Simon Dickerson
Raji Narayanan

IBM WebSphere Cast Iron Introduction and Technical Overview

IBM WebSphere Cast Iron Cloud Integration enables companies to integrate applications, regardless of whether the applications are located on-premise or in public or private clouds.

This IBM® Redpaper™ provides an overview of the core components and use of WebSphere Cast Iron Standard and Enterprise editions and includes a brief introduction to the Express edition. It covers the use of the following form factors:

- The virtual appliance, WebSphere Cast Iron Hypervisor Edition
- The physical appliance, WebSphere DataPower Cast Iron Appliance XH40
- The multi-tenant cloud service, WebSphere Cast Iron Live

Integrating cloud applications

Cloud computing has become a business evolution that is impacting all facets of business today, including sales, marketing, human capital management, supply chain, finance, manufacturing, quoting, ordering, service, support, email, and so on. As each of these functions move to the cloud they require migration and integration. Additionally, each vendor providing a cloud solution creates their own interface to the application. This fact creates a challenge in IT organizations in companies of all sizes and all locations globally as they attempt to understand and then manage these unique application interfaces and integrate applications from cloud to cloud and cloud to on premise. It is for these reasons that IBM Cast Iron® is an invaluable solution to a modern IT infrastructure

Cloud computing can provide companies with the following capabilities to meet business needs:

- Flexibility in working practices, such as working from home, the office, or with mobile computing, which creates the need to access data and files from anywhere at any time
- Easy implementation of applications that are offered as a software as a service (SaaS) and that can be configured and running in a matter of days
- Low cost solutions that can be paid for incrementally
- A reduction in the IT resource that is required for the maintenance of servers and infrastructure
- A reduction of impact from software upgrades and maintenance

However, the adoption of SaaS applications can mean that additional silos of information are created, which can lead to further disparate sources of information. These disparate sources create a challenge in that there is a hybrid architecture that includes on-premise solutions and the new SaaS applications. Integrating these applications and data can be key to the efficiency of the business. Integration can ensure that business users have access to the information that they require with the least possible friction, thus maximizing productivity and improving customer interaction.

Integration can be beneficial in the following situations:

- Migrating data to new SaaS applications
- Providing a single view of customer information in cloud and on-premise applications, for example a cloud-based CRM system and an on-premise Enterprise Resource Planning (ERP) system
- Using information from cloud marketing automation in e-commerce solutions
- Providing connectivity between private cloud systems and business partner solutions

An integration solution must bridge the gap between the on-premise existing systems and new cloud applications, platform, and infrastructure, providing a rapid and easy-to-use method of setting up the integrations.

WebSphere Cast Iron provides a solution that meets the challenge of integrating cloud applications with on-premise systems, cloud applications-to-cloud applications, and on-premise to on-premise applications, as illustrated in Figure 1 on page 3. The WebSphere Cast Iron environment focuses on the business requirements, the applications, and the business user requirements. It removes the daunting complexity of integration.

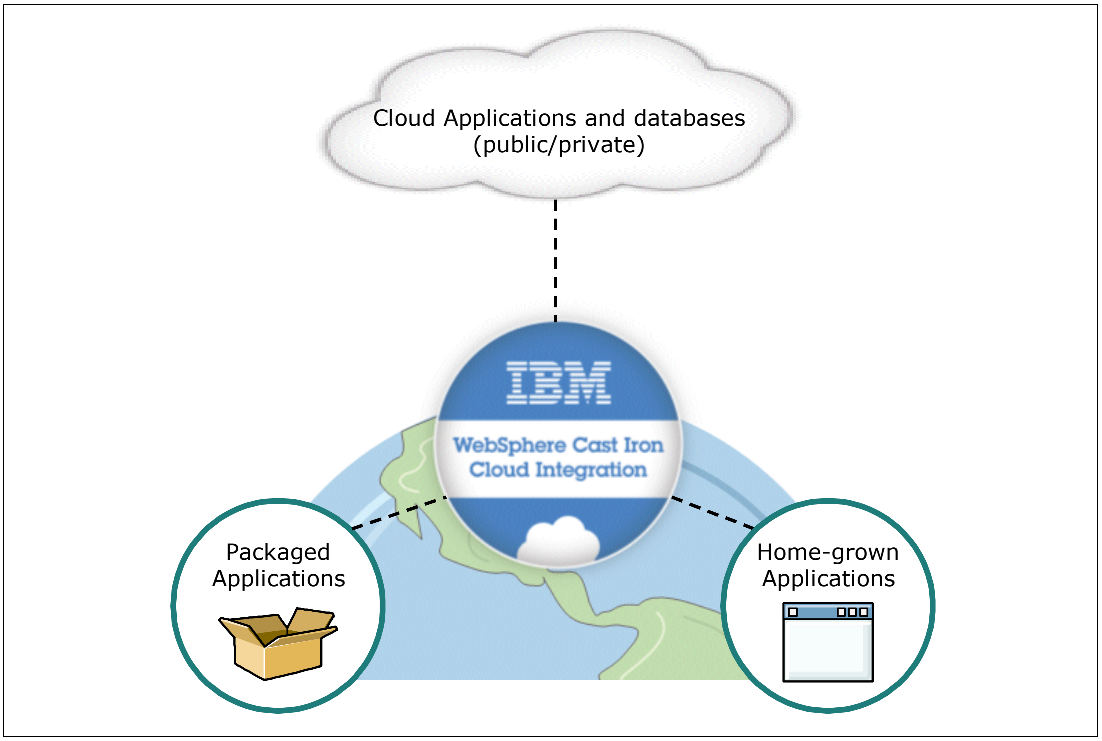


Figure 1 Cast Iron integration

Example use cases

WebSphere Cast Iron was beneficial in the following typical use cases:

- Exposing ERP information to sales agents

  A company needed to make sales order information that was stored in multiple ERP systems available to sales agents who used a SaaS-based CRM system. Integrating the ERP information with the CRM system provides the agents visibility to the necessary information and avoids the agents having to sign on to multiple systems. The company had only a small IT department, with no substantial development capability, so they need a tool that business analysts can use.

  The initial project involved using WebSphere Cast Iron to create three orchestrations that took less than two weeks to develop. The orchestrations provided real-time integration of the ERP information into the CRM system. Over time, the integration capabilities were expanded. There are now over 150 orchestrations connecting a multitude of systems that are running both in batch and on request.

- Integrating separate insurance policy systems

  An insurance company has multiple systems that are used to manage new business. Having one system that provides quotes for policies and another system that is used to administer policies occasionally resulted in duplicate data entry and as a result, generated error conditions. In addition, many of the workflows involved in bringing in new business are manual, and vendors must provide documents in multiple formats.

  The company used WebSphere Cast Iron to provide bidirectional integration between the two systems. Policy quote information is integrated into the administrative system and vice versa, making it necessary to enter data only one time and ensuring the integrity of the data in both systems. Integrating data through key business systems ensured that files were routed correctly and that workflow was not interrupted if a user is unavailable. Vendors must provide data only in one format with the integration solution making the data available to the systems where it is needed in the proper format. Finally, the integration solution eliminates errors that were introduced because of duplicate data entry.

- Synchronizing e-commerce systems and linking to the SaaS CRM application to provide a single view of customers

  A commercial enterprise has two e-commerce systems that are populated with customer and order data that is currently not synchronized. The enterprise also has a CRM application where customer data is kept and a fulfilment system with inventory and product data. The enterprise used WebSphere Cast Iron to integrate the data in these systems to provide a more cohesive view of customer data.

  The initial implementation included the following orchestrations that provided the initial population of data throughout the systems and the synchronization of information between systems:

  - Data providing a single view of the customer (SVC) was populated into the SaaS CRM system from the enterprise’s existing systems using a web services integration.
  - An orchestration with a short polling interval pushes account, order, inventory, and product information to the SVC from the e-commerce and fulfilment systems, ensuring that the customer service staff has the most current information.
  - A nightly refresh of all stock keeping units (SKUs) in the e-commerce system occurs based on information from the fulfillment system. The orchestration uses secure file transfer to perform the refresh.
  - Synchronization of customer records between the two e-commerce systems occurs on a regular basis, ensuring that when customers sign in to a system their data is available to them.

The WebSphere Cast Iron approach to integrating applications

WebSphere Cast Iron provides an approach to integrating applications that does not require any programming knowledge. You can build integration flows in WebSphere Cast Iron Studio (referred to in this paper as *Studio*), which is a graphical development environment that is installed to a personal computer (PC). With Studio, you create an integration *project* that contains one or more *orchestrations*. Each orchestration is built with a number of *activities* that define the flow of data. You can define the details of an activity from the configuration panes within Studio.

A project contains all of the assets that are required for the orchestrations to run, including any file schemas, WSDL files, and functions. The project also defines connectivity to the sources of data, the *endpoints*. Cast Iron contains many built-in connectors to applications (for example SAP), databases, and web services that make connecting to these endpoints straightforward.

Figure 2 outlines the Cast Iron approach. Within Studio, there is a simulated runtime environment to enable unit testing of the orchestrations with visibility of all data at all points. After you test the project, you can publish the project to the run time, where you can then manage and monitor the project. A browser-based interface, the *Web Management Console (WMC)*, provides the capability to manage all functions on the run time.

Cast Iron Live: Cast Iron Live provides the same Studio development environment and runtime functionality as a physical or virtual Integration Appliance but through a cloud-based service. In Cast Iron Live, you publish a project to an environment, for example development, test, and production. The environment contains the functions of an on-premise Integration Appliance.

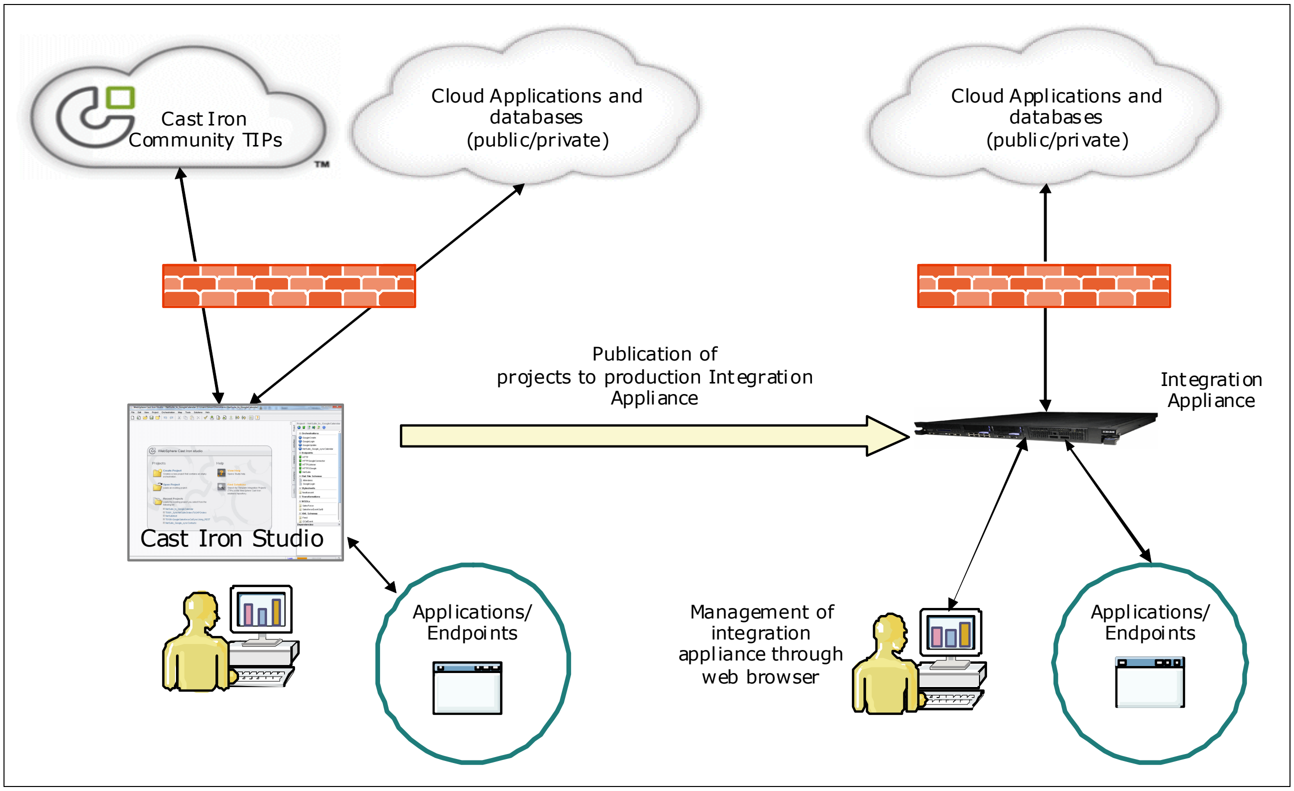


Figure 2 Cast Iron approach to developing integrations

As with any tool, there are preferred practices to implement designs. When processing large amounts of data, a poor design can mean that the integration takes an extraordinary amount of time or that it simply does not work. Each endpoint is different in its requirements, and any integration flow must use the characteristics of the endpoint to its best advantage, for example processing batches of data to optimize the speed of transactions.

Cast Iron also provides *Template Integration Projects (TIPs)* that encapsulate a specific integration use case between specific endpoints and that include preferred practices. You can download these TIPs from the Cast Iron community and modify to fit your precise needs.

Cast Iron Express uses a unique approach to the design and build of the integration workflows and is introduced in "Express edition" on page 23.

Architecture overview

Cast Iron provides the following models of implementation:

- An appliance model where the appliance can be either physical hardware or a virtual machine and where the Integration Appliance is installed on-premise, behind the firewall
- A SaaS model (Cast Iron Live)

Each of the three form factors (physical, virtual, or Cast Iron Live) allows for cloud-to-cloud, cloud to on premise and on premise to on premise integration. Each form factor allows for real time, near real time, and batch integrations.

This section introduces the components of a Cast Iron implementation, differentiating between the on-premise Integration Appliance and the cloud model (Cast Iron Live).

On-premise Integration Appliance

An on-premise appliance normally resides behind the firewall and not within a DMZ. For the development, test, and production life cycles, the runtime environments are typically separated and each has its own Integration Appliance that accesses the endpoints that are necessary for that environment. Figure 2 on page 5 outlines the approach for using an on-premise Integration Appliance.

You have full control of projects and their orchestrations on the appliance through the WMC. The orchestrations are started through a mechanism that is defined by an activity in the orchestration. These activities include scheduling, polling, or waiting for an incoming request, such as an HTTP Receive Request. Each orchestration must include at least one of these *starter activities.*

The Integration Appliance requires connectivity to the endpoints that are required by the orchestration. Data flows through the Integration Appliance and is stored internally as XML variables. The Integration Appliance allows you to control the logging for each orchestration.

Cast Iron Live

The multi-tenant Cast Iron Live service includes the following key components that allow you to design, run, and manage integrations, all in the cloud:

- A clustered runtime engine that runs the integrations and that has built-in fault-tolerance and recovery mechanisms
- A multi-tenant highly-available system to store the designed integrations
- A load-balancer to intelligently manage the loads throughout the various runtime engines
- Highly-available file-systems to store and manage logs that are related to the integrations

The design environment, referred to in Cast Iron Live as *Designer*, provides the same design capabilities and user interface as Studio does for the on-premise Integration Appliance.

You can access the Cast Iron Live interface from a web browser, as shown in Figure 3.

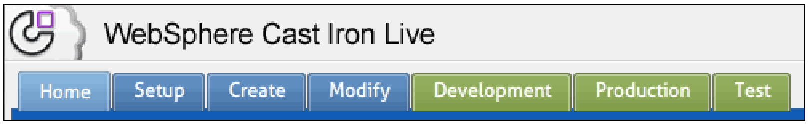
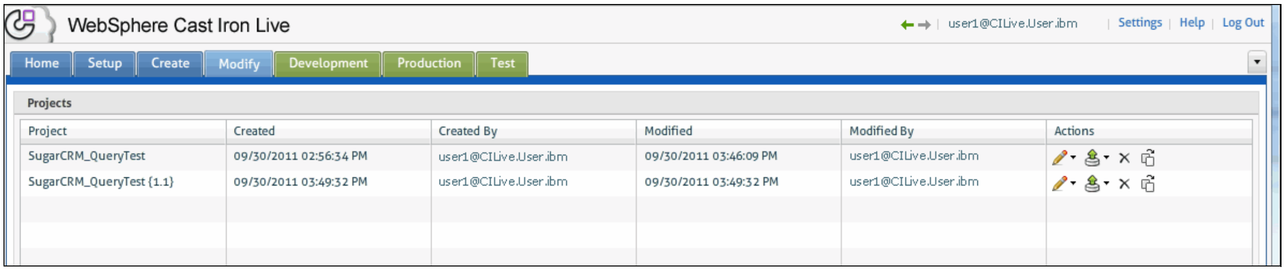


Figure 3 Cast Iron Live tabs for life cycle environments

You access the runtime environments through the Development, Production, and Test tabs. You can create projects in the Create tab and later change projects using the Modify tab.

The Modify tab, Figure 4, shows all projects that you are allowed to view.



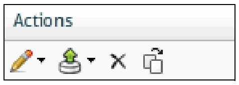
Projects

Project	Created	Created By	Modified	Modified By	Actions
SugarCRM_QueryTest	09/30/2011 02:56:34 PM	user1@CILive.User.ibm	09/30/2011 03:46:09 PM	user1@CILive.User.ibm	✎▾ ⇧▾ ✕ ⧉
SugarCRM_QueryTest (1.1)	09/30/2011 03:49:32 PM	user1@CILive.User.ibm	09/30/2011 03:49:32 PM	user1@CILive.User.ibm	✎▾ ⇧▾ ✕ ⧉

Figure 4 Cast Iron Live Modify tab showing a list of projects

Each project has the following actions, as shown in Figure 5, which are initiated by clicking an icon:

- Edit Project (✎▾)
- Publish (⇧▾)
- Delete (✕)
- Create New Version(⧉)



Actions

✎▾ ⇧▾ ✕ ⧉

Figure 5 Cast Iron Live project Actions icons

To edit a project, select **Edit Project**, and then select **Edit Project in Designer**. The Studio development environment opens with the same functionality that the on-premise installation has. You can then publish the project to one of the runtime environments by selecting **Publish**.

When using Studio in Cast Iron Live, development is the same as for an installation on the local PC. Studio in Cast Iron Live has access to the same endpoints that the PC has access to. When publishing to a runtime environment, the orchestration is run in the Cast Iron Live cloud.

Figure 6 on page 8 illustrates the Cast Iron Live process. A developer accesses Cast Iron Live through a web browser, creates projects, and develops these projects with Studio (which runs in a Java runtime environment on the PC). When complete, the developer publishes the project to a test environment. After testing, the projects are published to a production environment. You can also manage the environment, the projects, logs, and other components through a web browser. If you connect to applications or endpoints behind a firewall, the Secure Connector is required, as described in the following section.

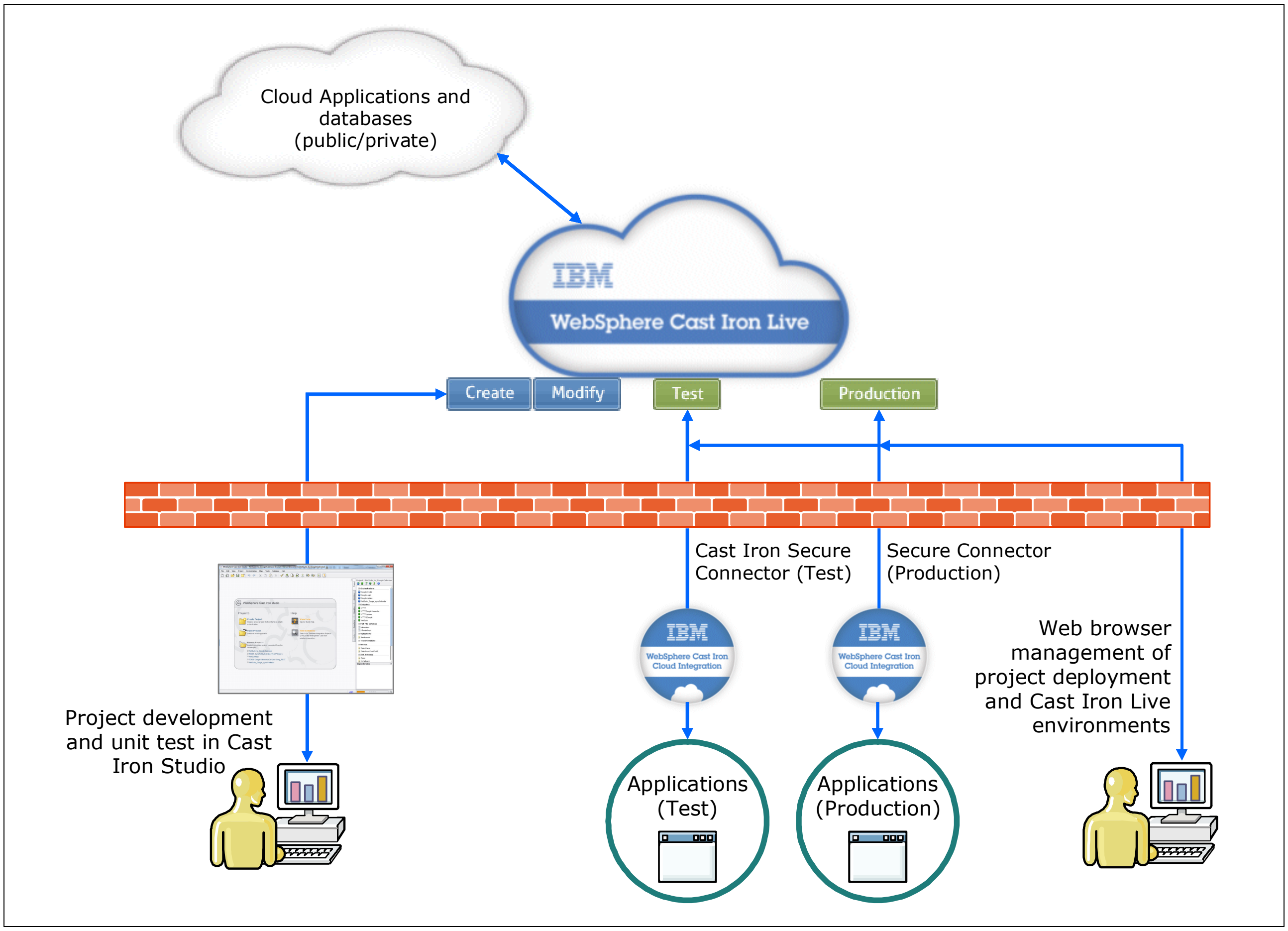

Cloud Applications and databases (public/private)
IBM
WebSphere Cast Iron Live
Create
Modify
Test
Production
Cast Iron Secure Connector (Test)
Secure Connector (Production)
IBM WebSphere Cast Iron Cloud Integration
IBM WebSphere Cast Iron Cloud Integration
Web browser management of project deployment and Cast Iron Live environments
Project development and unit test in Cast Iron Studio
Applications (Test)
Applications (Production)

Figure 6   Cast Iron Live architecture

Secure Connector

The Cast Iron Live Secure Connector allows orchestrations in Cast Iron Live to connect to endpoints behind a firewall. A separate Secure Connector is configured for each environment and is installed to a computer that is running the Windows or Linux operating systems behind the firewall.

The Secure Connector includes the following security features:

- The Secure Connector always initiates communication with Cast Iron Live and that communication is validated before the Secure Connector attempts further processing.
- Communication between the Secure Connector and Cast Iron Live is based on the standard SSL 128-bit encryption over HTTPS through port 443. When the Secure Connector starts, it undergoes the SSL/TLS handshake, authenticates through standard X.509 certificates, and establishes a TLS-encrypted tunnel if all connections are successful.
- When the TLS connection is established successfully, the Secure Connector sends a request to Cast Iron Live for authentication. Based on information that is provided by the Secure Connector, including a private key, Cast Iron Live ensures that only the correct Secure Connector is granted access to a particular environment of a tenant.

- Data transmission and requests that are inbound to the Secure Connector from Cast Iron Live are limited to the available set of endpoint connectors that are provided by Cast Iron. Users must explicitly specify address and authentication information from each endpoint to which they are connecting.
- Local scripts or executables cannot be run directly through the Secure Connector.
- The Cast Iron Secure Connector can have the network or IPs, that it can access internally, restricted.

Further security information is available in the IBM white paper, *WebSphere Cast Iron Live Security*, which you can download from:

http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=SA&subtype=WH&htmlfid=WSW14139USEN

Developing integrations

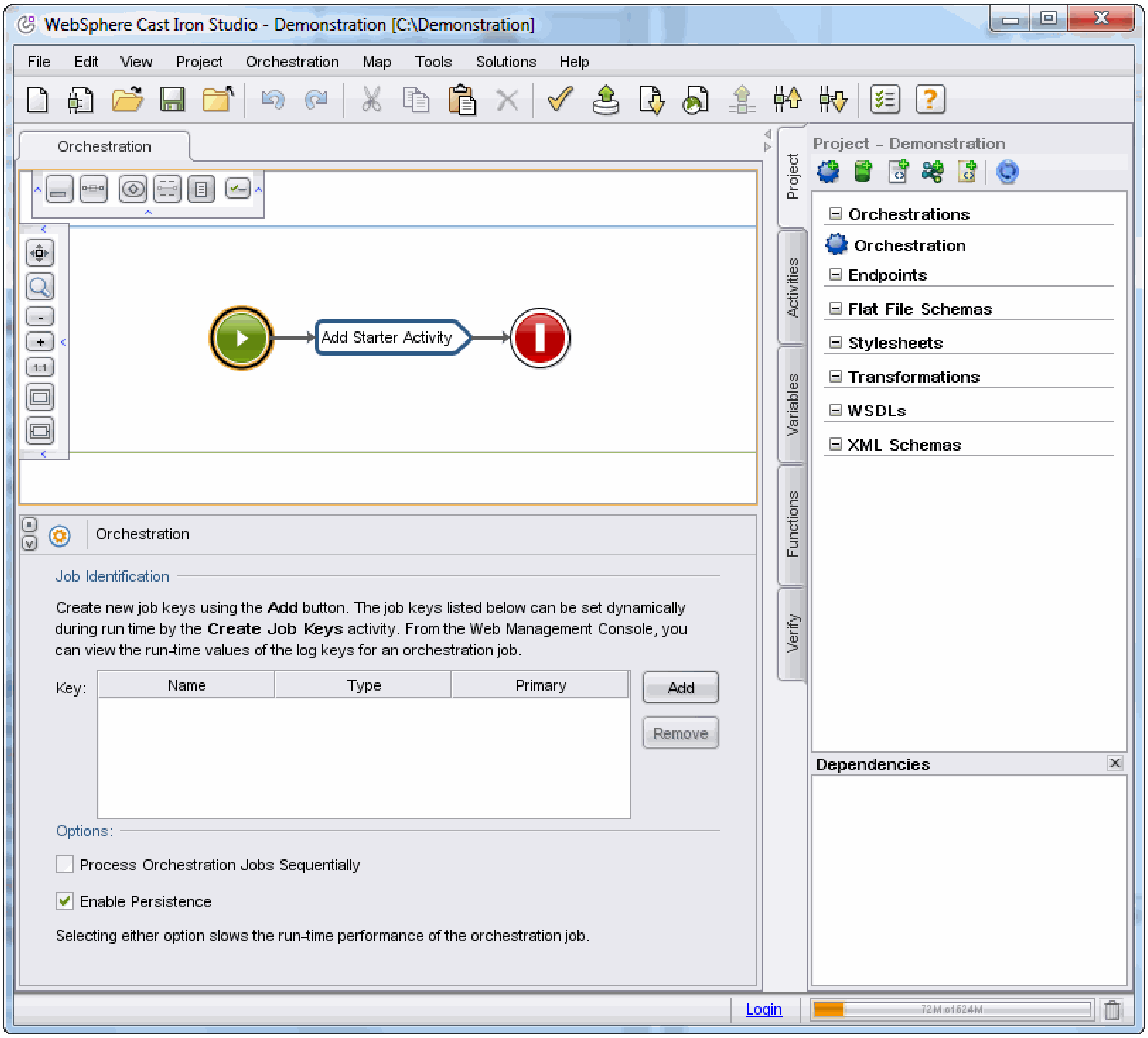
Integration workflows, or orchestrations, are contained within a project in Studio. When the project is complete, you publish it to the runtime environment.

Studio includes the following capabilities:

- The ability to design integrations from scratch using a drag-and-drop design palette
- A Template Integration Projects repository that allows you to reuse existing orchestrations
- Built-in connectivity to many cloud and on-premise applications
- Built-in transformation capabilities to improve data quality before the data reaches the target system
- Built-in functions to transform data from source format to a the format that is required by the target and to build relevant mapping rules
- The ability to build sophisticated workflows that include error-handling routines and re-try logic
- The ability to process messages in batch or real time
- Tools to unit test and verify whole or partial integration workflows before deploying them to the Integration Appliance
- One-click publishing of the completed integration workflows to the Integration Appliance
- A Connector Development Kit (CDK) to build connectors to custom applications that expose standards-based interfaces

The following Studio main components, illustrated in Figure 7 on page 10, enable the rapid development of integration projects:

- The main workspace where the activities are placed to build the orchestration (the top left pane in the figure)
- The configuration pane that contains the checklist and items that are configured for each activity (the bottom left pane in the figure)
- A tabulated section for assets of the project (the column to the right of the figure)



Orchestration

Job Identification

Create new job keys using the **Add** button. The job keys listed below can be set dynamically during run time by the **Create Job Keys** activity. From the Web Management Console, you can view the run-time values of the log keys for an orchestration job.

Key:

Name	Type	Primary

Add

Remove

Options:

□ Process Orchestration Jobs Sequentially

☑ Enable Persistence

Selecting either option slows the run-time performance of the orchestration job.

Figure 7 Cast Iron Studio showing a blank canvas

An orchestration is built by dragging activities onto the main workspace. Selecting an activity on the workspace opens the configuration pane for that activity. The configuration consists of a checklist of items that must be configured.

Connectors

Cast Iron contains built-in connectors to many applications. However, providing a connector for every application in the market is not practical. Companies also often build their own internal applications to which they want to connect. Although you can rely on generic connectivity through web services or can access data directly from the database to connect to applications, you can also use the Connector Development Kit (CDK) to develop a specific connector to an application.

The CDK is a wizard-driven tool that enables the development of a connector to abstract, specific functionality of the application, for example inserting new records or updating existing records. The CDK provides all the necessary tools to build the connector and also to perform unit testing, integration testing, and packaging and deploying the connector.

When new connectors are released by IBM Cast Iron, you can download them from Studio.

Built-in connectors: Studio built-in connectors have a discovery mechanism that lists objects, data fields, and parameters for the endpoint activity. For example, with the SAP Invoke RFC activity, you can browse the list of RFCs. When you select an RFC, the Cast Iron connector automatically retrieves the list of input parameters and the data fields that are returned.

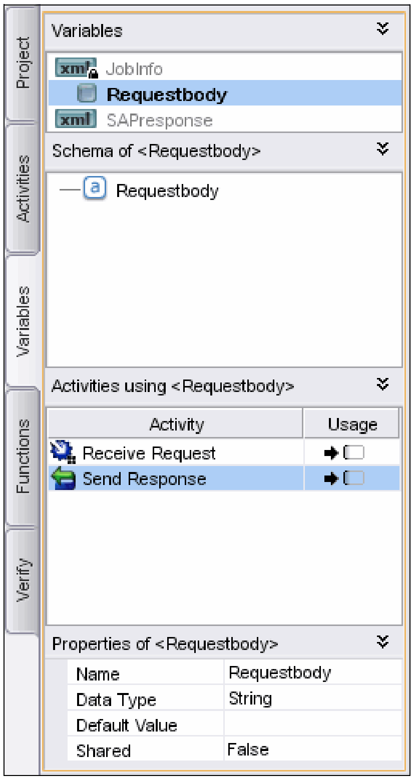
Secure communications

When communicating with endpoints, such as applications, databases, and flat-files, Cast Iron can communicate using a variety of secure communication protocols:

- HTTPS (HTTP over SSL): Supports bilateral authentication, privacy, and integrity
- Secure web services using SOAP/HTTP over SSL: Supports bilateral authentication, privacy, and integrity
- Secure FTP (FTP over SSH) and FTPS (FTP over SSL or Implicit FTPS): Supports secure mechanisms for FTP server authentication, privacy, and integrity
- Secure Databases (SSL): Supports secure mechanism for database access

Data variables

Variables contain the data within the orchestration and are defined by a schema or data type. Mapping data for an activity uses variables, and new variables can be created as required. The Studio Variables tab, Figure 8 on page 12, shows all of the variables, their properties within the project, and the schema and activities where the variables are used. From this tab, you can jump to the activity where the variable is used.



Variables

- JobInfo
- Requestbody
- SAPresponse

Schema of <Requestbody>

- Requestbody

Activities using <Requestbody>

Activity	Usage
Receive Request	➜□
Send Response	➜□

Properties of <Requestbody>

Name	Requestbody
Data Type	String
Default Value	
Shared	False

Figure 8   Studio assets section showing Variables tab

Activities and configurations

You build a Cast Iron orchestration from a number of activities, which are listed by category in the Studio Activities tab, shown in Figure 9. Activities exist for all connectors, and there are activities for handling workflow logic, for example error handling with Try/Catch blocks and transformations. Some activities are defined as starter activities, as described in , “Defining a starter activity for an orchestration job” on page 16.

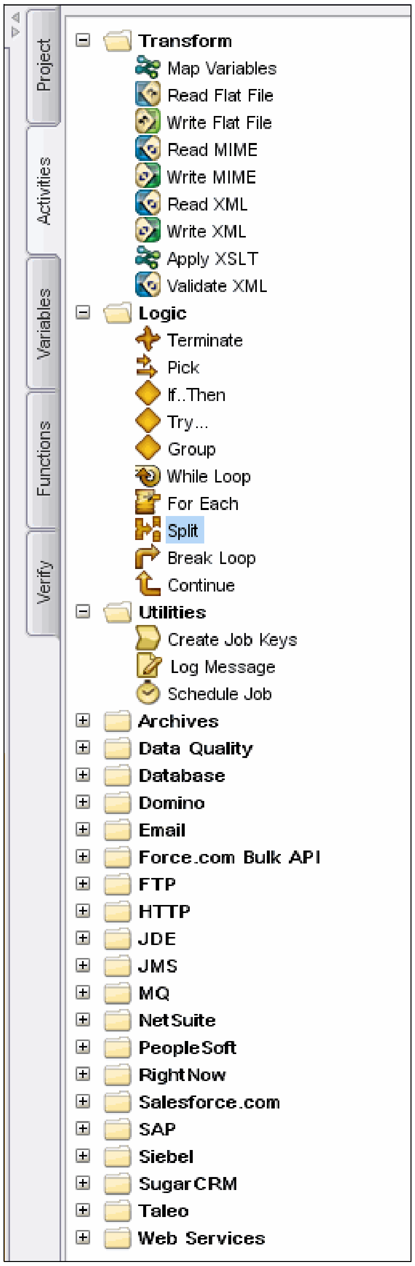


Figure 9 Studio Activities tab

You use an activity by dragging it to the workspace. After you place an activity on the workspace, you can move, delete, or reconfigure it at any time.

Figure 10 shows an orchestration in the workspace with the following activities:

- A Receive Request, which is the starter activity that receives a request from an HTTP endpoint
- A SAP activity, the Invoke RFC, which is the orchestration that the chosen RFC uses to retrieve data from SAP
- A Salesforce.com connector Upsert Objects, which performs the Salesforce.com function to update or insert a record to the selected Salesforce.com object
- A Send Response, which sends a response back to the HTTP requestor

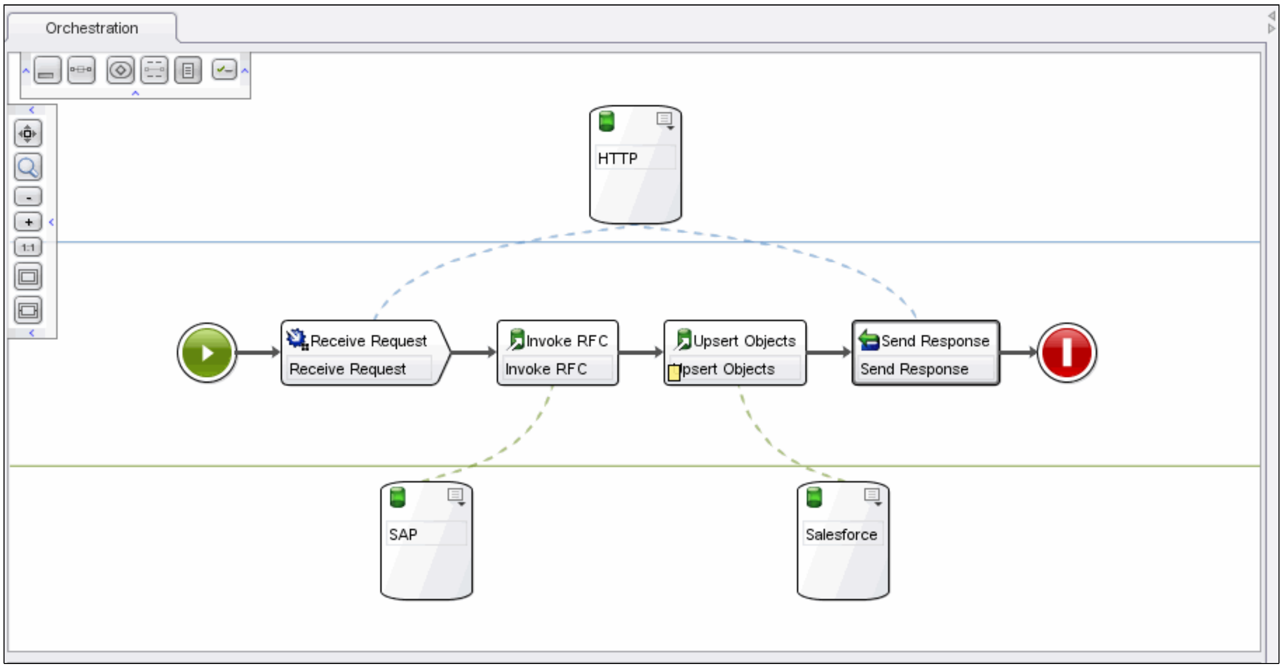
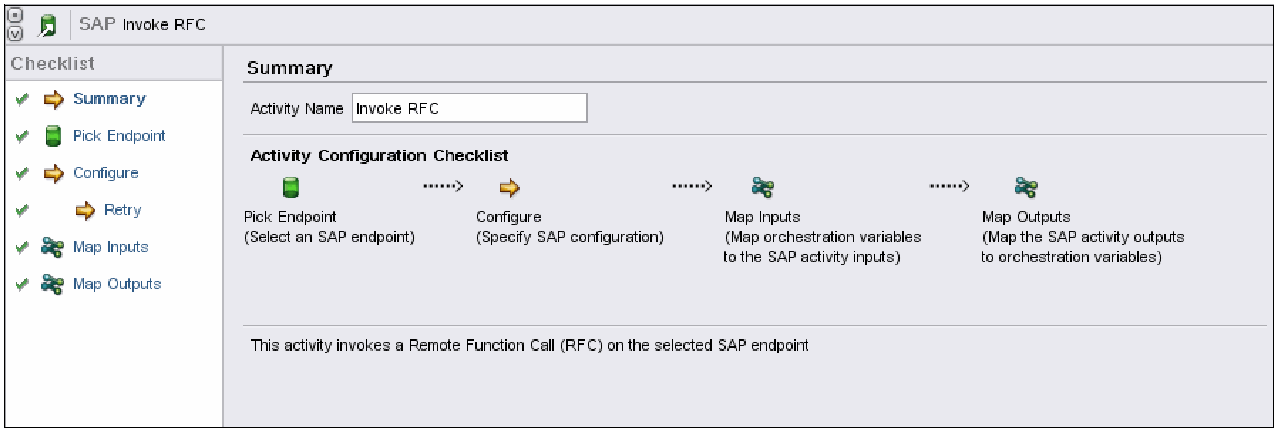


Figure 10 Studio canvas section showing initial development of an orchestration

When you select an activity in the workspace, Figure 11, the configuration pane is presented in the lower section of Studio. Activities for endpoints include a checklist of items and the input and output mapping requirements. Logic and transformation activities list items that are required for the configuration of that activity.



SAP Invoke RFC

Checklist

- ✓ Summary
- ✓ Pick Endpoint
- ✓ Configure
- ✓ Retry
- ✓ Map Inputs
- ✓ Map Outputs

Summary

Activity Name Invoke RFC

Activity Configuration Checklist

Pick Endpoint (Select an SAP endpoint) → Configure (Specify SAP configuration) → Map Inputs (Map orchestration variables to the SAP activity inputs) → Map Outputs (Map the SAP activity outputs to orchestration variables)

This activity invokes a Remote Function Call (RFC) on the selected SAP endpoint

Figure 11 Studio configuration pane for an activity

You cannot verify or publish a project if there are incomplete configurations, so Studio provides the ability to validate the orchestrations before publishing them.

Transforming and mapping data

The activities for an endpoint include a checklist of items for mapping input and output values, except for those activities that only have either inputs or outputs, for example HTTP Receive Request.

In a mapping, the *map inputs* define data that is sent to the activity. The *map outputs* determine the data that can be used by other activities in the orchestration. Mapping is easy to configure using the configuration panes.

Defining map inputs

Use the Map Inputs option in the checklist to specify data that is sent to the activity:

1. Click **Select Inputs** in the From Orchestration section to select the variables that are to be used. You can select more than one input variable.
2. Map the data fields to the activity input in the To Activity section by dragging the activity.
3. A function is applied to a data field by dropping the function to the center section and mapping the input data to the function and the function to the activity.

   Figure 12 shows an example of mapped inputs.

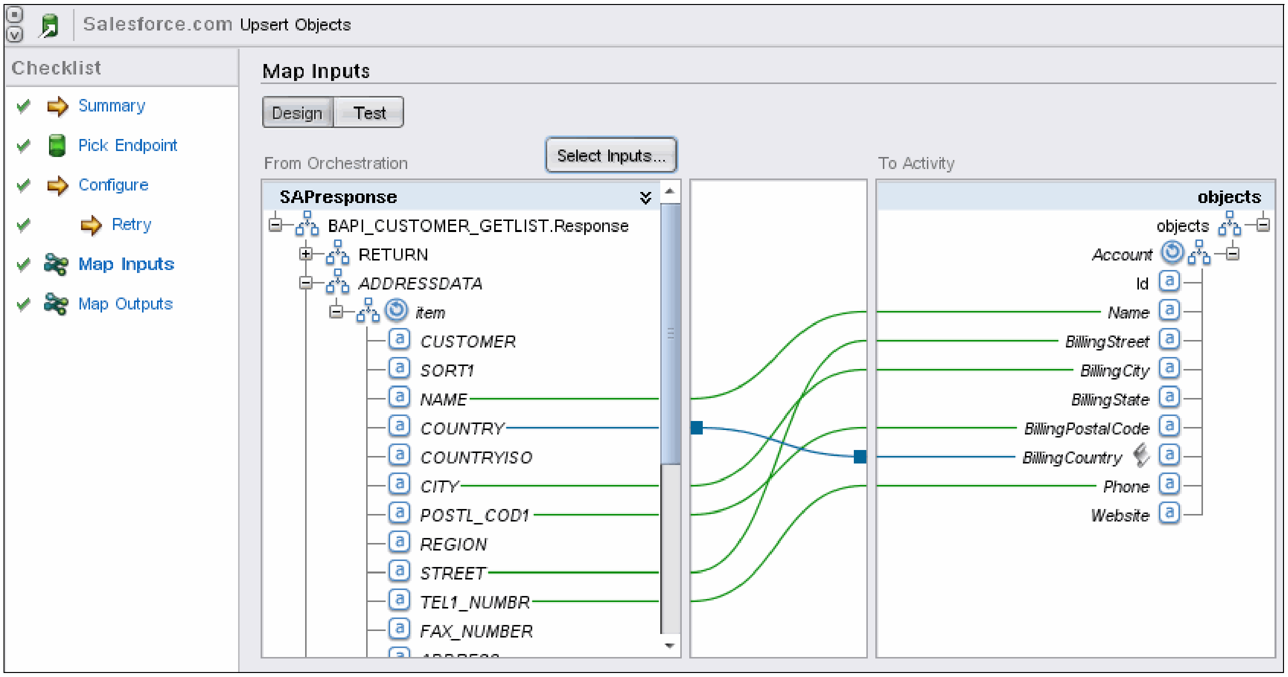


Figure 12   Studio example of mapped inputs

Map outputs

Use the Map Outputs option in the checklist to select the data that is required by other activities, for example, data that is required to be passed to an endpoint or return codes from an activity that need to be queried. One or more new variables are created for the data. Figure 13 shows an example of mapped outputs.

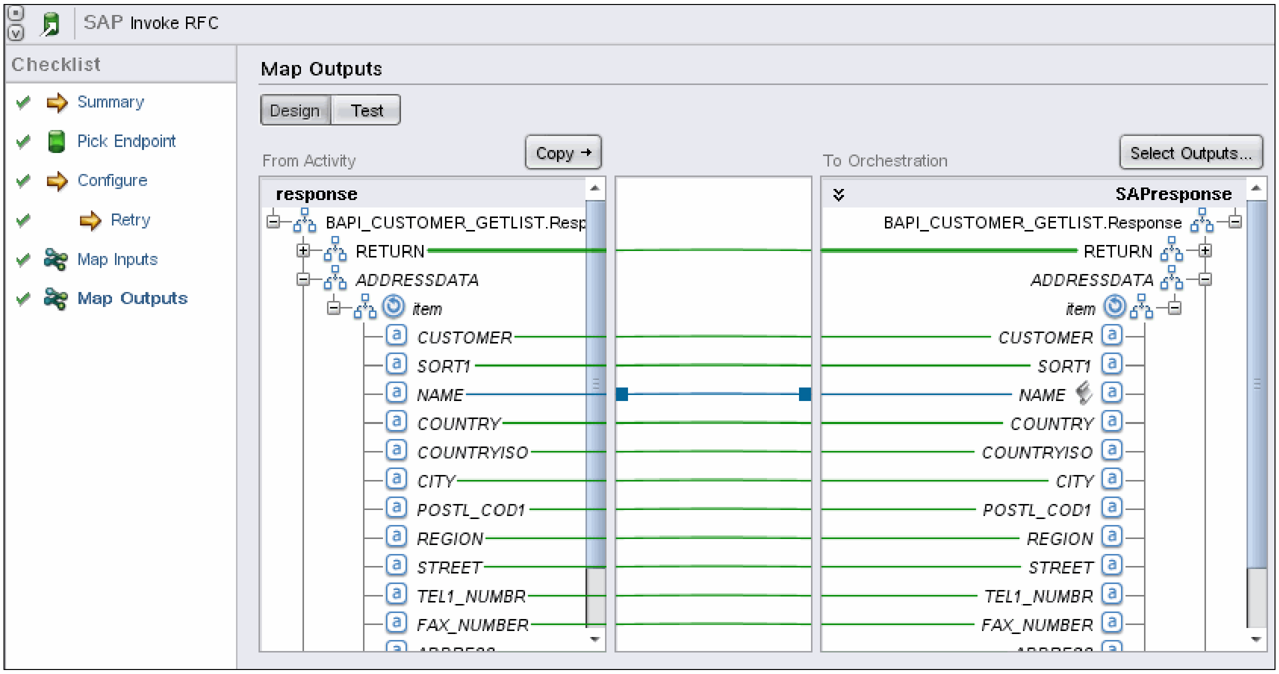


Figure 13 Studio example of mapped outputs

Defining a starter activity for an orchestration job

Orchestrations start according to the *starter activity* that is used. An orchestration must have at least one starter activity and can have more than one starter activity when a Pick activity is used. Starter activities are denoted in the Activities tab by a specific icon ().

When an orchestration starts, a new job is created on the appliance with a unique identifier. The WMC gives you full control over how an orchestration job is run and whether jobs can run in parallel.

Orchestrations can include the following starter activities:

- Schedule Job

  A new orchestration job is created at each scheduled time and the date-time value is available in Map Outputs.

- HTML Receive Request

  An HTTP request can be sent to the appliance to start the orchestration. A unique URL must be specified in the configuration for the activity, and the HTTP headers and body are available in the Map Outputs section.

- Provide Service

  A web service request can be made to the appliance to start an orchestration job.

- Endpoint specific activities

  These activities include receiving an IDOC from an SAP system and polling activities, such as for getting updated database records or updated records from Salesforce.com.

Error handling

Studio provides the following activities for handling errors:

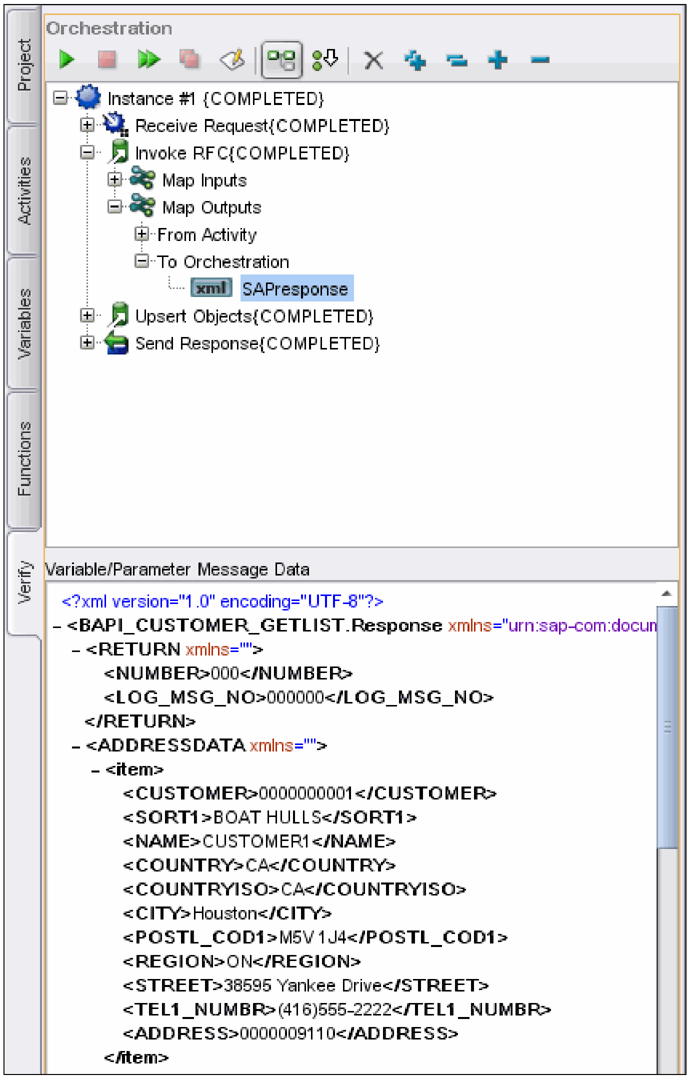
- Catch blocks for the entire orchestration
- Try...Catch blocks containing other activities where multiple Catch blocks can be added
- If...Then logic for checking results from activities
- Retry connections for endpoints
- XML validation

In addition, you can set up Simple Network Management Protocol (SNMP) notifications on the Integration Appliance using the WMC.

Testing

Studio provides the Verify tool to unit test and verify that orchestrations are functioning as expected. An orchestration job can be started, and the orchestration then waits for the starter activity to initiate. The HTTP Post utility provides an easy way to test and verify the orchestration. Detailed logged information about how the data flows through the various steps in the orchestration is provided to help you easily identify problem areas.

Figure 14 shows a test of an orchestration in the Verify tab. Each activity is shown in the top panel, and you can view the data at any point for any activity. The data for the variable SAP Response is shown in the lower panel.

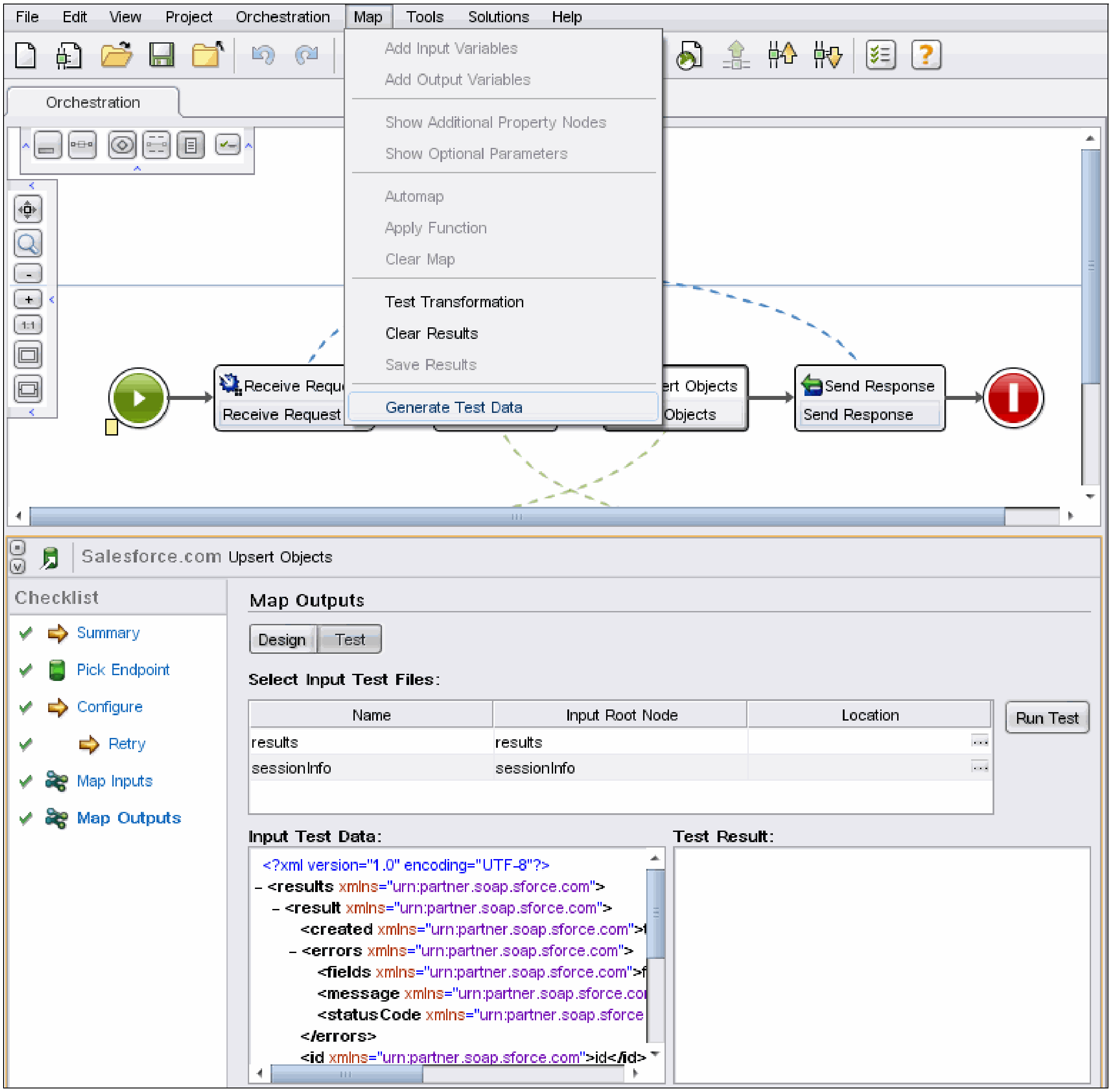


```
  <?xml version="1.0" encoding="UTF-8"?>
- <BAPI_CUSTOMER_GETLIST.Response xmlns="urn:sap-com:docun
  - <RETURN xmlns="">
      <NUMBER>000</NUMBER>
      <LOG_MSG_NO>000000</LOG_MSG_NO>
    </RETURN>
  - <ADDRESSDATA xmlns="">
    - <item>
        <CUSTOMER>0000000001</CUSTOMER>
        <SORT1>BOAT HULLS</SORT1>
        <NAME>CUSTOMER1</NAME>
        <COUNTRY>CA</COUNTRY>
        <COUNTRYISO>CA</COUNTRYISO>
        <CITY>Houston</CITY>
        <POSTL_COD1>M5V 1J4</POSTL_COD1>
        <REGION>ON</REGION>
        <STREET>38595 Yankee Drive</STREET>
        <TEL1_NUMBR>(416)555-2222</TEL1_NUMBR>
        <ADDRESS>0000009110</ADDRESS>
      </item>
```

Figure 14 Example of Verify tab used for testing the orchestration

Individual activities can also be tested with the verify tool. Rather than running the entire orchestration, you can use a Verify Activity menu item.

You can test the mapping for both Map Inputs and Map Outputs in the configuration pane. This testing is useful for confirming any transformations that might occur in the map. Figure 15 shows a test of the map for the Salesforce.com Upsert activity. The ability to generate test data is provided.



Select Input Test Files:

Name	Input Root Node	Location
results	results	...
sessionInfo	sessionInfo	...

Input Test Data:

```
 <?xml version="1.0" encoding="UTF-8"?>
– <results xmlns="urn:partner.soap.sforce.com">
  – <result xmlns="urn:partner.soap.sforce.com">
      <created xmlns="urn:partner.soap.sforce.com">t
    – <errors xmlns="urn:partner.soap.sforce.com">
        <fields xmlns="urn:partner.soap.sforce.com">f
        <message xmlns="urn:partner.soap.sforce.co
        <statusCode xmlns="urn:partner.soap.sforce
      </errors>
      <id xmlns="urn:partner.soap.sforce.com">id</id>
```

Figure 15 Testing the mapping for Map Output for an activity

Reuse: Template Integration Projects

Although Studio provides the capability to build sophisticated integrations from the beginning, Cast Iron also provides Template Integration Projects that encapsulate a specific integration use case between specific endpoints and include preferred practices. A common repository that includes all of the available TIPs that are published is available for download by any customer or IBM Business Partner. These TIPs are authored by internal or external integration experts who have expertise with the endpoints for which they authored the integrations.

Using the TIPs provides the following key benefits:

- Save costs because 80% of the developer effort to build an integration is already codified in the template. The vast majority of the remaining effort involves using a wizard to configure the endpoint and apply any special customizations, such as custom fields or objects mapping and special business logic.
- Readily referenceable preferred practices because each certified TIP is carefully authored and provides a method to reference and learn from other users and Cast Iron experts.
- Flexible to adapt to changes because the TIPs and TIP configuration wizard allow a user to modify an integration when business needs change and additional fields or new business rules must be accommodated. The wizard automatically provides options for a user to select custom fields, change maps, or change business rules.

Deploying projects

After testing the project, publish it to the runtime environment with a single Studio menu command (**File** → **Publish**). After publishing the project, click **Run Configuration** in the WMC to start it. The enabled orchestrations in the project run perpetually with the orchestration jobs created, according to the starter activities used.

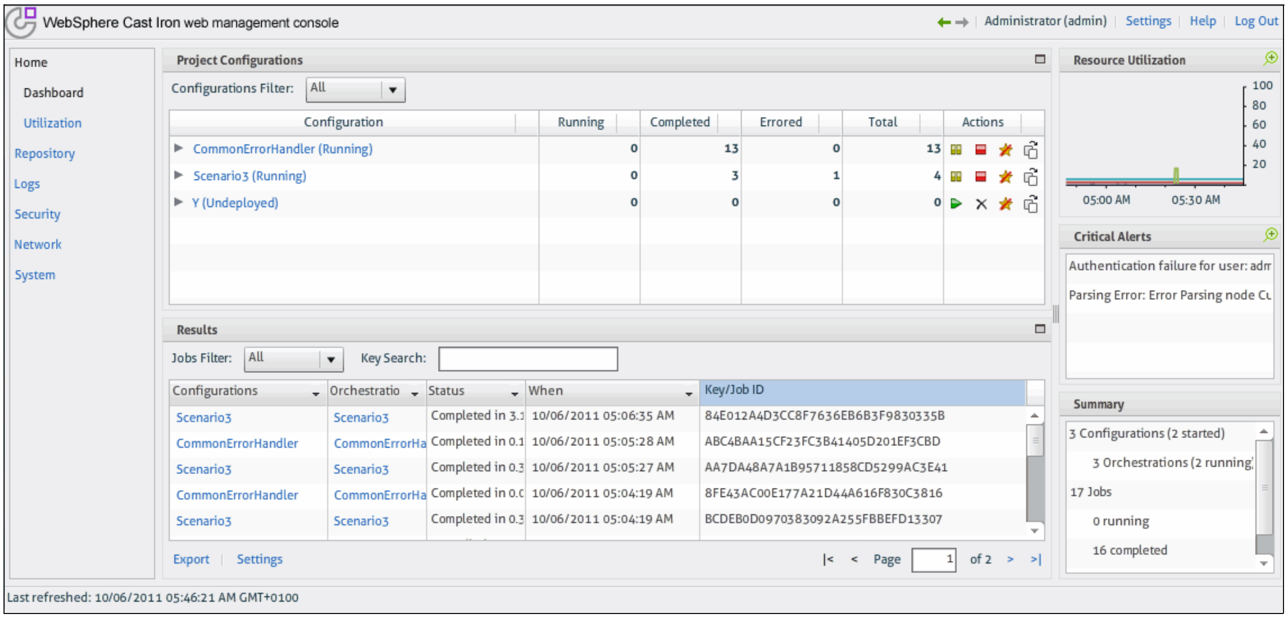
Cast Iron also allows you to export projects from Studio. The exported project can then be imported into the run time (Cast Iron Live or an Integration Appliance) using the WMC. This technique is useful in situations where Studio and the Integration Appliance are on separate networks or when developers do not have the authority to publish directly to the runtime, but instead, export the project to a source control system where the administrator picks up the project for import into the runtime.

Updating projects

Integration rules and workflows are subject to changes as business evolves. Several capabilities are provided that allow administrators to manage and deploy changes to the Cast Iron environments. When you make changes to the orchestrations in Studio, you can easily save the orchestrations to a new version. The Cast Iron Live service also includes a one-click approach to creating a new version on the saved projects. In addition, you can clone published project configurations as a new version to keep a history of changes.

Management and monitoring

You can use the WMC dashboard to obtain an overview of the status of the Integration Appliance, as shown in Figure 16. Each project configuration is shown with a list of jobs, and an instant view of resource utilization is provided. You can drill down into each project configuration to view detailed logs, which are available for each job and for the system.



Project Configurations

Configurations Filter: All

Configuration	Running	Completed	Errored	Total	Actions
▶ CommonErrorHandler (Running)	0	13	0	13	
▶ Scenario3 (Running)	0	3	1	4	
▶ Y (Undeployed)	0	0	0	0	

Results

Jobs Filter: All Key Search:

Configurations	Orchestratio	Status	When	Key/Job ID
Scenario3	Scenario3	Completed in 3.1	10/06/2011 05:06:35 AM	84E012A4D3CC8F7636EB6B3F9830335B
CommonErrorHandler	CommonErrorHa	Completed in 0.1	10/06/2011 05:05:28 AM	ABC4BAA15CF23FC3B41405D201EF3CBD
Scenario3	Scenario3	Completed in 0.3	10/06/2011 05:05:27 AM	AA7DA48A7A1B95711858CD5299AC3E41
CommonErrorHandler	CommonErrorHa	Completed in 0.0	10/06/2011 05:04:19 AM	8FE43AC00E177A21D44A616F830C3816
Scenario3	Scenario3	Completed in 0.3	10/06/2011 05:04:19 AM	BCDEB0D0970383092A255FBBEFD13307

Page 1 of 2

Critical Alerts

Authentication failure for user: adm

Parsing Error: Error Parsing node Cu

Summary

3 Configurations (2 started)

3 Orchestrations (2 running)

17 Jobs

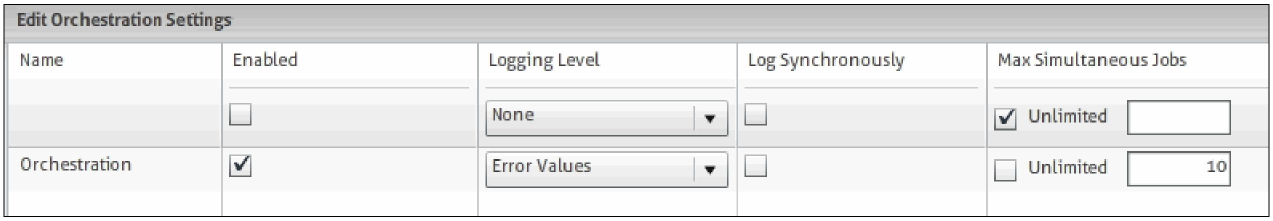
0 running

16 completed

Last refreshed: 10/06/2011 05:46:21 AM GMT+0100

Figure 16 Web Management Console dashboard

Logging level settings, shown in Figure 17, allow for separate levels of error logging from none through to logging all details of each activity.



Edit Orchestration Settings

Name	Enabled	Logging Level	Log Synchronously	Max Simultaneous Jobs	
	□	None	□	☑ Unlimited	
Orchestration	☑	Error Values	□	□ Unlimited	10

Figure 17 Web Management Console orchestration settings

The dashboard also shows the resource utilization of the Integration Appliance, and you can drill down to more detail.

You can configure notifications. A management API is available that enables you to write scripts for all aspects of the Integration Appliance, including monitoring orchestrations, logging, and security.

Availability and scaling

The Cast Iron architecture and design includes the following features for ensuring reliable and fast service and for providing for maximum availability:

- A high availability (HA) topology

- Parallel appliance capability
- Orchestration design to optimize efficiency
- Very low downtime for deployment of updates to projects
- Very low maintenance downtime for upgrades to the appliance

Upgrading to a new release

New releases of Studio and the Integration Appliance are easily installed. A new version of Studio is installed to the PC and does not overwrite the currently installed version. Thus, you can open existing projects and save them with the new version.

For the Integration Appliance, an update file is released (`.vcrypt2` for Cast Iron Hypervisor, `.scrypt2` for the physical appliance). You upload this file to the Integration Appliance through the WMC.

IBM provides advance notice for any scheduled downtime during upgrades to Cast Iron Live. Not all upgrades require downtime.

Cast Iron editions

IBM WebSphere® Cast Iron is available in the following editions:

- WebSphere Cast Iron Standard Edition

  This edition includes the base functionality for the development and deployment of orchestrations and connectivity to the SaaS and mid-market endpoints. The full mapping and function capabilities and the Connector Developer Kit (CDK) are included. The Standard Edition provides access to a repository of reusable templates to help you develop your integrations.

- WebSphere Cast Iron Enterprise Edition

  This edition includes all the capability of the Standard Edition and adds data quality, enterprise connectivity (for example SAP), high availability (requires two appliances), and management APIs to the Integration Appliance. The Enterprise Edition provides access to the repository of reusable templates and gives you the ability to create new reusable templates.

- WebSphere Cast Iron Express Edition

  This edition is a cloud-based offering that provides for integration between Salesforce.com and other data sources. Integrations can be created in minutes through a web browser. Full monitoring and administration is provided.

Standard and Enterprise editions are available on the following form factors:

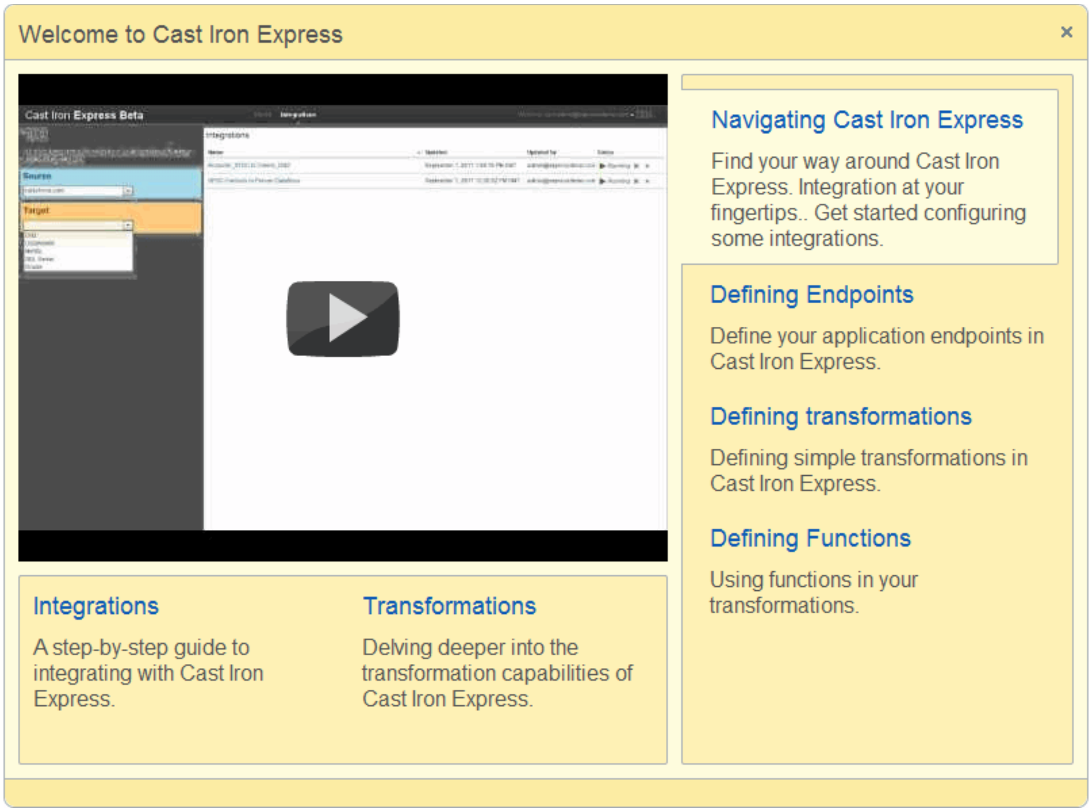
- An appliance running within a virtual machine (WebSphere Cast Iron Hypervisor edition)
- A physical appliance (IBM WebSphere DataPower® Cast Iron Appliance XH40)
- The multi-tenant cloud offering Cast Iron Live

The functionality and orchestration development across these form factors are identical.

Express edition

Cast Iron Express Edition is a cloud solution that allows Salesforce.com administrators to configure, run, and manage integration projects between Salesforce.com and other data sources. The focus on simplicity allows you to configure connections and build maps between application endpoints. You can access, monitor, and administer all projects from a centralized management console.

You use Cast Iron Express Edition through a web browser. Figure 18 shows the Welcome window that opens when you first sign in to Cast Iron Express Edition. Full help with video tutorials is provided from this welcome window.



Welcome to Cast Iron Express

Navigating Cast Iron Express

Find your way around Cast Iron Express. Integration at your fingertips.. Get started configuring some integrations.

Defining Endpoints

Define your application endpoints in Cast Iron Express.

Defining transformations

Defining simple transformations in Cast Iron Express.

Defining Functions

Using functions in your transformations.

Integrations

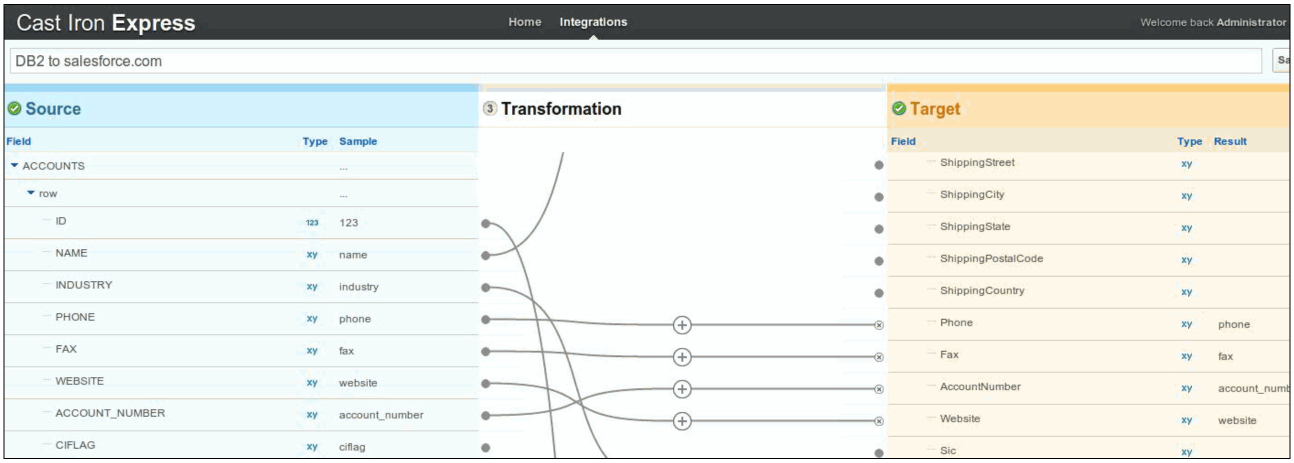
A step-by-step guide to integrating with Cast Iron Express.

Transformations

Delving deeper into the transformation capabilities of Cast Iron Express.

Figure 18 Express Edition welcome window

The user interface for Cast Iron Express Edition is different from the Cast Iron Live user interface. Figure 19 shows an example of the creation of the field mapping between endpoints.



DB2 to salesforce.com

Source

Field	Type	Sample
ACCOUNTS		...
row		...
ID	123	123
NAME	xy	name
INDUSTRY	xy	industry
PHONE	xy	phone
FAX	xy	fax
WEBSITE	xy	website
ACCOUNT_NUMBER	xy	account_number
CIFLAG	xy	ciflag

Target

Field	Type	Result
ShippingStreet	xy	
ShippingCity	xy	
ShippingState	xy	
ShippingPostalCode	xy	
ShippingCountry	xy	
Phone	xy	phone
Fax	xy	fax
AccountNumber	xy	account_numb
Website	xy	website
Sic	xy	

Figure 19 Example of Express Edition mapping capability

You can sign up for a trial of Cast Iron Express Edition at:

http://express.castiron.com/express/

The team who wrote this paper

This paper was produced by the following WebSphere Cast Iron specialists:

Simon Dickerson is a technical pre-sales consultant working for IBM in the UK and Ireland, dedicated to Cast Iron. He worked in a variety of IT roles from training, support, consulting, and focused technical sales for over 10 years. He worked across a number of application areas including content management and document security, mobile applications, and data quality on a variety of platforms and across many industries. Simom holds a degree from the University of Reading in Cybernetics and Control Engineering with Mathematics.

Raji Narayanan is a Senior Product Manager for IBM WebSphere Cast Iron. She has 13+ years of experience in product management and engineering in various start-up, mid-sized and large companies with the more recent ones being IBM WebSphere Cast Iron and Informatica. Raji's domain areas of expertise include cloud & SaaS integration, data integration, data auality, EAI and ERP. Raji has deep experience in driving product strategy and roadmap, release planning and prioritization, sales enablement and product positioning. Prior to her product management role, Raji also spent several years leading product development teams and delivering significant offerings and capabilities to market. Raji holds an MBA from the Indian Institute of Management and a Master's degree in Mathematics from the Indian Institute of Technology.

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks
- Follow us on Twitter:

  http://twitter.com/ibmredbooks
- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806
- Explore new IBM Redbooks® publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm
- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document REDP-4840-00 was created or updated on January 25, 2012.

Send us your comments in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:
  ibm.com/redbooks
- Send your comments in an email to:
  redbooks@us.ibm.com
- Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400 U.S.A.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

Cast Iron®
DataPower®
IBM®
Redbooks®
Redpaper™
Redbooks (logo) ®
WebSphere®

The following terms are trademarks of other companies:

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other company, product, or service names may be trademarks or service marks of others.