Cloud Integration Cheat Sheet

Boomi

What is cloud integration?

Cloud integration simply means integrating cloud applications with on-premise applications or other applications in the cloud. Before Dell Boomi AtomSphere, customers needed to install and maintain integration software or hardware appliances on-premise. AtomSphere enables any combination of SaaS and On Premise integration directly from the web without requiring software packages or hardware appliances.

How is cloud integration different from traditional middle-ware products?

Traditional middleware products were built before cloud was prevalent. They typically are traditional integration server products that have connectors or adaptors built to connect applications together. A number of them have added or will add new connectors for SaaS applications such as salesforce.com.

However, key concerns from IT — as they focus on their cloud strategy — are not addressed in these solutions, such as (1) automatic upgrades, (2) governance of data as it pertains to integration and movement of data between various clouds and on-premise applications, and (3) centralized development and operations to ensure external compliance, and scalably manage integration infrastructure for an ever-growing number of clouds in use by an organization.

Is cloud integration compatible with my existing middle-ware solution?

Yes, cloud integration solutions, such as AtomSphere, contain Java Message Service (JMS) connectors to plug into existing middleware investments such as MQ Series, Progress Software, TIBCO and webMethods. This cloud gateway gives users the ability to audit and govern the movement of data, enforce security of data prior to leaving the firewall, and access connectivity to most major SaaS applications.

What is the difference between a single-tenant approach and a multi-tenant approach to cloud integration?

Single-tenant integration really just moves the traditional IT problems companies face today into a new data center. By architecting a multi-tenant integration solution, users naturally benefit from the collective intelligence of the community (a recent example being Boomi Suggest) because everyone is using one highly scalable and secure instance of the platform. Users also benefit from rapid innovation and product improvement only possible by a true SaaS offering. Users, for example, can centrally develop, deploy, and manage integrations across all of their applications from one central platform.

What are the common mistakes IT makes when undertaking a cloud integration project?

It is key that the integration project is thought of as a business process enabler and explained that way to the project stakeholders. By designing the integration in this context, the true value of the integration will be apparent, and the complexity of the project will be apparent and rationalized from the beginning.

Is custom coding a good option for cloud integration?

Be sure to consider the full range of what is required to be done if custom coding the integration is being considered. Custom coding involves writing code to integrate applications vs. using a cloud integration platform. This is often a tempting path, but the maintenance cost of this project will always outweigh the initial build of the integration. Cloud integration platforms, due to their multi-tenancy, are optimal approaches to ensure the integrations remain operational, even as the apps being connected are upgraded or changed by the SaaS ISVs themselves. Custom-coded solutions must also contemplate governance capabilities, as it is critical to have traceability of all data as it leaves the enterprise.

Tips for Integration Process Development and Testing

Start simple and iterate. Create a very simple process that moves data from source to destination to make sure the connections, filters, and basic mapping are correct. Then begin introducing validations, routing, advanced mapping/functions, custom error handling, logging, etc.

Start with the end points. When building a new process from scratch, begin by configuring the end points: the source connection/operation in the Start step and the destination connection/operation in a Connector step. The operation configuration (for most connectors) will automatically generate the Profile components that you will use in the mapping, decisions, or other steps.

Limit the amount of records. Don't select all 100,000 records from your database in the first test. During initial testing, mock up small test files or use filters to restrict the number of records being processed (e.g. select specific records by ID or name). As you get more comfortable with the process, then introduce larger data sets.

For more information visit **www.boomi.com**.