

CentraSite

Overview of Run-Time Governance

Version 9.6

April 2014

This document applies to CentraSite Version 9.6.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2005-2014 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors..

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://documentation.softwareag.com/legal/>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <http://documentation.softwareag.com/legal/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices and license terms, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". This document is part of the product documentation, located at <http://documentation.softwareag.com/legal/> and/or in the root installation directory of the licensed product(s).

Document ID: IINM-RG-INTRO-96-20140318

Table of Contents

Preface	v
1 Overview of the Run-Time Governance Environment	1
Run-Time Targets	3
Virtualized Services in CentraSite Control	4
Virtualizing APIs Using the CentraSite Business UI	5
The Run-Time Governance Reference Information	6

Preface

This document provides an overview of CentraSite's run-time governance environment.

The content is organized under the following sections:

Overview of the Run-Time Governance Environment

Describes the basic components of CentraSite's run-time governance environment.

1 Overview of the Run-Time Governance Environment

- Run-Time Targets 3
- Virtualized Services in CentraSite Control 4
- Virtualizing APIs Using the CentraSite Business UI 5
- The Run-Time Governance Reference Information 6

When you use CentraSite to govern Web services at run time, the basic components in the run-time environment are:

- A policy enforcement point such as webMethods Mediator.

webMethods Mediator is a service mediation and policy enforcement application for Web services. The Web services can be SOAP-based, REST-based or plain XML Web services.

Mediator hosts virtual services, which are proxy services that receive requests from consumers on behalf of particular Web services. Mediator can also host virtualized APIs, which are proxy APIs for Web services that have been externalized as Application Programming Interfaces (APIs).

Mediator enforces the run-time governance policies or rules that you define for your virtual services or virtualized APIs (such as security enforcement and audit-trail logging) and handles mediation measures between consumer and provider (such as message transformation and message routing).

Besides serving as an intermediary between consumer applications and native services, Mediator also collects performance statistics and event information about the traffic flowing between consumers and the native services, and reports this data to CentraSite.

Mediator is designed for use with CentraSite. The Mediator application is delivered as a package called WmMediator, which runs on Integration Server. It provides an infrastructure for the run-time governance policies or rules that you define.

- Virtualized services or virtualized APIs.

You can choose to expose your Web services as virtualized services. When you create a virtualized service, you also define:

- Run-time governance policies for the virtualized service.
- The consumer applications that are used to allow consumers to access the virtualized service.

As an alternative to exposing your Web services as virtualized services, you can instead expose the Web services as virtualized APIs, which perform a role similar to virtualized services. When you virtualize an API, you also define:

- Policy enforcement rules for the API.
- API keys, which enable users to securely access the API.

You define all these components in CentraSite, and you store and manage them from CentraSite's UDDI registry/repository.

- Native services, which are Web services that process requests submitted by consumers. If a native service produces a response, it returns the response to the virtualized service or virtualized API, and the virtualized service or virtualized API returns it to the consumer.
- CentraSite, which serves several key roles in the run-time environment. Besides serving as the system of record for the artifacts in the SOA environment (such as virtualized services or virtu-

alized APIs and their run-time governance policies or rules), CentraSite provides the tools you use to define these artifacts and deploy them to Mediator. Additionally, CentraSite receives and logs the performance metrics and event data collected by webMethods Mediator and provides tools for viewing this data.

Subsequent sections of the CentraSite documentation provide the information you need for designing and configuring your run-time governance environment in CentraSite. The following sub-sections provide a brief overview of those subsequent sections:

Run-Time Targets

To use an instance of CentraSite with webMethods Mediator, you must define a run-time target that identifies the specific instance of Mediator that you want to use. A run-time target is a registry object that represents a particular instance of a policy enforcement point (in this case, an instance of webMethods Mediator). The target object specifies the address of the Mediator's deployment endpoint, which is the endpoint that CentraSite uses to interact with Mediator to deploy virtualized services or virtualized APIs. You define a target using the CentraSite Control user interface.

If you use multiple Mediators with an instance of CentraSite, you must create a target for each Mediator. To make the Mediators easier to distinguish when they are viewed in CentraSite, consider adopting a naming convention for targets that clearly identifies to which environment the target belongs (e.g., development, test, production). You can deploy any given virtualized service or virtualized API to one or more run-time targets.

Instead of (or in addition to) using webMethods Mediator for mediation and/or policy enforcement, you can use other third-party products with CentraSite. Support for third-party policy-enforcement and run-time governance tools is available through integrations that are provided by members of the CentraSite Community. These tools are made available through the CentraSite Community Web site at <http://www.centrasite.org>.

In addition to using Mediator, you can use webMethods Insight. Insight is an additional monitoring tool from Software AG that you can use with CentraSite. Insight enables you to see what is happening in real-time with service transactions as they flow across any system. It provides visibility and control at the transaction level to heterogeneous SOA environments. CentraSite provides support for Insight as a target type out-of-the-box. For more information about Insight's uses and capabilities, see the Insight user documentation.

For details, see the section "Run-Time Targets", which follows this "Overview of Run-Time Governance".

Virtualized Services in CentraSite Control

If you choose to expose Web services as virtualized services, you need to:

- Create the virtualized services.

A virtualized service is a service that runs on Mediator and acts as the consumer-facing proxy for a native service that runs elsewhere on the network. You can create virtualized services for Web services that are SOAP-based, REST-based or plain XML Web services. A virtualized service provides a layer of abstraction between the service consumer and the service provider, and promotes loose coupling by providing independence of location, protocol and format between the consuming application and the provider service. You create virtualized services using the CentraSite Control user interface, and store them in the CentraSite registry/repository.

- Create run-time governance policies for the virtualized services.

A run-time governance policy is a sequence of actions that are carried out by a policy-enforcement point (such as Mediator) when a consumer requests a particular Web service through the policy-enforcement point. The actions in a run-time policy perform activities such as identifying/authenticating consumers, validating digital signatures and capturing performance metrics. You create run-time policies using the CentraSite Control user interface, and store them in the CentraSite registry/repository.

- Create consumer applications, which specify the consumers that are allowed to consume services and other assets at run time.

A consumer application is a computer application that consumes (invokes) assets (services, BPEL processes or XML schemas) at run time. Typically, you create a consumer application to specify the consumers that are allowed to consume a particular asset. Then, you include the consumer application in the Consumers profile of the asset.

A consumer application defines the precise consumer identifiers (for example, a list of user names in HTTP headers, a range of IP addresses, etc.). Thus the policy enforcement point (such as Mediator) can identify or authenticate the consumers that are requesting an asset. You create consumer applications using the CentraSite Control user interface, and store them in the CentraSite registry/repository.

- Deploy the virtualized services and consumer applications to Mediator.

For details, see the section “Virtualized Services in CentraSite Control”, which follows this “Overview of Run-Time Governance”.

Virtualizing APIs Using the CentraSite Business UI

As an alternative to exposing your Web services as virtualized services, you can instead expose the Web services as virtualized APIs.

Like virtualized services, virtualized APIs run on webMethods Mediator and act as the consumer-facing proxy for a native service that runs elsewhere on the network. You can create a virtualized API for Web services that are SOAP-based, REST-based or plain XML Web services.

CentraSite provides an Application Programming Interface (API) Management platform that you can implement to expose Web services as virtualized APIs.

CentraSite's Application Programming Interface (API) Management platform enables enterprises to selectively externalize their new and existing assets as APIs across various channels, monitor the interface's lifecycle with an integrated infrastructure, and make sure the needs of developers and application using the API are met.

Application Program Interfaces (APIs) are the new distribution channel for CentraSite assets. With an integrated infrastructure, you can:

- Securely expose your APIs to external developers and partners (that is, any external entities with which your enterprise interacts, such as suppliers and other vendors, dealers and distributors, customers, government agencies, trade organizations and so forth).
- Provide design time and run time governance capabilities to the APIs.

To support this new distribution channel, CentraSite's API Management enables developers, architects and business developers to:

- Publish the right APIs into their organization's central registry.
- Discover APIs and use them to assemble new applications.
- Manage the entire process of creating, publishing, deploying, and retiring APIs.
- Obtain detailed information about an API, including: the list of its consumers; its technical support contacts; its disposition in the development lifecycle; usage tips; and performance data.
- Control access to CentraSite and to the metadata for individual APIs listed in the registry.
- Impose mandatory approval processes to ensure that APIs accepted into the SOA adhere to organizational standards and policies.
- Get notifications on the APIs they use.
- Model the lifecycle process associated with each API and specify the events that are to be triggered when an API transitions from one lifecycle state to another.

For details, see the following sections:

- “API Management”
- “Virtualizing APIs Using the CentraSite Business UI”, which follows this “Overview of Run-Time Governance”.

The Run-Time Governance Reference Information

This documentation set provides a section called “Run-Time Governance Reference”, which provides details about:

- The run-time events and Key Performance Indicator (KPI) metrics that can be collected and reported for each virtualized service or virtualized API deployed in your system.

This section also describes how to configure CentraSite to receive the events and metrics from the policy-enforcement point (such as Mediator) that collects them.

- The run-time actions for virtualized services. (There is a separate set of run-time actions for virtualized APIs.)

You use these actions only when you are using CentraSite Control to create run-time policies for virtualized services. This section provides:

- An alphabetic reference of all actions and their parameters.
 - A listing of the action evaluation order and action dependencies.
 - Some common combinations of actions used to authenticate/identify consumers.
- The run-time actions for virtualized APIs.

You use these actions only when you are using the CentraSite Business UI to create policy enforcement rules for virtualized APIs. These actions are similar in functionality to the actions for virtualized services. This section provides:

- An alphabetic reference of all actions and their parameters.
- A listing of the action evaluation order and action dependencies.

For details, see the section “Run-Time Governance Reference”, which follows this “Overview of Run-Time Governance”.