiar with scripting, or even those who are simply HTML designers, can use JSP tags for generating simple output without having to learn Java. Another advantage of Java Server Pages is that they are document-centric. Servlets, on the other hand, look and act like programs. A Java Server Page can contain Java program fragments that instantiate and execute Java classes, but these occur inside an HTML template file and are primarily used to generate dynamic content.

XSL and XML

The W3C, understanding the problem, also proposed the concept of eXtensible Style Language (XSL). XSL is used for specifying how XML data can be represented as HTML pages. This is actually a language that transforms XML data into HTML pages according to some presentation rules, e.g. tables, headings, bullets, etc. Given the fact that XML can be used for structuring data, XSL will be used for presenting them to different terminals as HTML pages utilising the strong and well-accepted concept of web browsers.

The idea behind XSL was the following: XML data stored on back-end systems will be presented in different terminals, with different display capabilities and different abstractions levels. Maintaining the same data structure, but changing only the presentation format according to the needs of the users is the ultimate goal. Although XSL and XML are good, open, generic concepts, that adequately solve the problem of presentation vs functional view, they do not have clear relationships with existing distributed OO platforms. Thus, it is the responsibility of Java and other programming languages to support XML and XSL within their frameworks, and it is anticipated that XML and XSL will be supported by ASP, JSP, or Servlets in the coming years.

Vaggelis Ouzounis
Senior Researcher, GMD FOKUS

www.pace-eu.net

www.fokus.gmd.de/cc/ecco/

This Newsletter is published by the E-Commerce Competence Center (ECCO) of GMD FOKUS, Berlin, for the IST project Public Administration and e-Commerce in Europe
Contact: Dr. V. Tschammer (tschammer@fokus.gmd.de)
Mr. V. Ouzounis (ouzounis@fokus.gmd.de)
Ms. C. Hardey (hardey@fokus.gmd.de)