Paper 141

SAS Integration Technologies Overview

Mark Gass, SAS Institute Inc., Cary, NC
Steve Jenisch, SAS Institute Inc., Cary, NC
Chip Kelly, SAS Institute Inc., Cary, NC

ABSTRACT

Two key application design principles are evident in modern systems. First, server functionality should be decoupled from presentation functionality, allowing common back-end applications to be exploited across a diverse set of customized user interfaces. Second, enterprise-class solutions often entail the integration of several disparate vendor applications, such that application interoperability and flexibility of deployment are required.

Several middleware technology standards have emerged as enablers of these principles. Component-based distributed object platforms expressed by the Object Management Group's CORBA/IIOP and Microsoft's COM address these considerations within an object-oriented programming model. Similarly, message-queuing platforms expressed by IBM's MQSeries and Microsoft's MSMQ address these considerations within a message-passing programming model.

Access to and presentation from the SAS System's business intelligence, data warehousing, and decision support facilities has been extended by these middleware environments. Deployment in thin client, Web, and n-tier embedded application configurations is now possible, as is development in an expanded set of programming languages including C++, Java, and Visual Basic.

These new integration technologies deliver the enabling infrastructure supporting the next generation of solutions from SAS Institute and delivers that same infrastructure to an enterprise's solution integrators. An overview of SAS Institute's positioning of these technologies will be presented.

INTRODUCTION

The future has arrived, and is taking up two spaces in your parking lot because it's so big and bright. Upon closer examination, sunglasses recommended, you see there was no revolution after all. It was really just a series of reasonably paced steps that took you from server centric, mainframe-based application architectures, through the storms of the client/server heyday, all the way back to the web and it's own brand of server centricities. The web has spawned the next generation of change that will deliver on both the security and efficiency of the mainframe and the allure of desktop platforms and personal productivity.

This paper, and presentation at SUGI24, describes the work that SAS Institute Inc. is undertaking to ensure that users of SAS software get to ride the wake of the web and take advantage of emerging technologies that facilitate the intertwining of diverse application architectures. Just as the web leveled the playing field on the client with the ubiquity of the browser, so too are integration technologies leveling the field of differences among enterprise resource planning(ERP) software, data warehousing applications, data mining efforts, and enterprise decision support. The browser, and the web itself, is now one of many desktop vehicles available to integration technologists. A new product from SAS Institute Inc., presently known as SAS Integration Technologies, is introduced in this presentation. It provides several methods of integrating the SAS System with other members of the enterprise application architecture.

Where do we start when software is used for as many diverse applications as SAS software is? One of the goals expressed by many SAS software project managers is to be able to use object-oriented programming and distributed objects to deliver SAS System functionality to many different types of clients and applications. To meet that goal, Object Request Brokers (ORB) offer new functionality for distributed systems integration, based on two prevalent protocols – the Object Management Group's Common Object Request Broker Architecture (CORBA) and Microsoft's Common Object Model (COM). Integration of SAS software-based applications into architectures rooted in one of those protocols is made possible with SAS Integration Technologies. The first section of the presentation gives some sample use case scenarios with SAS software as either client or server in an ORB-based application.

What's next? How about tapping into the most flexible and scalable middleware, message-queuing middleware, and allowing SAS application developers to take advantage of a programmatic interface to IBM's MQSeries or Microsoft's MSMQ? The next section will demonstrate the power of linking message-queuing's asynchronous communications capabilities with the SAS System, and provide an overview of our implementation.

Directories have emerged as an essential element of security models, object distribution models, and distributed network models. Using a directory can be as simple as being reading and writing directory entries. The real benefits come when applications can programmatically interface with directory servers in order to complete a given task. SAS Integration Technologies provides just that interface, allowing SAS program developers using either the DATA step or SCL to use directory services. The next section shows how that works.

You may be getting the feeling that the process of integration requires some knowledge of quite a few different areas of specialization – and you're right. There is an entire market segment dedicated to providing the right tools for practitioners of the integration trade. What we have shown you so far is how SAS Integration Technologies gives you a good sample of those tools, designed specifically for you, the SAS application developer. The last section of the presentation introduces a new language for the SAS System that binds all of the capabilities of the new integration tools with the power of SAS software. The combination if integration technologies with the traditional strengths of the SAS System yields a powerful publish and subscribe architecture for SAS software-based applications. The final section of the presentation shows how this new scripting language can be used to deliver SAS content in response to many different triggering mechanisms.

CONTACT INFORMATION

To find this paper on the web:
http://www.sas.com/rnd/web/papers/IntegrationTech/IntegrationTech01.html

Your comments and questions are valued and encouraged. Contact the author at:

Chip Kelly SASCWK@SAS.COM