

SAS System and Efficient Integration to Advanced Web User Interface tools

Abstract

The Customers have adopted web browsers as the main window to enterprise applications. Nowadays almost all of our Customers in eNiklas demand browser based user interfaces. We have developed several HTML bound reporting systems using SAS/Intnet.

The Customers want also different kinds of applications, like budgeting, implemented in web browsers. These kinds of applications, that require more advanced user interfaces, have been implemented with SAS/AF so far. HTML is too limited for these purposes.

The progress in web browsers and web servers has opened up new advanced ways to implement user interfaces in web browser.

eNiklas

eNiklas has vast experience in the Internet and IT-consulting business. We have constantly focused on the same business idea: To help companies explore their potential to grow.

This concept has made us expand rapidly. During the last decade, we have gone from nothing to become a substantial player in the e-business solutions area. Today we are an international company. We are well established with over 10 international offices and our sights set on further geographical expansion. The ultimate goal of eNiklas is to build a great, truly global company that makes a fair profit, has a heart for its employees and is dedicated to total customer satisfaction.

Our primary focus is to provide a new category of professional services called eSolutions. Our eSolutions integrate strategy, customer information and the Internet to help clients succeed in electronic commerce, enhance relationships with customers and increase revenues. We will provide long-term relationships that enhance the value of our solutions and services, and we will endeavour to provide our clients with substantial economic value both during the initial phases of our relation and over the life of our on-going relationship (i.e. solutions, relationships, and value).

We will co-develop - with our customers and partners - highly sophisticated eSolutions based on our industrial insights combined with our unique combination of core competencies (business, analytical and information technology skills) for the global markets, and we will deliver the appropriate service to ensure customer satisfaction. Our solutions will be both cross-industry and industry-specific solutions for the Financial services industry, Pharmaceutical industry, Telecommunications, Manufacturing, Distribution/Logistics, Energy and Chemical and Public sector. As business intelligence and data warehousing pioneers, customer relationship management (CRM) experts, balanced scorecard (BSC) experts and leading-edge business strategists, we are helping clients navigate emerging electronic business opportunities and drive revenue growth.

Overview

HTML Limits

There are certain things that you can’t do with plain HTML. Javascript helps a little bit, but it is only slight improvement. The main problem is that you have to download all the needed information for one screen in one package. This can be partly circumvented by regenerating the page according to user actions. But that can not be done in every case. Let’s think a situation where there are several classifying elements to subset the report. When user have not yet done any selections there might be thousands of available values in the lowest level. Transferring these values to the browser might take
awhile. If it would be possible to download these values only if the user decides so, the user interface would be much friendlier.

HTML is also very limited of its user interface elements. People are used to Windows standards in user interfaces and they are beginning to demand usability also from applications in the browsers. Using graphics to simulate Windows components is a very clumsy way around.

**New Techniques**

**ActiveX (COM and DCOM)**

ActiveX is a marketing name for a set of technologies and services, all based on the Component Object Model (COM). ActiveX controls are COM components that can be manipulated visually by GUI development tools. These controls can be written in C++, Java, VB, Delphi etc.. An ActiveX component is self-registering and it is optimized for downloading and executing.

Component Object Model (COM) is the foundation for ActiveX components. COM is a cross-platform open standard that is language and tool-neutral, versionable and programmable. The COM architecture is a very scalable programming model. COM lets clients access components in the same process, on the same machine or even across machines (DCOM). That is to say, Distributed Component Object Model (DCOM) is just COM with a longer wire. DCOM extends interactions between components across networks.

Using ActiveX with SAS will be easiest with SAS/Interation technologies (explained later), which is available in SAS Version 8. Communication with SAS System is also possible in version 6.12 using ODBC or unstandard communication methods.

**JavaServer Pages (JSP), servlets and taglets**

JavaServer Pages (JSP) technology allows Web developers and designers to develop and maintain dynamic Web pages that leverage existing business systems. As part of the Java family, the JSP technology enables rapid development of platform-independent web-based applications. JavaServer Pages technology separates the user interface from content generation enabling designers to change the overall page layout without altering the underlying dynamic content.

JavaServer Pages technology uses XML-like tags and scriptlets written in the Java programming language to encapsulate the logic that generates the content for the page. The application logic can reside in server-based resources (such as JavaBeans component architecture) that the page accesses with these tags and scriptlets. Any and all formatting (HTML or XML) tags are passed directly back to the response page. By separating the page logic from its design and display and supporting a reusable component-based design, JSP technology makes it faster and easier than ever to build web-based applications.

The most convenient way to communicate with the SAS Server is to use JConnect classes provided with SAS/Intrnet. There are also SAS JDBC drivers for both version 6 and 8. SAS provides also JavaBeans-compliant components called TransformationBeans, which encapsulate the look&feel of normal HTML components into Java objects.

**Code behind HTML and Microsoft’s DHTML**

Microsoft Visual Studio 6.0 has also the opportunity to create Java code that controls the HTML output. Microsoft Internet Explorer 5.0 introduces a dynamic HTML model (DHTML), which allows runtime modification of the dynamic HTML content. Code behind HTML uses the Windows Foundation Classes library to encapsulate the functionality of basic HTML objects to Java classes. This makes the technology Windows-dependent.

Microsoft’s DHTML model differs from the World Wide Web Consortium’s definition. Since Dynamic HTML is still in its infancy and current implementations are experimental (http://www.w3.org/Style/), Microsoft’s has “refined” the standard. The idea of the DHTML is to give
developers a way to create dynamic Web pages. A dynamic Web page is one where the page's structure, style, or content can be changed after the page is loaded in the browser without having to request a new page from a Web server. By using DHTML, you can create a page that can interact with the user without using additional controls and without requiring multiple trips to a server to update the page.

Communication with SAS System would be much like in ActiveX.

**AppDevStudio (webAF and webEIS)**

AppDevStudio is the most convenient way to build Java applets (and applications), which communicate with the SAS System. As mentioned before, AppDevStudio provides the Java classes that can interact with a SAS server using SAS/CONNECT, RMI or CORBA. In the future, JavaServer Pages will be integrated to webAF IDE. SAS Institute uses the term ‘webAF Server Pages’ instead of JSP.

Applications that have a lot of users require a middleware server. It is used to balance the load by restricting the number of concurrent SAS sessions. The middleware server hasn’t yet been proven to be bug free so it should be investigated in the future.

**SAS/Integration Technologies**

SAS/EIT includes components that are needed to integrate SAS Version 8 with Microsoft’s COM/DCOM technologies. For example Enterprise Guide and Enterprise Reporter 3.0 have been built on this technology.

**Conclusion**

SAS supports all the new available techniques very well. The hard thing is to choose the right tool to the right task. This can be achieved by better understanding the principals in each technique.

**Trademark notes**

SAS is a registered trademark of SAS Institute Inc.
SAS/IntrNet is a trademark of SAS Institute Inc.