



SAP INTEGRATION APPROACHES

Best Practices for SAP application integration projects

Abstract: One of the most pervasive challenges for SAP shops is integrating SAP to other applications within their organization. Without proper integration between SAP and these systems, organizations fail to fully automate and optimize their business processes. Learn some of the most common challenges faced when integrating SAP, a brief overview of SAP interfaces and integration points, and various approaches for SAP integration.

www.mulesoft.com | info@mulesoft.com | 1-877-MULE-OSS

1 SAP Integration Overview

As one of the most popular integrated enterprise resource planning (ERP) solutions on the market, SAP has enabled organizations to optimize many of their most essential business functions in finance, human capital management, enterprise performance management, and supplier relationship management. One of the most pervasive challenges for SAP shops is integrating SAP to other applications within their organization. Without proper integration between SAP and these systems, organizations fail to fully automate and optimize their business processes.

This whitepaper provides an overview of some of the most common challenges faced when integrating SAP, a brief overview of SAP interfaces and integration points, various approaches for SAP integration and a suggested best architecture for solving the SAP integration challenge.

2 Common SAP Integration Challenges

SAP has numerous touch points to other applications within the enterprise. It would be difficult to enumerate all possible integration use cases involving SAP. Therefore, we will focus on three of the common use cases: CRM integration, supplier integration and integration with third party purchase order systems.

Perhaps one of the most common use cases is integration with CRM. Frequently organizations need to synchronize customer data between their CRM and ERP systems synchronously. If a new customer comes on board it may be necessary to get information for financials, performance management or other SAP-managed business functions. Failure to do so, or failure to do so in a timely manner as is common with some data sync approaches can reduce business agility. As Salesforce.com, a software as a services application, has become widely adopted the problem has been further compounded by the technological differences between many common on-premise and cloud-centric integration approaches.

A second common scenario is integration with suppliers. When SAP creates a purchase requisition indicating a firm needs to purchase a component or raw material it is necessary for a request to be sent to all possible suppliers of the good. Once suppliers respond with quotes these must be routed back to SAP for further processing. An integration layer is therefore needed between SAP and the heterogeneous ways of communicating with suppliers.

Finally, many SAP customers utilize purchase order systems that must be integrated with SAP. To ensure the successful and timely functioning of the business processes, purchase orders must be promptly transmitted to SAP.

These are just a few of the multitude of use cases. Even more challenging to organizations is designing for the use cases they have not yet imagined. Organizations need to build a software architecture that facilitates quickly building new applications, including those that integrate with SAP.

3 SAP Interface and Integration Technologies

While integrating with SAP can be highly challenging, SAP does provide a number of key building blocks that facilitate this process for application developers.

The first of these are Intermediate Documents (IDocs). IDocs is a standard data format defined by SAP for the exchange of information between SAP and non-SAP applications. IDocs are typically used when information needs to be sent to or from SAP without notification requirements. IDocs are primarily used to transfer master data in and out of SAP. For example, using IDocs you can retrieve suppliers, cost centers, activity types, logistics information such as bill of materials and much more. For many SAP integration scenarios IDocs would be retrieved or sent to access SAP information.

A second core SAP technology are SAP Business Application Programming Interfaces (BAPI). BAPIs are defined interfaces that can be called either by SAP applications or non-SAP applications. BAPIs are typically used in synchronous scenarios where two-way communication is required. Many operations can be completed using a BAPI, with or without retrieval or IDocs. For example, if an organization needed to manipulate their costs centers from an external application, they could get a list of profit or cost centers or create new ones using a BAPI. Likewise, if they wanted to plan new orders or change existing orders, this could be done via a BAPI. There are literally hundred of BAPIs available for SAP that provide a broad set of functions for SAP integration.

Finally, SAP offers a middleware component called the SAP Java Connector (JCo). JCo enables development of SAP-compatible components and applications in Java. For Java developers who must integrate with SAP it is a necessary component. For this reason it is now extremely pervasive in SAP environments.

4 Approaches to SAP Integration

There are several approaches to integrating with SAP, each with specific strengths and weaknesses.

Point-to-Point direct integration with SAP interfaces: The key interfaces of IDocs, BAPIs and JCo can, in theory, be directly accessed by software applications. In an effort for expediency many developers looking for quick solutions to integrate with SAP directly wire their applications to SAP. The primary advantage of doing so is the initial time to write a single application against SAP.

While point-to-point integration can have a low initial hurdle in some cases, in the long term it is generally the most expensive and most cumbersome way to integrate with SAP. In these types of architectures, applications become tightly coupled with SAP. This makes them very brittle as changes to the SAP environment changes over time. Furthermore, as business processes change and new integration scenarios are required, each becomes progressively more complex as it involves additional

touch points and tight dependencies. The end result is a brittle environment that fails to deliver business value in the long term. For this reason these sorts of integrations are seldom more than expedient “hacks” and are generally not recommended.

Service Oriented Architecture (SOA) stacks: The next alternatives for SAP integration are SOA stacks. Large vendors including IBM, Oracle and others offer these integration solutions. SOA stacks are comprised of multiple products including application servers, enterprise services busses, orchestration engines, management tools, and development tools. Very often these include half a dozen or more products.

The primary advantage of utilizing a SOA stack is that it creates a relatively robust integration architecture that can address most use cases. Applications are loosely coupled so when changes are needed they can be quickly addressed. Application maintenance costs are lower than in point-to-point approaches and the overall platform is more reliable. Furthermore, because these stacks can service-enable SAP, once implemented, the cost and complexity of further applications to support new business processes is considerably less than in point-to-point or hub-and-spoke approaches.

However, SOA stacks come with as many flaws as benefits. SOA stacks involve multiple products which all must be deployed and configured. A full SOA implementation utilizing one of these stacks can take multiple years and extraordinary up front costs. In the interim, new application development can grind to a halt. Finally, all developers must be trained on the proprietary tools of these vendors. Recruiting new developers is also often quite difficult going forward as specialized skills are now needed above and beyond SAP knowledge. Needless to say, many organizations struggle with implementing, maintaining and staffing a massive IBM or Oracle solution to integrate with their already sizable SAP implementation.

Lightweight standalone Enterprise Service Bus (ESB): Standalone ESBs are the newest and most modern approach to integrating with SAP. ESBs are one of the fundamental components of a SOA stack. However, unlike the legacy stacks, standalone ESBs can operate without outside app servers or other infrastructure components. Furthermore, they typically provide their own management tools or integrate with any management tool the organization may already use. Finally, they generally utilize industry-standard technologies and development tools that developers already are familiar with.

In almost all cases a standalone ESB is the optimal solution for SAP integration. These products allow organizations to service-enable SAP without a massive upfront investment, developer-training program or multi-year rollout. They allow organizations to meet their most critical business objectives in a short timeframe while simultaneously laying the foundation for future growth. While offering all the advantages of a SOA stack, standalone ESBs reduce the upfront cost and risk associated with building a solid foundation for all variety of SAP integration scenarios.

5 Mule ESB: The best way to integrate with SAP

There is only one lightweight, standalone ESB certified by SAP for SAP integration, Mule ESB. For almost all SAP integration scenarios, Mule's SAP Enterprise Connector can provide an interface to SAP. The connector can:

- Execute any standard and custom BAPI in SAP
- Receive calls from ABAP code exposing external code running in Mule as a BAPI inside SAP
- Send and Receive IDocs
- For easier usage and integration, all SAP objects can be treated as XML documents

Mule ESB has a number of advantages over other solutions for particular integration scenarios. For example, in the very popular use case of integrating between SAP and Salesforce.com, Mule not only provides the only SAP-certified connector for a lightweight, standalone ESB, it also provides industry-leading Cloud Connect technology which simplifies integration with Salesforce's numerous APIs. Delivery of these types of integrations is easier than ever before.

Mule ESB is easy for developers to learn and understand, meaning any Java developer familiar with SAP can be quickly productive with Mule. This reduces training, staffing and other development costs and increases the speed at which the organization can deliver applications that capitalize on changing business needs.

Compared to other alternatives, Mule enables organizations to quickly deliver SAP-integrated applications to address immediate business opportunities while laying the foundation for future projects. Mule eliminates the huge projects associated with large-scale SOA implementations and delivers superior return on investment compared to alternative approaches.

It's little wonder that Mule ESB is the world's most widely used integration platform and that SAP Ventures has actually taken an equity stake in MuleSoft, the company behind the Mule ESB. It truly is the best choice for SAP integration.

SAP® Certified
Integration with SAP Applications

SAP | **PartnerEdge**

About MuleSoft

MuleSoft provides the most widely used integration platform for connecting SaaS and enterprise applications in the cloud and on-premise. With the rise of cloud and mobile, enterprises face a choice: become overwhelmed by the resulting explosion of end points or seize the opportunity to gain competitive advantage. Founded on the idea that connecting applications should not be hard, MuleSoft lets organizations harness the power of their applications through integration. MuleSoft's Anypoint™ technology eliminates costly, time-intensive point-to-point integration, enabling business agility. Delivered as a packaged integration experience, Mule CloudHub™ and Mule ESB™ are built on proven open source technology for the fastest, most reliable integration without vendor lock-in. Supporting billions of transactions per day, MuleSoft is used in production by thousands of enterprises, including Walmart, MasterCard, Nokia, Nestlé and Honeywell, and powers integrations with leading SaaS vendors such as Salesforce.com, NetSuite, Workday, Intuit and Zuora.

For more information: www.mulesoft.com, or email info@mulesoft.com.

Download Mule ESB: <http://www.mulesoft.com/download/>

MuleSoft and the MuleSoft logo are trademarks of MuleSoft Inc. in the United States and/or other countries. All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

All contents Copyright © 2011, MuleSoft Inc.