SAPinsider

This article appeared in the Oct▪Nov▪Dec 2004 issue of *SAP Insider* and appears here with permission from the publisher, Wellesley Information Services (WIS), **www.WISpubs.com**.

WIS Wellesley Information Services

Flexible Adaptation of Business Processes Using Web Services

Regular Feature

SAP NetWeaver Unleashed

Martin Huvar, SAP AG

Sven Leukert, SAP AG

How can you reduce the typically high reimplementation effort and expenditure of resources that normally accompany the reconfiguration of any business process?

Consider this business scenario: As a service to its business customers, large and small, a bank performs credit limit (or worthiness) checks in the context of the credit approval process. To do this, the bank's IT team has implemented Web services, which suit this process perfectly — they offer an open standards-based approach so that business customers can access credit check information directly from their own company's online or other applications.

For smaller businesses, the bank's simple check is based on credit card and basic order information; think of a travel agency checking the credit of a client purchasing a family vacation. For larger, corporate customers, the bank performs a more complex credit limit check; imagine a large-scale credit application in a construction project, for example. This is performed by an internal bank system built just for this purpose with an in-house, mainframe-based risk management system.

However, customer needs inevitably change: the travel agency begins to sell more elaborate group packages, and these transactions will need the complex credit checks that the bank already offers its larger customers. Because this process is already in place, it shouldn't require excessive time, expense, or resources to adapt and implement — right?

Unfortunately, infrastructure and application design can put up big barriers to process improvement. In some system landscapes, such a seemingly minor process change would have significant repercussions: new integration costs, costly changes to both the client and server applications, completely rerouted communication paths, possibly even a complete replacement of applications. But with Web services and SAP NetWeaver — including SAP Web Application Server (SAP Web AS) and SAP Exchange Infrastructure (SAP XI) — firmly in place, this example of a business process can be adapted with minor changes to communication paths, and without affecting the bank customers' view of the process at all.

Once you have such an infrastructure, you gain the flexibility to reconfigure a business process from year to year or from one quarter to the next, without disrupting customers or end users and with lots of reuse to contain the costs of change.

Before we delve into an example of how such a system landscape can help an SAP customer easily adapt its processes, let's look at the system infrastructure itself.

Start with an Infrastructure That Can Build On Your Current Business Processes

In our example, the bank already has most of the main components of SAP NetWeaver in use somewhere in its IT landscape — be it the SAP Web Application Server as a platform for SAP applications or custom development, the SAP Enterprise Portal for providing a unified user experience across several applications, or the SAP Exchange Infrastructure for integrating processes across the landscape (see sidebar on the next page). The bank has also started to develop its Web services based on the Enterprise Services Architecture to support reuse, flexibility, and adaptability of working business processes (see sidebar on ESA on the following page).

A Services-Enabled System Landscape

In our example, the system landscape contains a banking application built on top of the SAP Web Application Server (SAP Web AS), as well as a risk management system on a legacy platform. Here's how the credit limit checks work at the start:

- The simple credit limit check is offered as a Web service through the Web service infrastructure provided with SAP Web AS. That is, a SOAP message from the travel agency (based on a previously published WSDL document describing the service) suffices to get the required response back.
- For the corporate customers, a similar functionality is offered. The difference is that content is mapped internally at the bank and is routed to the risk management system via SAP XI.

The credit limit checking services as implemented in the bank's overall IT landscape are shown in **Figure 1**. The bank's mainframe-based system has been SOAP-enabled and connected to SAP XI so that various other applications can also access the system and include it in more complex business processes.

In addition, a central interface and maintenance strategy has been adopted based on the Integration Repository of SAP XI. All interfaces are described in this repository and can be used in the context of the Web services framework or in conjunction with SAP XI's Integration Server.[1]

Web Services Capabilities of SAP NetWeaver

SAP NetWeaver provides a large set of capabilities that support the creation and usage of Web services. Most are offered in the context of the Web services framework of the SAP Web Application Server, and by the process integration functionality of the SAP Exchange Infrastructure.

SAP Web Application Server allows companies to extend their solutions by exposing and integrating Web services according to their available development skills and needs independent of the technical environment. This includes:

- Full support for Web service standards (WSDL, SOAP, UDDI, WS-I, etc.)
- Efficient development, testing, and error correction cycle
- Consistent architecture and identical feature sets for the two environments used in SAP NetWeaver (ABAP and Java)
- Support of Web service development for new applications, and based on existing applications and standard SAP interfaces

SAP Exchange Infrastructure (SAP XI) is SAP's infrastructure for message-based process integration to connect systems from different vendors (SAP and non-SAP), in different versions, and implemented in different programming languages. This supports "value-added" Web services. For example, when a simple Web service with a fixed sender and receiver, and with matching interfaces on both ends, does not suffice, SAP XI can provide additional functionality for more complex scenarios:

- Routing of messages (Web service requests) based on message content, so that different receivers can actually fulfill a Web service request
- Mapping of message content between differing sender and receiver interfaces
- Central interface schema and mapping storage and maintenance
- Central configuration of messaging participants, communication channels, etc.
- Enablement of "legacy systems" by adapter integration

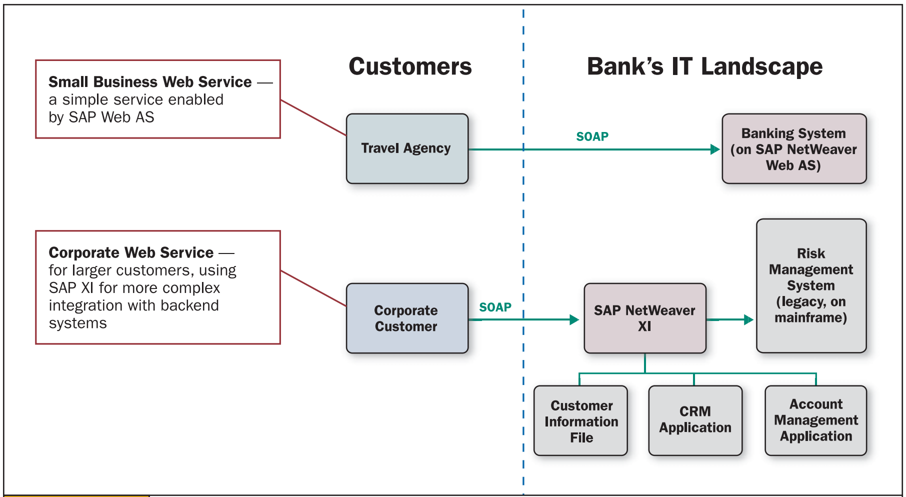
Customers
Bank's IT Landscape
Small Business Web Service — a simple service enabled by SAP Web AS
Travel Agency
SOAP
Banking System (on SAP NetWeaver Web AS)
Corporate Web Service — for larger customers, using SAP XI for more complex integration with backend systems
Risk Management System (legacy, on mainframe)
Corporate Customer
SOAP
SAP NetWeaver XI
Customer Information File
CRM Application
Account Management Application

Figure 1 Starting Point: Two Separate Processes for Different Types of Customers

[1] For more on the Integration Repository and Integration Server in SAP Exchange Infrastructure, see "SAP Exchange Infrastructure 3.0 — Enterprise Ready!" by Anders Ranum in the April-June 2004 issue of *SAP Insider* (www.SAPinsider.com).

How ESA Supports Changing Business Processes

Enabled by SAP NetWeaver, Enterprise Services Architecture (ESA) helped the bank in this example adapt the IT infrastructure to changing business needs. ESA is designed to support the principles of reusability, modular design, separation of interfaces from backend systems, and change or creation of new composite applications based on existing building blocks.

In our example, the bank can more easily alter the underlying process for the credit-limit-check functionality, without impacting the applications "using" the service, because of their approach to application design. Further, by using Web services for reusability and including current functionality into services-based business processes, the Enterprise Services Architecture blueprint enables customers to see their existing systems as the building blocks of future business processes.

The bank took the first step toward ESA by implementing SAP NetWeaver components, as an integrated platform based on Web services and the foundation for innovative cross-enterprise processes in the future. For instance, SAP XI and the Web service infrastructure can use the same mechanisms and structures, allowing the bank to easily provide and consume Web services — either point-to-point or mediated via SAP XI — and to move from one to the other without great efforts by its IT team.

Decoupling the client's view from the underlying systems also provides unprecedented flexibility. The existing functionality of a business application, such as the complex credit check, can be added to another service easily using SAP XI. Customers can also change the SAP-delivered process by adding or deleting process steps, altering user interfaces, changing participants (single expert/expert team), or changing the sequence of services during the design time of the process.

From a one-time investment in service enablement and SAP NetWeaver deployment, customers can get repeat benefits in easily being able to build and deliver a variety of solutions.

✔ Note!

In the SAP environment, Web services can be created solely by configuration. No implementation or change of existing coding is necessary. With just a few mouse clicks, the Web services creation wizard creates the credit-limit-check Web services within seconds based on the existing interface of the application.

An Ideal Environment for Developing and Implementing Web Services

To develop Web services, the bank's application developers use the Web service framework of SAP Web AS.[2] Based on the standard credit-limit-check interfaces of the application maintained in the Integration Repository, the application developers create a Web service.

On the client (or consumer[3]) side — the application being used by the bank's customers to request the credit check — any Web service-enabled environment can of course be used, whether it is a .NET, JSP, or C++ application, or SAP NetWeaver. In the case of SAP NetWeaver, SAP Web AS offers developers comprehensive support on the client side, from client proxy generation to integration with dynamic and comprehensive user interfaces created with Web Dynpro.

Adapt Processes with Minimal Impact on Your Applications and Interfaces

Let's say the travel agency begins selling some high-end group packages to its business clients, which require a more involved credit check. This means retrieving exactly the same information gathered for the bank's larger customers. The catch is that the travel agency and other small businesses only need this service on a case-by-case basis — only the travel agent's biggest accounts require this elaborate check.

This means the small business credit check process needs more intelligence: based on the content of the transferred data, a routing decision must determine whether to call the same simple Web service as before or to call the more elaborate service on the risk management system. In the latter case, the data also needs to be mapped to the interface of the mainframe-based system.

With our sample landscape, it's purely a matter of redirecting the SOAP call from the smaller customers to the SAP XI entry point and setting up the routing and mapping within SAP XI. From here, SAP XI will determine where the request will be forwarded. The client and server application can be

[2] The Web service framework supports both ABAP and Java and offers all the tools needed to enable SAP's standard interfaces (BAPIs, RFCs, IDOCs, EJBs, Java Classes) as Web services.

[3] Recall that the Web service world is divided into providers (server) and consumers (client). Those who call and use a Web service are consumers; the bank in this case is the Web service provider. For more information, see Martin Huvar's article "Web Services for Platform-Independent, Standards-Based Information Sharing" in the October-December 2003 issue of *SAP Insider* (www.SAPinsider.com).

reused without adjustment. What's more, the risk management system is already connected to SAP XI, since it was used for the corporate customer credit limit check. The end result? To the bank's customers, nothing changes except for extended functionality in the credit check service.

The Four Steps to Changing the Process

Since SAP XI is already present in the system landscape and used to integrate the mainframe-based risk management system, and since all the interfaces have already been modeled in the Integration Repository, the effort for reconfiguring the small-business scenario is minimal:

- For the bank to receive messages, the travel agency needs to be recognized (configured) as a message sender or receiver in the Integration Directory of SAP XI.
- Messages (Web service requests) received from the travel agency need to be routed to the correct receiver system and interface depending on the message content. To this end, receiver and interface determination rules are configured in the Integration Directory.
- The technical channel for exchanging messages with a particular system or business partner needs to be configured — in this case, it is SOAP-based message exchange with the travel agency.
- Finally, XML messages received from the travel agency and sent to the risk management system need to map from sender to receiver interface. A Java mapping program generates the message and interface mappings designed with the graphical mapping tool of the Integration Repository.

Once these four steps are performed, the switchover is complete (see **Figure 2**) — all without causing any changes at the small business and the large corporate customer sides.

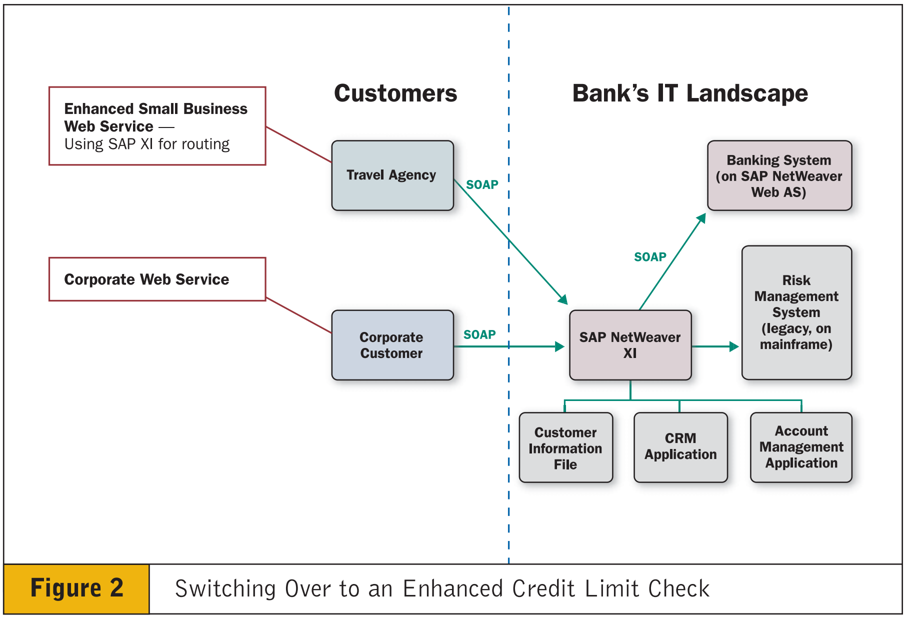
Enhanced Small Business Web Service — Using SAP XI for routing
Corporate Web Service
Customers
Bank's IT Landscape
Travel Agency
Corporate Customer
SOAP
SOAP
SOAP
Banking System (on SAP NetWeaver Web AS)
SAP NetWeaver XI
Risk Management System (legacy, on mainframe)
Customer Information File
CRM Application
Account Management Application

Figure 2 Switching Over to an Enhanced Credit Limit Check

Conclusion

This scenario shows some of the capabilities of SAP NetWeaver and how they support effective use of Web services — a big step toward higher-level services that can handle end-to-end processes that are described and supported by Enterprise Services Architecture. SAP NetWeaver already takes some great strides toward technology and resources for bringing reusability, flexibility, and adaptability — all core principles driving Enterprise Services Architecture.

With no great effort, this scenario was enhanced for a new and more sophisticated process by integrating and reusing existing technologies (SAP XI) and functions (risk management system). Without affecting the bank's business customers, and without changing the banking system, a more valuable function for serving business needs was established. Within just a few days, a new process like this can be deployed to productive use. This is just one example of how SAP NetWeaver and ESA are more than an application platform — they are the enablers of change for future business needs.

Web services are just one step toward this vision of greater adaptability and flexibility in your processes. SAP NetWeaver technology and the design principles of ESA are also a key part of the evolution toward an infrastructure that minimizes effort and costs when it comes to ever-changing business processes.

For more information on SAP NetWeaver, ESA, or Web service functionality, visit **www.sap.com/solutions/netweaver**. ■

Martin Huvar joined SAP about eleven years ago. After some years of development and consulting in different technology areas, he worked as Product Manager for the SAP Business Connector and XML Technology. He is now the responsible Product Manager for Enterprise Services and Web Dynpro at SAP. Martin can be reached at martin.huvar@sap.com.

Sven Leukert joined SAP in 1998, after studying mathematics in Berlin and receiving a Ph.D. in mathematics from the University of North Carolina, Chapel Hill. He worked in product management for CRM Middleware until 2000, and has been involved in the development of the SAP Exchange Infrastructure as a Product Manager from its beginnings. He is currently a Product Manager for SAP NetWeaver, focusing on cross-component aspects and processes.