HP Systinet

Software Version: 10.02
Windows and Linux Operating Systems

User Guide

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 2003-2015 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Intel® Xeon® and Intel® Core i7® are registered trademarks of Intel Corporation in the U.S. and other countries.

Microsoft®,Windows®,Windows® XP and Windows 7® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of TheOpenGroup.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: **http://h20230.www2.hp.com/selfsolve/manuals**

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: **http://h20229.www2.hp.com/passport-registration.html**

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: **http://www.hp.com/go/hpsoftwaresupport**

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

HP Software Solutions Now accesses the HPSW Solution and Integration Portal Web site. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this Web site is **http://h20230.www2.hp.com/sc/solutions/index.jsp**

About this PDF Version of Online Help

This document is a PDF version of the online help. This PDF file is provided so you can easily print multiple topics from the help information or read the online help in PDF format. Because this content was originally created to be viewed as online help in a web browser, some topics may not be formatted properly. Some interactive topics may not be present in this PDF version. Those topics can be successfully printed from within the online help.

Contents

User Guide

This HP Systinet User Guide describes concepts and processes applicable to normal users and contains the following topics:

Concepts

Task Areas

Chapter 1: Introduction to HP Systinet

This guide provides a brief introduction to HP Systinet, the user interface, and some basic account management tasks:

- "API Management" below

- "User Interface Overview" on page 19

- "How to Manage Your Profile" on page 52

- "How to Manage Your Credentials" on page 52

- "How to Switch Domains" on page 53

API Management

API (Application Programming Interface) Management is a new feature that makes Systinet 10 capable of authoring and collaborating REST services in a similar manner to those of web services.
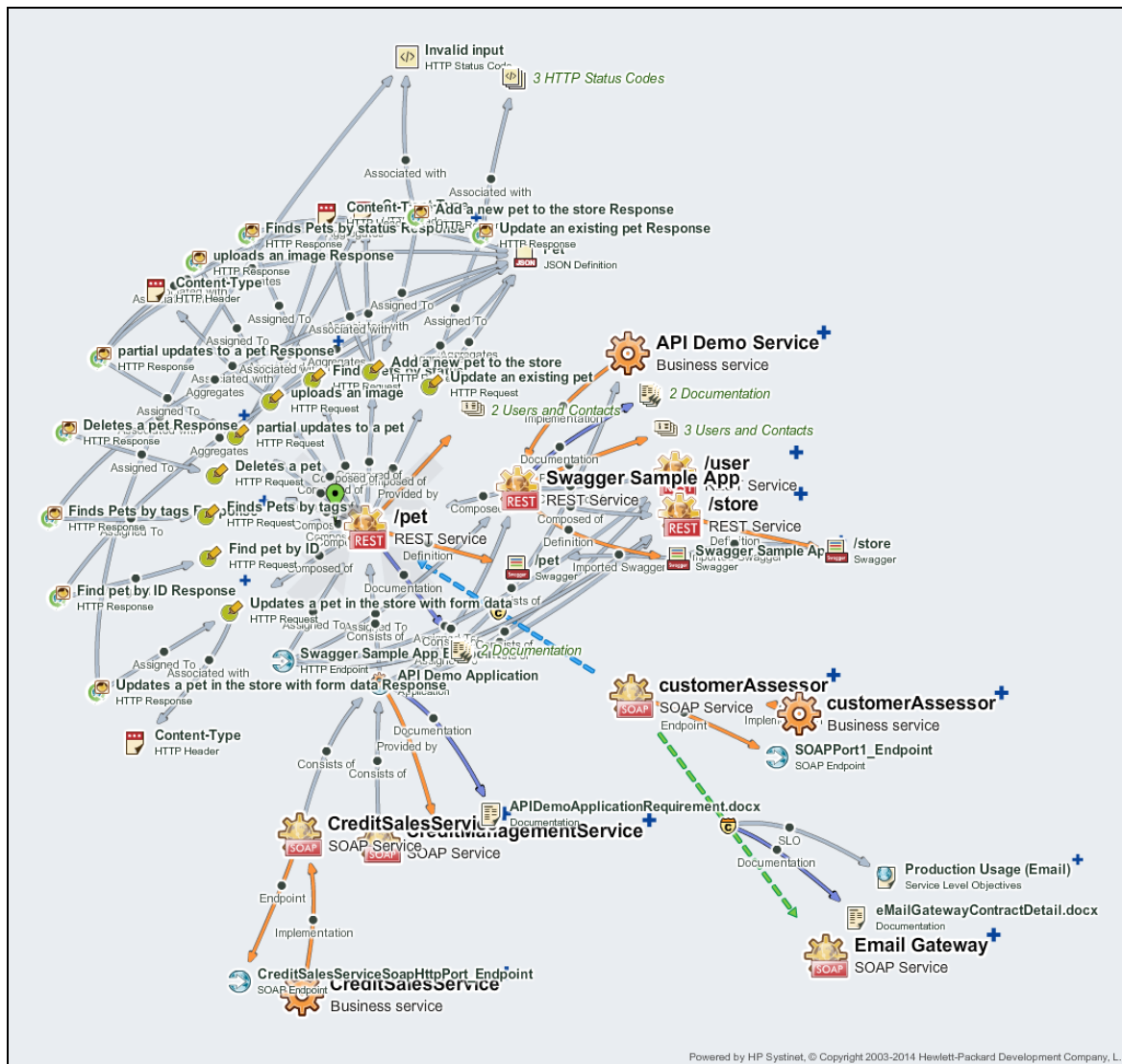
REST service is a new feature for API Management and can be created by the following ways:

- **Manual entry**: Systinet 10 User Interface (UI) contains new components designed specifically for REST services. UI of REST services is customizable and could be reused for other artifact types.

- **CSV import**: Systinet 10 contains bug fixes for CSV import.

- **Swagger import**: Swagger format importing is supported for out-of-the-box (OOTB) and is customizable.

- **WADL import**: WADL format importing is supported for OOTB and is customizable.

- **Discovered artifacts**: REST services could be discovered from runtime Policy Enforcement Points (PEP) as an effort to reverse engineer services model from Run-time to Design-time.

- **New artifacts types**: New artifact types are introduced for API Management.
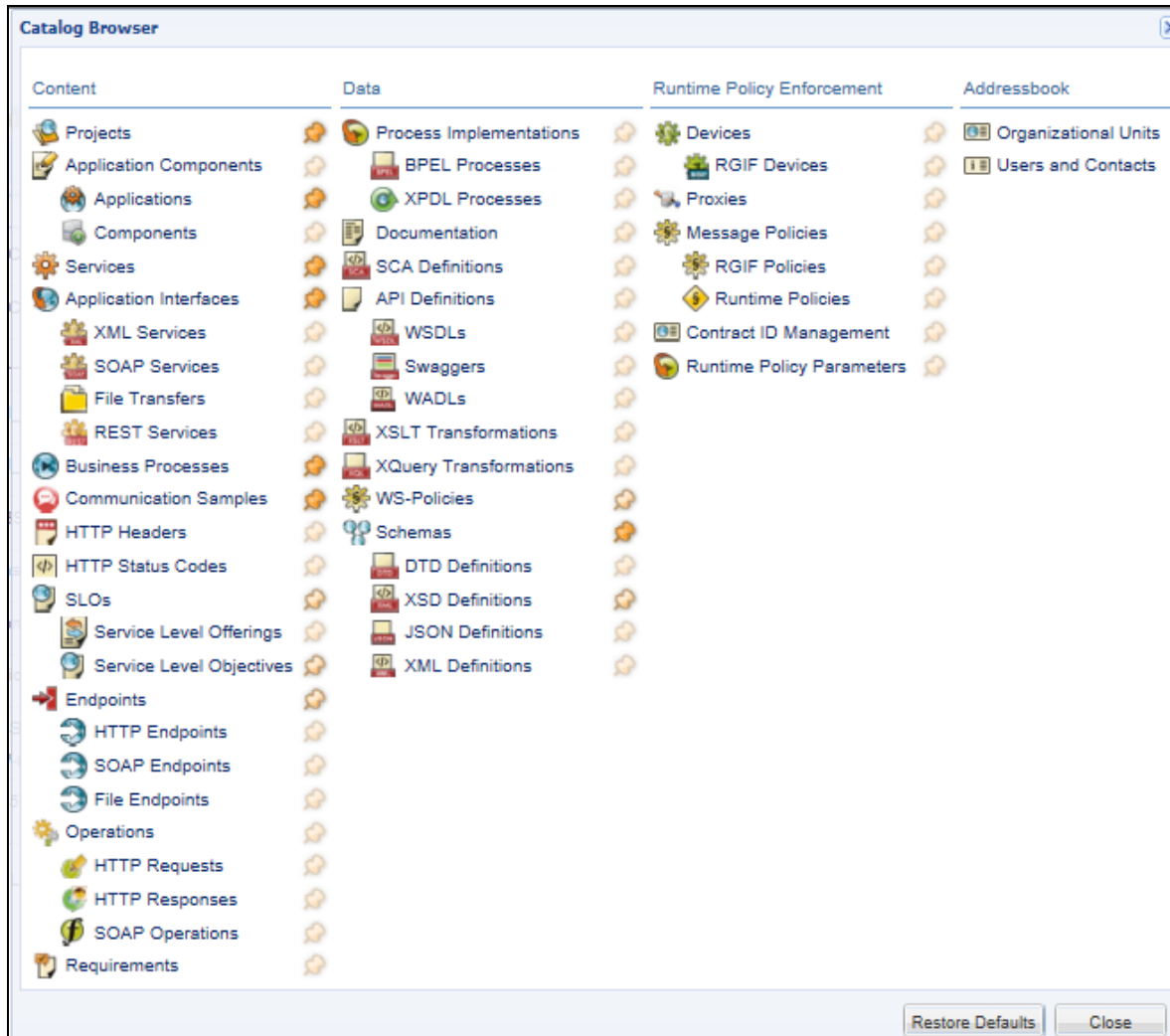
SOA Governance

SOA is no longer an emerging technology but a standard that is a routine part of application development. Currently, most large organizations using SOA in some form are having issues dealing with or transitioning to SOA. Occasionally, these organizations resolve the problems with SOA without explicitly knowing how or why. They often struggle to organize their IT applications, services, or software development departments and system integration. It is clear that in large multinational companies that this can be a difficult task requiring a comprehensive software solution.

SOA is an architectural concept wherein the basic building blocks of software development are services. A system or application consists of a set of autonomous services communicating with each other. Service-oriented architecture is about designing and developing business functionality built on the basis of shared reusable services. By combining these services, an organization can effectively respond to new demands and deliver new functionality better, faster, and cheaper. An important concept here is reusability. Individual modules or components within large-scale information systems may have the same functionality for different implementations and are integrated into systems designed to obtain the finished, certified, and trusted software service.

Over time, the number of services grows and the relationships and dependencies between them become increasingly complex. In the event of a failure of just one service, the entire functionality of the system may become compromised. Not only must an organization have an overview of the developed, offered, and deployed applications and services but must be able to effectively manage this growing infrastructure. Is there a service you need and how to you find it? Who maintains the service and what is its status? What happens when the service is altered? How can I ensure that the services used have the necessary certification? How to ensure that services are in line with both IT and business policies? These are precisely the issues addressed by SOA governance.

SOA governance manages best practices, service development, relationships, lifecycle management, specific quality requirements, the definition and enforcement of policies (polices - rules, restrictions, regulations) that services must meet during lifecycle stages either during the design stage or while running their own services. An important aspect is that of contract management, where it is possible to define the conditions and relationships between the providers and consumers of services. All these procedures may be operated manually or semi-automatically on the basis of written directives, instructions, sent spreadsheets, documents, or emails. However, with a good infrastructure, SOA lifecycle processes facilitate the deployment and operation of SOA within an organization which can be fully automated.

Systinet provides the foundation for SOA governance through its powerful SOA repository, and functionality and features focusing on the real problems in implementing SOA. Systinet is able to integrate with other products from the HP portfolio either in the field of quality management (Quality Center), or the running and monitoring the performance of services or applications from the perspective of end customers (HP Business Service Management).

SOA Repository

The basic software for automated SOA governance is usually an SOA repository. The repository serves as the primary resource for data source metadata (data about data) and other related information and not only for provided services, but also the complete SOA infrastructure. The structure of the content repository is typically defined with an extensible Service Definition Model (SDM). SDM defines a schema model, reflecting the SOA in the organization. The model includes business artifacts (entities), higher abstractions (Services, Business Processes, Applications, etc.), and technical artifacts (Implementations, Operations, Endpoints, XML Schema, WSDLs, etc.) It is important that the definition model is flexible and expandable so that it can effectively adjust to the environment of the organization. Information relevant to governance becomes part of the SOA data model definition. The metadata contains information needed for the creation, searching, validation, and approval of services in different stages of their lifecycle.

In principle, one can say that there are three types of metadata: business information, technical information, and information governance. Business information may be the type of service (application, infrastructure, business services), or the area on which services are focused (banking, health care, etc). Technical information would be service interfaces, security, protocols, etc. Governance information is primarily policy, lifecycle status, and relationships between services.

For details about the Systinet repository and SOA model, see "Catalog" on page 56 and "Service Definition Model (SDM)" on page 57.

Product Services

The main function of the Catalog is to manage content in the SOA repository. Users in specific roles can use and access the catalog in a different ways. Consumers of services are likely to be most interested in the services which provide required functionality under given conditions, and with required

quality. In contrast, service providers want to view a service in the catalog or simply create a register with all pertinent details and offer it to potential consumers. Implementation of catalog content should be automated where possible, particularly with regard to the technical and implementation details of services. For example, an existing web service described by a WSDL is registered in the system by uploading a WSDL. The system then loads the necessary information and creates a breakdown of the service according to the SOA model.

For details about discovering, publishing, and maintaining content in Systinet, see "Search and Browse Overview" on page 55, "Navigator" on page 71, "Authoring Overview" on page 104, and "Artifact Management Overview" on page 153.

Collaboration

In large organizations, it is important to provide a platform that enables users to quickly find the information relevant to them and to enable them to easily share their work with their colleagues. For example, users need to know about changes to services that affect them.

For details about collaboration, see "Collaboration Overview" on page 94.

Lifecycle Management

Services and applications have several lifecycle stages. Each stage has its specific features and important aspects. Each organization may have a set of specific stages requiring different levels of detail. A typical service lifecycle may consist of an initial stage defining the service and giving the combined requirements, with subsequent phases are developing, testing, production, and eventually the service is retired or replaced by a new version or a completely new service. From a larger perspective, this can be divided into stages of development (design time) and production (runtime). To enable a service to move from one stage to another, you need to ensure that the service has the necessary requirements and quality standards completed. In the Catalog, a service is described by a set of artifacts organized into a lifecycle tree, which can be defined for each stage of the lifecycle process. The process of moving between stages is associated with a request for approval and automatic actions are defined, performed, and controlled to alter the artifacts based on the transition.

For more details about lifecycle governance and version management, see "Lifecycle Overview" on page 238 and "Versioning Overview" on page 213.

Lifecycle processes are governed by the set of business rules defined in SOA policies. Service conformity is automatically checked and the rules can be enforced on the services. If a service does not meet the required rules, a remedy may be required. From a technical point of view, policies can be defined using the standard WS-Policy (Web Service Policy). During the initial stages of a service lifecycle, there should be an analysis of requirements and use cases, and a contact or functional architect. In subsequent stages the technical analysis, the means of implementation, interface, operations, testing, access addresses, and of course, other contacts (administrator, architect, project manager, etc.) are important. At the time of contract creation, it is again important to verify whether the service is defined by an SLO (Service Level Objective). When services are in production, other interesting parameters, such as safety, methods of verifying identity and authorization policies for monitoring load, permeability, content, or context-dependent routing may become important.

For more details about policy, see "Policy Management" in the *Concepts Guide*.

Contract Management

Contract Manager enables the management of the relationships between providers and consumers of services. The process works so that consumers find services they would like to use and can make

requests for their use with a defined level of service (SLO). These SLOs are essentially measurable performance indicators, such as throughput, availability, response time, the expected utilization of services, etc. The service provider is automatically notified of the request and can either approve or reject it based on the requirement. The service provider can easily view their consumers, which is important in assessing service popularity and reusability as well as determining who will be affected by a change or disruption to a service. Runtime enforcement may also be based on contract management using the RGIF feature.

For more details about contracts, see "Contracts Overview" on page 223 and "RGIF Overview" on page 255.

Integration

The possibility of simple integration with other systems is an important characteristic of every modern enterprise software company. In terms of the service lifecycle it is useful in the development stage to integrate development and testing tools. During service runtime organizations require access to information systems monitoring services in production environments. The repository can integrate with other systems for automated data publishing, retrieval, and repeated use. Process development services may also be connected to software project management tools. Integration may be broad, but it is necessary to have the appropriate tools and technologies enabling an effectively implemented integration. One example is the remote interface for access to the SOA repository interface exposed via REST API (Representational State Transfer Application Programming Interface). SOA artifacts in the REST architectural style represent source-specific data (resources) which can be accessed via HTTP basic operations for returning, creating, editing and deleting data.

For more details about integration, see "Product Integration Overview" on page 291.

Reporting

Most organizations need frequent and accurate summaries of the status of their repository content. For example, the contractual status of services, or usage of services. Systinet provides an extensive set of default reports with additional customization options.

For more details about reporting, see "Reporting Overview" on page 343.

Administration

The complexity of SOA governance can quickly become difficult to manage and can place large workloads on system administrators. Systinet provides comprehensive administration of users, their access to content and functionality, and the configuration of the product itself.

For more details about administration, see in *Administration Guide*.

User Interface Overview

All pages in Systinet (except the Navigator tab) follow the same format.
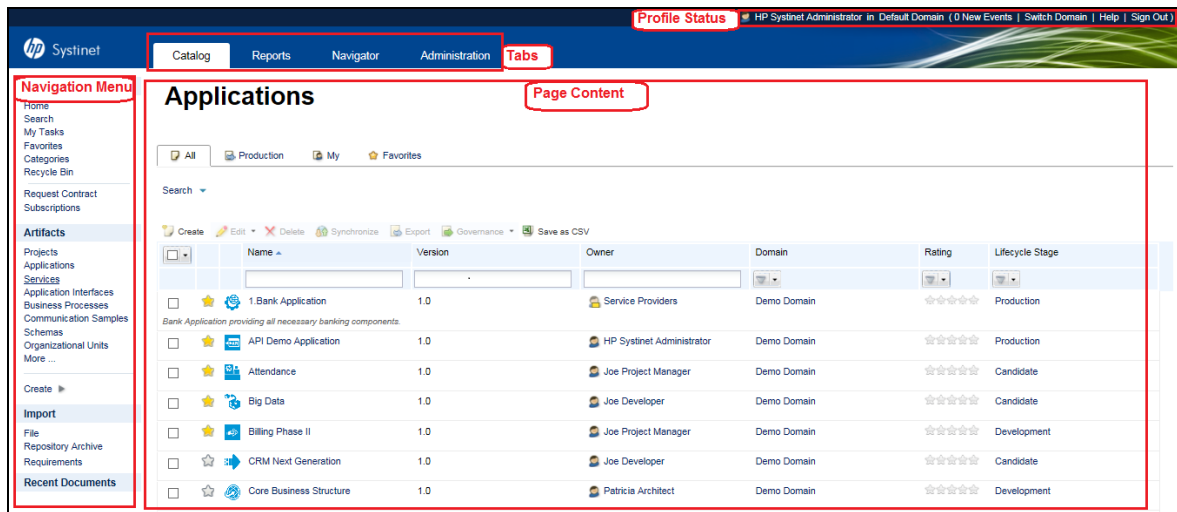
Each page contains the following UI elements:

- **Profile Status**

  The Profile Status area displays your user details and the current domain you are logged in to and access to profile related functionality.
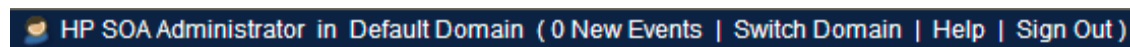
  The Profile Status area provides access to the following functionality:

  - **User Name** - Click your user name to view and edit your profile. For details, see "How to Manage Your Profile" on page 52.

  - **Events** - Click the New Events to open the list of events relevant to you. For details, see "How to Use Events" on page 98.

  - **Switch Domain** - Switch to another working domain. For details, see "How to Switch Domains" on page 53.

  - **Help** - Access this documentation.

  - **Sign out** - Log out of Systinet.

- **Tabs**

  The UI is split into a number of tabs organized by major functional areas and provide quick access to the relevant tab homepages.

- **Catalog**

  The Catalog is the main functional tab for most users providing the following functionality.

  - Search and browse the content of the Catalog.

  - Author and publish new content.

  - Edit, categorize, and manage your content.

  For more details see "Catalog Tab" on the next page.

- **Reports**

  The Reporting tab provides access to the major reporting functionality of Systinet.

  For details, see "Reporting" on page 343.

  - **Navigator**

    The Navigator provides a visual representation of the Catalog content enabling you to explore the relationships and structure of your Catalog content.

    For details, see "Navigator" on page 71.

- **Administration**

  The administration tab provides access to the main administration functions of Systinet.

  For details, see the *Administration Guide*.

  > **Note:** The tabs visible to you and the functionality available in each vary depending on your role.

- **Menu**

  Each tab contains a menu of navigation links specific to the functionality of the tab. The links available depend on your role.

  For details of the functionality of the menu in each tab, see the following topics:

  - "Catalog Tab" on the next page

  - "Reporting" on page 343

  - *Administration Guide*

- **Page Content**

  Systinet consists of hundreds of different pages but they can be categorized into the following types:

- **Tab Homepages**

  Each tab displays a homepage providing a set of specific components for that tab. For details, see the following topics:

  - "Catalog Tab" below

  - "Reporting" on page 343

  - "Navigator Tab" on page 82

  - *Administration Guide*

- **Browse Pages**

  Browse pages enable you to list, filter and search within specific artifact types. For details, see "How to Browse by Artifact Type" on page 65.

- **Artifact Detail Pages**

  Artifact detail pages display all the information about a specific artifact. For details, see "Artifact Detail Page" on page 26.

- Pages displaying content with specific UI formats, such as Categories.

  A few pages display information in specific formats. The topics describing the specific functionality describe these pages.

- Pages to perform specific functionality, such as Create Artifact pages.

  There are numerous pages designed to perform specific functionality. The topics describing the specific functionality describe these pages.

Catalog Tab

The Catalog tab provides access to the Catalog and the functionality to work with its content.

The Catalog tab is the main access point for users in the following roles:

- Business Partner

- Business Analyst

- Service Provider

When you sign in or click the Catalog tab, the Catalog Homepage opens:

Catalog Homepage
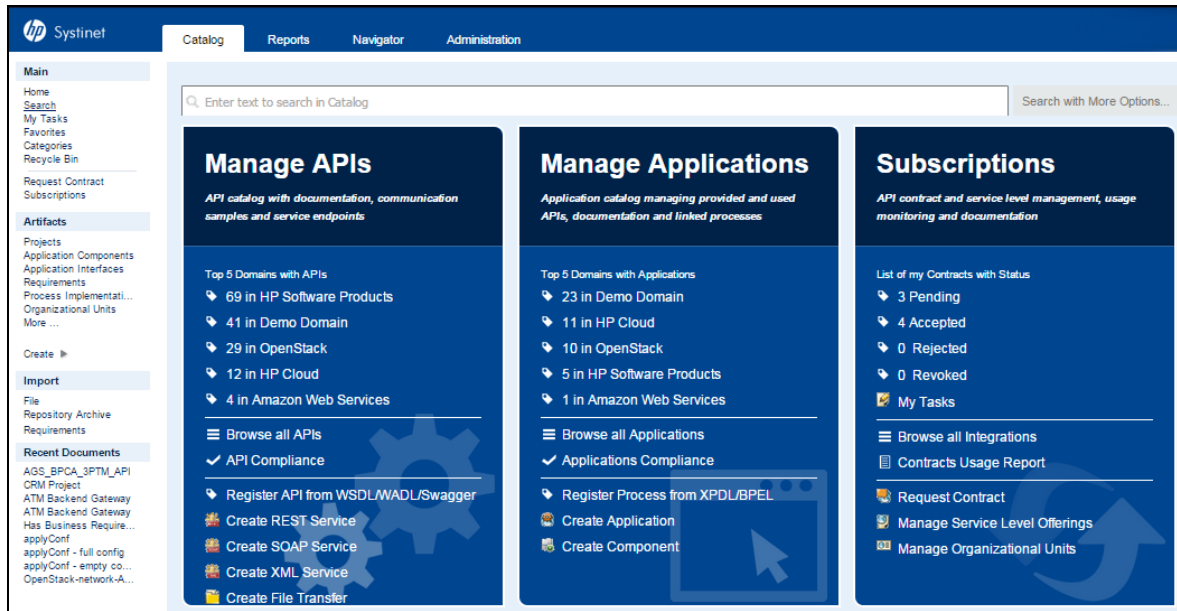
The Catalog Homepage provides an entry point to the Catalog and contains the following components:

- **Manage APIs:** It gives the list of all the APIs with documentation, examples and services endpoints.

- **Applications:** It gives the list of all applications with provided and consumed APIs.

- **Subscriptions:** It gives the list of contracts with their status and usage monitoring.

- **Recent Documents:** The Recent Documents menu displays the last few artifacts that you viewed for easy access. For details, see "How to Access Recent Documents".

Catalog Menu

The Catalog menu provides a set of quick navigation links to service functionality, divided into the following sections:

- **Main**

  The Main menu provides access to commonly used artifact lists, such as Favorites, Contracts, and Categories, as well as providing access to the Recycle Bin. The Home link enables you to return to the Catalog Homepage from any other page you view in the Catalog tab.

- **Artifacts**

  The Artifacts menu is the entry point to browsing artifacts by type and to artifact creation. For details, see "How to Browse by Artifact Type" on page 65 and "How to Create Artifacts" on page 117. Artifact types not shown in the list are accessible and can be added using the Catalog Browser. For details, see "How to Use the Catalog Browser" on page 66.

- **Import**

The import menu lists the following ways to add external content to the repository:

- ○ **File** - Import a data file. For details, see "How to Publish Content" on page 127.

- ○ **Repository Archive** - Import an archive created using export functionality. For details, see "How to Export and Import Artifacts" on page 201.

- ○ **UDDI** - Import content from UDDI Registries. For details, see "How to Import from Registry" on page 296.

- ○ **UCMDB** - Discover and import content stored in *HP Universal Configuration Management Database* For details, see "How to Discover BSM / UCMDB Content" on page 326.

- ■ **Recent Documents**

    The Recent Documents menu displays the last few artifacts that you viewed for easy access. For details, see "How to Access Recent Documents" on page 69.

- • **Search**

    The Search component provides a quick method to locate content in the repository and access to a more detailed search. For details, see "How to Search the Catalog" on page 60.

- • **Subscriptions**

    The Subscriptions component provides a summary of the contracts you are part of either as the provider of the consumer. For more details, see "Contract Management" in the *Concepts Guide*.

> **Note:** The available components and homepage functionality vary according to your role.

Tables in the User Interface

Many pages in the Systinet user interface include tables of artifacts. Typical examples include Search results tables, Browse page artifact tables, and Artifact Details page Details tab related artifact tables.

Screenshot: Browse Application Interfaces Artifact Table

Tables in the UI include the following features:

- Sortable columns.

- Resizeable columns.

- Reorderable columns.

- Show/Hide columns in the drop-down menu at the right of the heading for each column.

- Paging for large numbers of items. The default paging for tables is for 10 items.

- Filters for columns when tables have multiple pages.

  - Dynamic filtering as you type and wildcards (for example, `serv*`) for text columns.

  - Numeric columns allow comparison operator filtering (>, <, =, etc.)

  - Columns with a fixed set of values (for example, artifact type or domain) contain drop-down selection filters.

  - Special filters for specific columns (for example, ratings).

- Bulk operations to perform actions on items selected from the table, such as Delete and Export.

- Create functionality to add an artifact of that type, or from a selectable list.

- Save the content of the table to a CSV file. The primary purpose is to view the content in spreadsheets.

> **Note:** The delimiter in the CSV file is a semi-colon (;).

- **Show/Hide Description** to expand each row to show the artifact description as well.

> **Note:** The format of tables and the functionality they contain varies depending on which page you are viewing, your user role, and the permissions you have for the artifacts in the table. The administrator can also customize the format of tables.

Artifact Detail Page

Artifact Detail pages in the Catalog tab present comprehensive details about each object in the Catalog and provide access to the functionality required perform most of the actions related to individual artifacts.

Each artifact contains a large amount of detail, so Artifact Detail pages are organized into a set of tabs enabling you to focus on particular aspects of the artifact.

Access Artifact Detail pages by clicking the name of an artifact wherever it appears in the Catalog tab.

Screenshot: Artifact Details Page Header

The page header displays the artifact name, artifact type, and artifact ratings. For details about ratings, see "How to Use Ratings" on page 101.

The detailed information about the artifact is contained in the set of Artifact Detail tabs displayed across the page and in the **More** menu.

> **Note:** The available tabs and functionality vary according to your role, the artifact type, and your permissions.

Artifact Details Page Tabs

Tabs	Description
"Artifact Details Page Overview Tab" on the next page	The most important information about the artifact, such as its major properties, the most important relationships, overviews of events, tasks, and discussions, and the set of common context actions.
"Artifact Details Page Documentation Tab" on page 32	Manage the documents associated with the artifact.
"Artifact Details Page Details Tab" on page 32	Additional properties that are not displayed in the Overview tab, tables of related artifacts, and the set of common context actions.
"Artifact Details Page Tree View Tab" on page 35	Shows an artifact structure browser. It either displays composite structure or impact report. In addition to artifact basic information (such as name, owner and type) the browser captures lifecycle information such as the current stage.
"Artifact Details Page Lifecycle Tab" on page 36	Information related to the artifact lifecycle, including as a graphical view of the lifecycle process showing stage details.
"Artifact Details Page Discussion Tab" on page 37	A discussion forum for the artifact.
"Artifact Details Page Categories Tab" on page 38	Categorization details for the artifact.
"Artifact Details Page Specification Tab" on page 41	A page for a full specification of the artifact including edit options with rich text and HTML tagging.
"Artifact Details Page Compliance Tab" on page 42	A detailed breakdown of the current policy compliance status of the artifact.
"Artifact Details Page Contracts Tab" on page 42	Includes all contracts where artifact is either contracted or consuming entity.
"Artifact Details Page Access Rights Tab" on page 45	View and manage the access rights for the artifact.
"Artifact Details Page Contacts Tab" on page 46	View and manage the contacts for the artifact.
"Artifact Details Page History Tab" on page 47	Lists artifact revisions enables users to compare selected two revisions and see the differences between them.
"Artifact Details Page Events Tab" on page 48	Contains all events related to the artifact.
"Artifact Details Page Relationships Tab" on page 49	View all the incoming and outgoing relationships for the artifact.

Artifact Details Page Tabs, continued

Tabs	Description
"Artifact Details Page Metrics Tab" on page 49	View statistics about the artifact that come from external sources, for example BSM and ALM, compliance status, and synchronization options.
"Artifact Details Page Dependency Analysis Tab"	Displays the list of artifacts that are depended on the current artifact that are based on defined relationships.
"Artifact Details Page Impacted Systems Tab"	Displays the list of artifacts that are independent from the current artifact.

Artifact Details Page Overview Tab

The Artifact Details page Overview tab is the default view of an artifact and contains the most important information about the artifact and access to the most common artifact operations.

Screenshot: Artifact Details Page Overview Tab

Artifact Overview Tab Content

Content	Description
Breadcrumb	Breadcrumb path provides information about parent artifacts of certain artifact and allows to navigate to parent of this certain artifact.

Artifact Overview Tab Content, continued

Content	Description
Properties	The first section of the page displays the most important properties of the artifact including options to compare versions, and set the provider. For details, see "How to Compare Versions" on page 218 and "How to Set the Provider" on page 226.
Related Artifacts	The most important relationships display in the Overview tab. In the case of a service, the implementations. Click **Details** to switch to the Details tab which shows all related artifacts. For details, see "Artifact Details Page Details Tab" on page 32.
Latest Events	Displays the latest actions affecting the artifact. Click More... to open the Events tab or the Tool icon to customize the Events component content. For details, see "Artifact Details Page Events Tab" on page 48 and "How to Use Events" on page 98.

Artifact Overview Tab Content, continued

Content	Description
Context Actions	Displays a set of context actions for the artifact relevant to the artifact type, its current status, and your role. The available actions include the following:

- **Favorite** - Add the artifact to your favorites. For details, see "How to Use Favorites" on page 70.

- **Edit** - Open the Edit artifact page. For details see "How to Edit Artifact Properties" on page 163.

- **Attach Document** - Attach a local or remote document to the artifact. For details, see "How to Attach Documentation" on page 123.

- **Download Content** - Copy a referenced data file to your local filesystem.

- **Change Data Content** - Replace an existing referenced data file with a new one. The dialog offers the similar options as for publishing. For details, see "How to Publish Content" on page 127.

- **Change WSDL** - Attach a new WSDL to a SOAP Service using document upload functionality.

- **Change Server Folder** - Change the repository location and name of a referenced data file. For details, see "Server Folders" on page 114.

- **Find Duplicates** - Locate any similar data files stored in different server folders.

- **Create SOAP Service from this WSDL** - Creates any SOAP Service and related implementation artifacts defined by the WSDL using the same functionality as Upload File with WSDL Decomposition set to Implementations. For details, see "How to Publish Content" on page 127.

- **Request Contract** - Create a contract request to use the artifact. For details, see "How to Create Contract" on page 227.

- **Navigator** - Switch to the graphical navigator view for the artifact. For details, see "Navigator" on page 71.

- **Delete** - Delete the artifact and its sub-artifacts. For details, see "How to Delete Artifacts" on page 194.

- **Export** - Create a ZIP archive for the artifact and its sub-artifacts. For details, see "How to Export and Import Artifacts" on page 201.

- **New Version** - Create a new version of the artifact and its sub-artifacts. For details, see "How to Create Versions" on page 216.

- **WSDL Browser** - Browse the content of the WSDL associated with a

Artifact Overview Tab Content, continued

Content	Description
	SOAP Service. For details, see "How to Browse WSDLs and XML Schemas" on page 138. • **Feeds** - Publish a feed for artifact changes or artifact comments. For details, see "How to Use Feeds" on page 101. • **Send Message** - Send an email notification to artifact stakeholders. For details, see "How to Use Notifications" on page 96. • **Change Domain** - Move the artifact to another domain. For details, see "How to Edit Artifact Domains" on page 192. • **Request Approval** - Create a request to approve the current lifecycle stage. For details, see "How to Request Approval" on page 250. • **Generate Document** - Generate documentation report of Application Interface artifact.
Last Edit	Displays the user and time for the last change to the artifact.
Compliance	Displays the current compliance status with an option to recalculate it. For details, see "How to Review Compliance Status" on page 249.
Synchronization	For content imported from an external source such as documentation or for product integration, the Synchronization component displays the current status compared to the external version with options to synchronize, import, or export the artifact. For details, see "How to Synchronize Artifacts" on page 200, "How to Synchronize with Registries" on page 303, and "How to Synchronize Discovered Content" on page 330.
My Tasks	Displays any actions you may be required to perform related to the artifact. For details, see "How to Use Tasks" on page 100.
Keywords	Displays any custom tags applied to the artifact with the option to edit them. For details, see "How to Edit Artifact Keywords" on page 177.
Contacts	Displays the contacts for the artifact with the option to edit them. For details, see "How to Manage Contacts" on page 119.
Projects	The current project the artifact belongs to with an option to change it. For details, see "How to Use Projects" on page 127.
Contract Capacity Management	Displays a table of Contract Capacity Management report with used throughput and availability information of Service Level Offerings. By OOTB, Systinet provides Contract Capacity Management in the Overview page of Application Interface artifacts. You can customize to add this table to any other page. For more details, refer to "How to show/hide Contract Capacity Management section" in Administration Guide.

Artifact Details Page Documentation Tab

The Artifact Details page Documentation tab provides a place to view and manage the documents associated with the artifact.

Screenshot: Artifact Details Page Documentation Tab

Artifact Documentation Tab Content

Content	Description
Documentation Table	Displays the documents associated with the artifact with functionality to manage them and add more. For details, see "How to Attach Documentation" on page 123.

Artifact Details Page Details Tab

The Artifact Details page Details tab displays additional artifact properties and displays tables of related artifacts.

Screenshot: Artifact Details Page Details Tab

Artifact Details Tab Content

Content	Description
Properties	The first section of the page displays the additional properties not shown in the Overview tab
Related Artifacts	Displays tables of related artifacts for each relevant artifact type. These tables enable you to create, add, and remove related artifacts. For details, see "How to Create Artifacts" on page 117, "How to Manage Relationships" on page 120, and "How to Attach Documentation" on page 123.

Artifact Details Tab Content, continued

Content	Description
Context Actions	Displays a set of context actions for the artifact relevant to the artifact type, its current status, and your role. The available actions include the following:

- **Favorite** - Add the artifact to your favorites. For details, see "How to Use Favorites" on page 70.

- **Edit** - Open the Edit artifact page. For details see "How to Edit Artifact Properties" on page 163.

- **Attach Document** - Attach a local or remote document to the artifact. For details, see "How to Attach Documentation" on page 123.

- **Download Content** - Copy a referenced data file to your local filesystem.

- **Change Data Content** - Replace an existing referenced data file with a new one. The dialog offers the similar options as for publishing. For details, see "How to Publish Content" on page 127.

- **Change WSDL** - Attach a new WSDL to a SOAP Service using document upload functionality.

- **Change Server Folder** - Change the repository location and name of a referenced data file. For details, see "Server Folders" on page 114.

- **Find Duplicates** - Locate any similar data files stored in different server folders.

- **Create SOAP Service from this WSDL** - Creates any SOAP Service and related implementation artifacts defined by the WSDL using the same functionality as Upload File with WSDL Decomposition set to Implementations. For details, see "How to Publish Content" on page 127.

- **Request Contract** - Create a contract request to use the artifact. For details, see "How to Create Contract" on page 227.

- **Navigator** - Switch to the graphical navigator view for the artifact. For details, see "Navigator" on page 71.

- **Delete** - Delete the artifact and its sub-artifacts. For details, see "How to Delete Artifacts" on page 194.

- **Export** - Create a ZIP archive for the artifact and its sub-artifacts. For details, see "How to Export and Import Artifacts" on page 201.

- **New Version** - Create a new version of the artifact and its sub-artifacts. For details, see "How to Create Versions" on page 216.

- **WSDL Browser** - Browse the content of the WSDL associated with a

Artifact Details Tab Content, continued

Content	Description
	SOAP Service. For details, see "How to Browse WSDLs and XML Schemas" on page 138. • **Feeds** - Publish a feed for artifact changes or artifact comments. For details, see "How to Use Feeds" on page 101. • **Send Message** - Send an email notification to artifact stakeholders. For details, see "How to Use Notifications" on page 96. • **Change Domain** - Move the artifact to another domain. For details, see "How to Edit Artifact Domains" on page 192. • **Request Approval** - Create a request to approve the current lifecycle stage. For details, see "How to Request Approval" on page 250. • **Generate Document** - Generate documentation report of Application Interface artifact.
Last Edit	Displays the user and time for the last change to the artifact.
Projects	The current project the artifact belongs to with an option to change it. For details, see "How to Use Projects" on page 127.

Artifact Details Page Tree View Tab

The Artifact Details page Tree View tab displays the artifact in the context of its aggregate hierarchy enabling you to explore its impacts and dependencies. For details about aggregation, see "Aggregation" on page 154.

The default view displays the Structure view showing sub-artifacts that depend on the current artifact.

Screenshot: Artifact Details Page Tree View Tab

Artifact Tree View Tab Content

Content	Description
Parent Breadcrumbs	In the default structure view, you can navigate to parent artifacts using the breadcrumbs. **Note:** There may be multiple parents but the breadcrumb only displays one.
Filter	Use the filter to highlight parent or sub-artifacts in the tree matching the input criteria.
Artifact Tree	The tree displays the related sub-artifacts in the Structure view and related parent artifacts in the Impact view, organized by artifact type. Expand groups of artifact types and click artifact names to view their details.
Show Impact / Structure / Lifecycle Tree	Switch between the default Structure view showing the tree of sub-artifacts, the Impact view showing the tree of parent artifacts, and the Lifecycle Tree view showing the lifecycle tree for the artifact.
Navigator	Open the graphical navigator view for the artifact. For details, see "Navigator" on page 71.

Artifact Details Page Lifecycle Tab

The Artifact Details page Lifecycle tab provides a visualization of the lifecycle process the artifact belongs to and the functionality required to participate in its lifecycle.

Screenshot: Artifact Details Page Lifecycle Tab

User Guide
Chapter 1: Introduction to HP Systinet
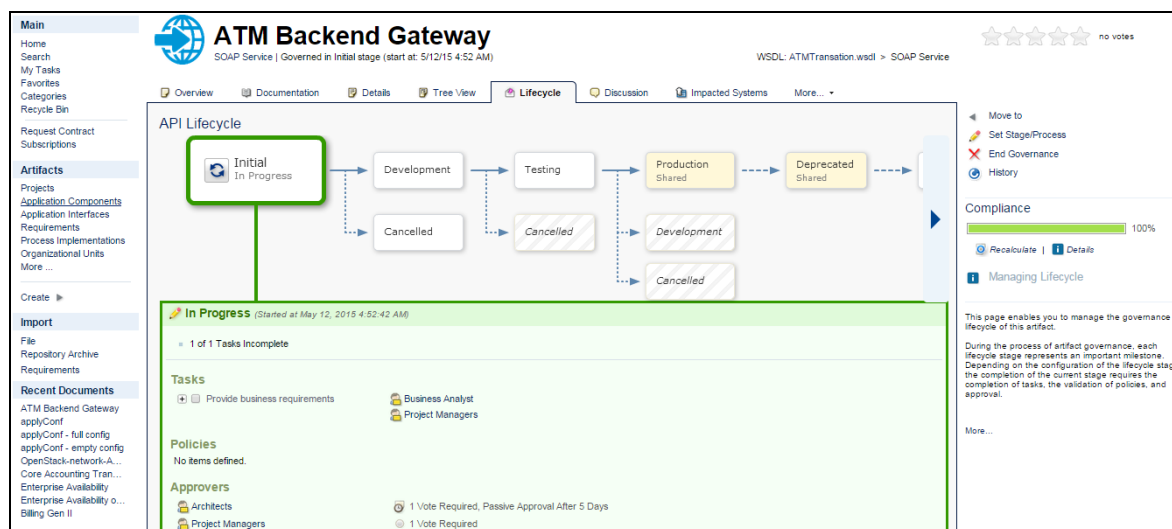
Artifact Lifecycle Tab Content

Content	Description
Lifecycle Process	The top part of the view displays the lifecycle stages for the lifecycle process the artifact belongs to. The larger stage box is the current stage in progress and the box highlighted in green matches the stage detail information below. Click a stage to view its details and use the arrow controls to view stages that are off the page.
Stage Details	The large green box displays the stage details for the stage selected above. For the current stage, the details start with a summary of the current lifecycle status. For all stages the details box displays the tasks, policies, and stage approvers. As a participant in the lifecycle you can use the stage details box to mark lifecycle tasks as complete, to validate policies associated with the stage, and to approve or reject approval requests. For details, see "How to Review/Close Lifecycle Tasks" on page 248, "How to Review/Validate Policies" on page 249, and "How to Approve/Reject Approval Requests" on page 251.
Context Actions	Displays a set of context actions for the artifact relevant to its current lifecycle status and your role. The available actions include the following: ● **Start Governance** - Put a non-governed artifact into an appropriate lifecycle process. For details, see "How to Start Governance" on page 247. ● **End Governance** - Remove the artifact from a lifecycle process. For details, see "How to End Governance" on page 251. ● **Set Stage/Process** - Change the current stage and lifecycle process for the artifact (administrator only). ● **Request Approval** - Request approval for the current lifecycle stage for the artifact. For details, see "How to Request Approval" on page 250. ● **Cancel Approval** - Cancel an approval request. ● **Notify Approvers** - Send a notification email to the stage approvers. ● **History** - View the lifecycle history for the artifact.
Compliance	Displays the current compliance status with an option to recalculate it. For details, see "How to Review Compliance Status" on page 249.

Artifact Details Page Discussion Tab

The Artifact Details page Discussion tab provides a forum for users to comment and discuss the artifact.

HP Systinet (10.02)   Page 37 of 359

Screenshot: Artifact Details Page Discussion Tab
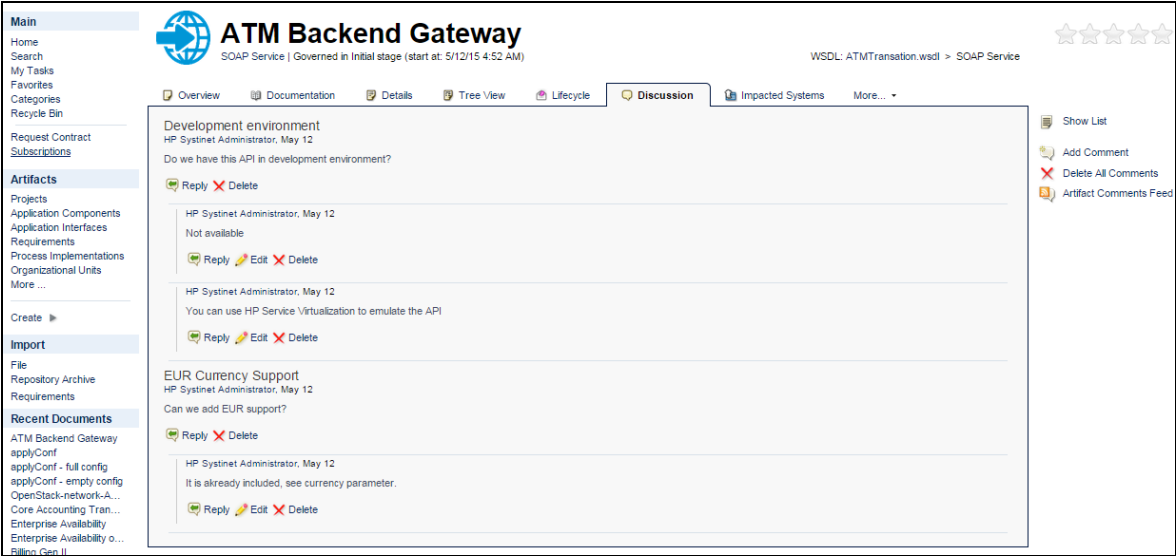
Artifact Discussion Tab Content

Content	Description
Discussion Threads	The main part of the page displays discussion threads and comments about the artifact with options to **Reply**, **Edit**, and **Delete** comments.
Context Actions	Displays a set of discussion context actions for the artifact. The available actions include the following: • **Show List / Threads** - Organize the discussion by comments or by threads. • **Sort** - In the list view, organize the comments by newest or oldest first. • **Add Comment** - Start a new discussion about the artifact. • **Delete All Comments** - Remove all comments and threads about the artifact (administrator only). • **Artifact Comments Feed** - Obtain a feed URL that you can use in a feed reader or in the Reports Tab. For details, see "How to Use Feeds" on page 101.

For details about using the Discussion features, see "How to Use Comments" on page 95.

Artifact Details Page Categories Tab

The Artifact Details page Categories tab provides a place to view and manage the categorizations applied to the artifact.

Screenshot: Artifact Details Page Categories Tab

Artifact Categories Tab Content

Content	Description
Categories Component	The Categories component displays the categorizations applied to the artifact and if you have permissions enables you to modify them or add more categories. For details, see "How to Categorize Artifacts" on page 170.

Artifact Categories Tab Content, continued

Content	Description
Context Actions	Displays a set of context actions for the artifact relevant to the artifact type, its current status, and your role. The available actions include the following:

- **Favorite** - Add the artifact to your favorites. For details, see "How to Use Favorites" on page 70.

- **Edit** - Open the Edit artifact page. For details see "How to Edit Artifact Properties" on page 163.

- **Attach Document** - Attach a local or remote document to the artifact. For details, see "How to Attach Documentation" on page 123.

- **Download Content** - Copy a referenced data file to your local filesystem.

- **Change Data Content** - Replace an existing referenced data file with a new one. The dialog offers the similar options as for publishing. For details, see "How to Publish Content" on page 127.

- **Change WSDL** - Attach a new WSDL to a SOAP Service using document upload functionality.

- **Change Server Folder** - Change the repository location and name of a referenced data file. For details, see "Server Folders" on page 114.

- **Find Duplicates** - Locate any similar data files stored in different server folders.

- **Create SOAP Service from this WSDL** - Creates any SOAP Service and related implementation artifacts defined by the WSDL using the same functionality as Upload File with WSDL Decomposition set to Implementations. For details, see "How to Publish Content" on page 127.

- **Request Contract** - Create a contract request to use the artifact. For details, see "How to Create Contract" on page 227.

- **Navigator** - Switch to the graphical navigator view for the artifact. For details, see "Navigator" on page 71.

- **Delete** - Delete the artifact and its sub-artifacts. For details, see "How to Delete Artifacts" on page 194.

- **Export** - Create a ZIP archive for the artifact and its sub-artifacts. For details, see "How to Export and Import Artifacts" on page 201.

Artifact Categories Tab Content, continued

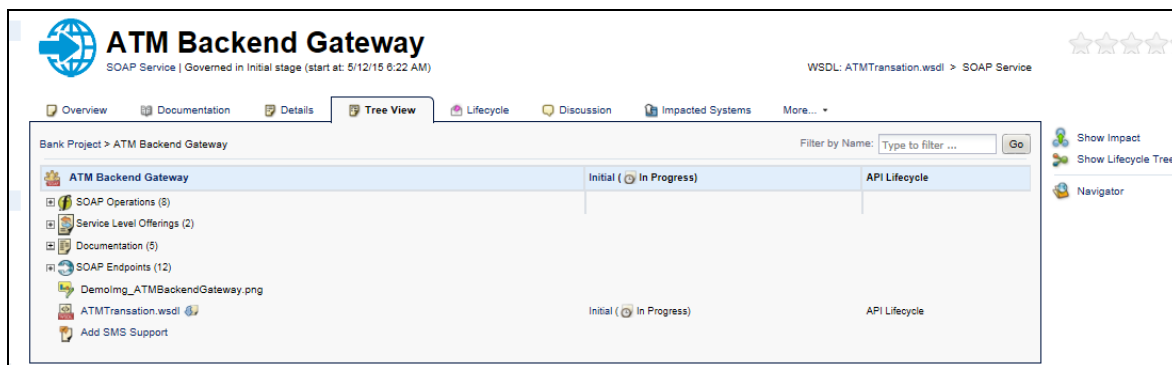| Content	Description
	• **New Version** - Create a new version of the artifact and its sub-artifacts. For details, see "How to Create Versions" on page 216. • **WSDL Browser** - Browse the content of the WSDL associated with a SOAP Service. For details, see "How to Browse WSDLs and XML Schemas" on page 138. • **Feeds** - Publish a feed for artifact changes or artifact comments. For details, see "How to Use Feeds" on page 101. • **Send Message** - Send an email notification to artifact stakeholders. For details, see "How to Use Notifications" on page 96. • **Change Domain** - Move the artifact to another domain. For details, see "How to Edit Artifact Domains" on page 192. • **Request Approval** - Create a request to approve the current lifecycle stage. For details, see "How to Request Approval" on page 250. • **Generate Document** - Generate documentation report of Application Interface artifact.

Artifact Details Page Specification Tab

The Artifact Details page Specification tab provides a place to view and manage a detailed description of the artifact including rich text and HTML support.

Screenshot: Artifact Details Page Specification Tab
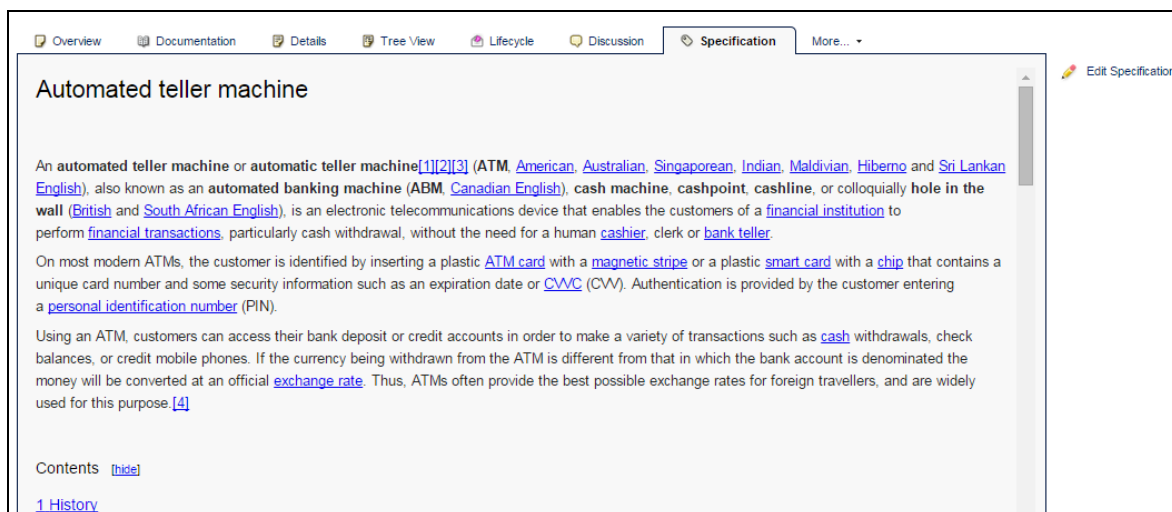
Artifact Specification Tab Content

Content	Description
Detailed Description	Displays the full description for the artifact.
Edit Specification	Changes the Detailed Description area to an input form with rich text and HTML support. For details, see "How to Edit Artifact Specifications" on page 178.

Artifact Details Page Compliance Tab

The Artifact Details page Compliance tab provides a detailed view of the policies that apply the current lifecycle stage that the artifact belongs to with an option to view the XML representation of the resources that the policies validate.

Screenshot: Artifact Details Page Compliance Tab

Artifact Compliance Tab Content

Content	Description
Show	Switch between the Policies view and the Annotated Sources view.
Policies	The table of currently applicable policies, their type, and current status.
Annotated Sources	Provides a detailed XML view of the relevant resources that the policies validate. This view is most useful to assertion and WSDL developers resolving validation errors.

Artifact Details Page Contracts Tab

The Artifact Details page Contracts tab provides a detailed view of the contracts the artifact is part of either as a provider or consumer artifact.

Screenshot: Artifact Details Page Contracts Tab

Artifact Contracts Tab Content

Content	Description
Contract Tables	Displays the contracts that the artifact is part of, either as a provider (Providing table) or as a consumer (Using table). Click a contract to view its details.

Artifact Contracts Tab Content, continued

Content	Description
Context Actions	Displays a set of context actions for the artifact relevant to the artifact type, its current status, and your role. The available actions include the following:

- **Favorite** - Add the artifact to your favorites. For details, see "How to Use Favorites" on page 70.

- **Edit** - Open the Edit artifact page. For details see "How to Edit Artifact Properties" on page 163.

- **Attach Document** - Attach a local or remote document to the artifact. For details, see "How to Attach Documentation" on page 123.

- **Download Content** - Copy a referenced data file to your local filesystem.

- **Change Data Content** - Replace an existing referenced data file with a new one. The dialog offers the similar options as for publishing. For details, see "How to Publish Content" on page 127.

- **Change WSDL** - Attach a new WSDL to a SOAP Service using document upload functionality.

- **Change Server Folder** - Change the repository location and name of a referenced data file. For details, see "Server Folders" on page 114.

- **Find Duplicates** - Locate any similar data files stored in different server folders.

- **Create SOAP Service from this WSDL** - Creates any SOAP Service and related implementation artifacts defined by the WSDL using the same functionality as Upload File with WSDL Decomposition set to Implementations. For details, see "How to Publish Content" on page 127.

- **Request Contract** - Create a contract request to use the artifact. For details, see "How to Create Contract" on page 227.

- **Navigator** - Switch to the graphical navigator view for the artifact. For details, see "Navigator" on page 71.

- **Delete** - Delete the artifact and its sub-artifacts. For details, see "How to Delete Artifacts" on page 194.

- **Export** - Create a ZIP archive for the artifact and its sub-artifacts. For details, see "How to Export and Import Artifacts" on page 201.

Artifact Contracts Tab Content, continued

Content	Description
	• **New Version** - Create a new version of the artifact and its sub-artifacts. For details, see "How to Create Versions" on page 216.
	• **WSDL Browser** - Browse the content of the WSDL associated with a SOAP Service. For details, see "How to Browse WSDLs and XML Schemas" on page 138.
	• **Feeds** - Publish a feed for artifact changes or artifact comments. For details, see "How to Use Feeds" on page 101.
	• **Send Message** - Send an email notification to artifact stakeholders. For details, see "How to Use Notifications" on page 96.
	• **Change Domain** - Move the artifact to another domain. For details, see "How to Edit Artifact Domains" on page 192.
	• **Request Approval** - Create a request to approve the current lifecycle stage. For details, see "How to Request Approval" on page 250.
	• **Generate Document** - Generate documentation report of Application Interface artifact.

Artifact Details Page Access Rights Tab

The Artifact Details page Access Rights tab provides a detailed view of the users, groups, and roles with permissions for the artifact and access management actions for users with appropriate permissions.

Screenshot: Artifact Details Page Access Rights Tab
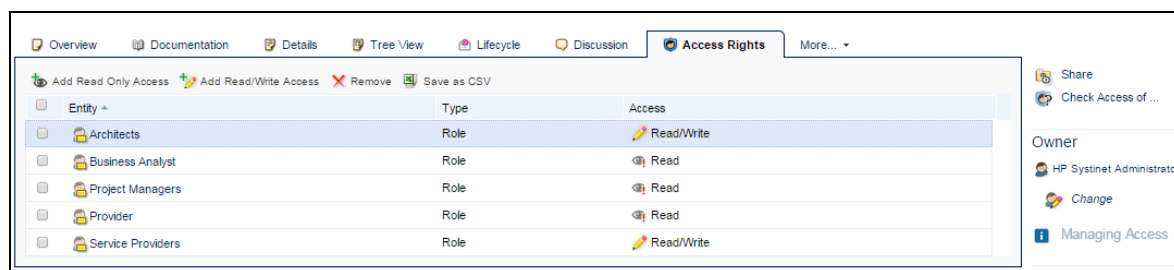
Artifact Compliance Tab Content

Content	Description
Access Rights Table	Displays the users, groups, and roles with permissions for the artifact and their rights. Users with appropriate permissions can manage the access rights using the table actions. For details, see "How to Edit Access Rights" on page 185.

Artifact Compliance Tab Content, continued

Content	Description
Context Actions	Displays a set of discussion context actions for the artifact depending on your role and artifact permissions. The available actions include the following: • **Share** - Make the artifact visible to all users by giving them all read permission. For details, see "How to Share Artifacts" on page 102. • **Check Access Of** - Verify the access rights for a particular user.
Owner	Displays the current user, group, or role ownership for the artifact with the option to change it for users with appropriate permissions. For details, see "How to Edit Artifact Ownership" on page 179.

Artifact Details Page Contacts Tab

The Artifact Details page Contacts tab provides a detailed view and management options for contacts for the artifact organized by their contact type.

Screenshot: Artifact Details Page Contacts Tab

Artifact Contacts Tab Content

Content	Description
Contacts	The set of contacts for the artifact organized by contact role. Users with appropriate permissions can manage the contacts. For details, see "How to Manage Contacts" on page 119.
Send Message	Send a notification email to selected users. For details, see "How to Use Notifications" on page 96.

Artifact Details Page History Tab

The Artifact Details page History tab provides a detailed view of the revisions of the artifact with functionality to compare revisions.

Screenshot: Artifact Details Page History Tab

Artifact History Tab Content

Content	Description
Revisions Table	Displays a list of the revisions with details about the time of the change, who performed it, and lifecycle details. Click a revision number to view its details including options to revert to that revision or use the **Compare Revisions** action to examine the changes in detail. For details, see "How to Compare Revisions" on page 193.

Artifact Details Page Events Tab

The Artifact Details page Events tab provides a detailed view of the actions performed on the artifact.

Screenshot: Artifact Details Page Events Tab

Artifact Events Tab Content

Content	Description
Events Table	Displays a list of the actions performed on the artifact with details about the time of the change, who performed it, and the type of action. Click an event to view the relevant artifact details or use the **Actor** filter to reduce the set of events to those you performed or those performed by other users.

Artifact Details Page Relationships Tab

The Artifact Details page Relationships tab provides a detailed view of the related artifacts organized by outgoing and incoming relationships.

Screenshot: Artifact Details Page Relationships Tab
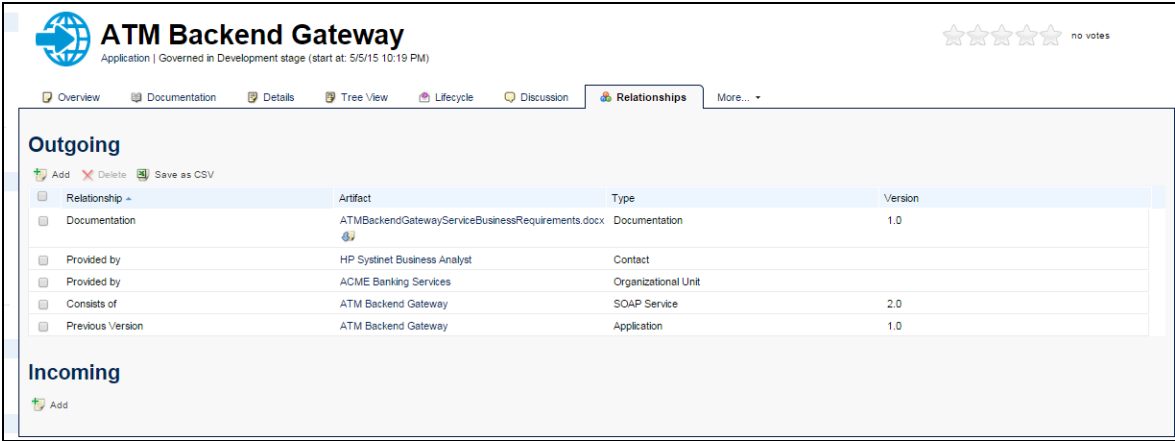
Artifact Relationships Tab Content

Content	Description
Relationship Tables	Displays the set of outgoing and incoming relationships with details of the related artifact, its type and version, and the relationship type.

Artifact Details Page Metrics Tab

The Artifact Details page Metrics tab provides a detailed view of statistics about the artifact provided by integrated products, compliance status, and synchronization options.

Screenshot: Artifact Details Page Metrics Tab

Artifact Metrics Tab Content

Content	Description
Quality (*ALM Server*)	Provides an option to register a SOAP Service in *Application Lifecycle Management* (ALM) and then displays the testing statistics generated by ALM. For details, see "How to Register Services in ALM" on page 337 and "Monitor ALM Quality Statistics of Registered Service" on page 341.
Performance and Availability (*BSM Server*)	Provides performance statistics from *Business Service Management* (BSM) for services imported from BSM. For details, see "How to Monitor BSM Performance Statistics" on page 330.
Compliance	Displays the current compliance status with an option to recalculate it. For details, see "How to Review Compliance Status" on page 249.
Synchronization	For content imported from an external source such as documentation or for product integration, the Synchronization component displays the current status compared to the external version with options to synchronize, import, or export the artifact. For details, see "How to Synchronize Artifacts" on page 200, "How to Synchronize with Registries" on page 303, and "How to Synchronize Discovered Content" on page 330.

Artifact Details Page Impacted Systems Tab

The Artifact Details Page Impacted System tab displays a set of artifacts that impact the current artifact. The displayed information contains Artifact name, Artifact type, Version, Domain, Owner and Lifecycle Stage of impacted artifacts.

Screenshot: Artifact Details Page Impact Systems Tab

Artifact Impact Systems Tab content

Impacted Systems table displays set of artifacts on which the current artifact has impact on. The displayed information contains artifact name, artifact type, version. domain, owner and lifecycle stage of impacted artifacts.

Content	Description
Name	Name of the service.
Version	The version number of the artifact.
Artifact	Type of artifact used.
Domain	Domain associated with the artifact.
Owner	Name of the artifact owner.
Lifecycle Stage	Lifecycle stages of the artifacts.

Artifact Details Page Dependency Analysis Tab

The Artifact Details Page Dependency Analysis tab displays the list of artifacts that the current artifacts are depended on defined relationships.

Screenshot: Artifact Details Page Dependency Analysis Tab

Artifact Dependency Analysis Tab content

Content	Description

Name	Name of the service.
Artifact	Type of artifact used.
Domain	Domain associated with the artifact.
Owner	Name of the artifact owner.
Lifecycle Stage	Lifecycle stages of the artifacts.

How to Manage Your Profile

When you first sign in, Systinet creates a user artifact for you based on your account in the external user store.

You can amend your user details.

To Change your User Details:

1. In the page header, click your user name to open the Artifact Detail page for your user artifact.

2. In the User page Overview or Details tab, click the **Edit** context action to open the Edit page.

3. Edit your details as required. For parameter details, see "Create Contact Page" on page 141.

4. Click **Save** to confirm the changes.

From your user details you can also manage the credentials you use to access remote and integrated product servers. For details, see "How to Manage Your Credentials" below.

If Systinet integrates with an LDAP server, you can synchronize your user details with your external account.

To Synchronize your User Details with your LDAP Account:

1. In the page header, click your user name to open the Artifact Detail page for your user artifact.

2. In the Overview or Details tab, click the **Synchronize** context action and confirm your decision.

Note: Synchronize uses the LDAP cache so changes in LDAP may not be immediately visible in Systinet.

How to Manage Your Credentials

When you use Systinet, you may need to access remote servers that require authorization.

Typical examples include the following:

- Uploading data content stored on a remote server. For more details, see "How to Publish Content" on page 127.

- Importing, exporting, and accessing data in an integrated UDDI registry. For more details, see "UDDI Registry Integration" on page 295.

> **Note:** In this release, if you require credentials to access BSM, and ALM you use the credentials stored with the server artifact when it was set up by the administrator. For more details, see "Product Integration Overview" on page 291.

Systinet enables you to retain the credentials you use to access a remote server or integrated registry in a Credentials Store associated with your profile.

To Manage your Credentials:

1. In the profile status section of any page click your user name to open your profile details page.

2. In the Overview or Details tab, click **Saved Passwords** to open the Saved Passwords dialog box.

3. Do any of the following:

   - To remove stored credentials, click **Delete** for those credentials.

   - To remove all stored credentials, click **Delete All** and confirm your decision.

   - To edit or add credentials:

      i. Click **Edit** for those credentials or click **Add Username / Password**, to open the Edit Username / Password dialog box.

      ii. Enter the resource address, and the credentials to use.

      iii. Optionally, click **Add Address** to add multiple resource addresses that use the same credentials.

      iv. Click **Save** to confirm the new or changed credentials.

How to Switch Domains

Systinet organizes the content of the Catalog into Domains that reflect the structure of your organization. The administrator manages these domains and access to them. You can have access to multiple domains in different roles. For more details about domains, see "Domains Overview" in the *Administration Guide*.

When you sign into Systinet, you sign into your default domain. You can switch domain and change your default domain.

To Switch Domain:

1. In the Account Status bar at the top of every page, click **Switch Domain** to open the Select Domain dialog box.

2. Select the domain you want to use and click **Switch Domain**.

   Your status in the Profile Details section of the page changes and any artifacts you create belong to this domain.

To Change your Default Domain:

1. In the Account Status bar at the top of every page, click **Switch Domain** to open the Select Domain dialog box.

2. Select the domain you want as your default and click **Set Default Domain**.

   The next time you sign in you access the selected domain.

Chapter 2: Search and Browse

This part includes the following topics:

Concepts

- "Search and Browse Overview" below

Tasks

- "How to Search the Catalog" on page 60

- "How to Use Saved Searches" on page 64

- "How to Browse by Artifact Type" on page 65

- "How to Use the Catalog Browser" on page 66

- "How to Browse by Categories" on page 68

- "How to Access Recent Documents" on page 69

- "How to Use Favorites" on page 70

Search and Browse Overview

Search and Browse takes advantage of the structure of the SDM Model to provide various search and browse methods, enabling you to locate the artifacts you want using the criteria that are most useful to you. For more details about the model, see "Service Definition Model (SDM)" on page 57.

The basic search input uses the common name and description properties of all artifacts to provide a method of searching the entire Catalog by the most widely known properties. You can expand this by adding additional criteria based on the artifact type, and specific properties of that artifact type. For more details, see . For more details, see "Catalog" on the next page.

Browsing allows you to start with a more specific criteria such as artifact type or categorization, list all the artifacts within that specific criteria and then narrow your search using the filtering functionality of tables. For more details, see "How to Browse by Artifact Type" on page 65 and "How to Browse by Categories" on page 68.

After locating the content you require, you can also access previously located content using Recent Documents, Favorites, and Stored Search functionality. For details, see .

In addition to these search and browse methods, Systinet provides a visual representation of your Catalog content with the Graphical Navigator. For details, see "Navigator" on page 71.

HP Software also provides a set of IDE plugins with integrated search functionality. For details, see *Plugin for Eclipse* and *Plugin for Visual Studio*.

Systinet also provides collaborative features that are useful in search and browse use cases. For details, see "Collaboration Overview" on page 94.

Catalog

The Catalog provides a single, central storage point enabling you to organize your data and metadata.

The Catalog stores and organizes your business data. The most important entities in your business, such as services and their implementations, users and groups, projects and business processes, are represented in the Catalog by artifact instances. The Catalog provide a central reference point for governance and provision. For more details about artifacts, see "Artifacts and Properties" on page 58.

The Catalog provides the following key features:

- **Visibility**

  Systinet provides numerous features to explore the content of the Catalog, to visualize the relationships between artifacts, and to keep your colleagues up-to-date with the current status of your work.

  - Locate Catalog content and its data and metadata using multiple search and browse methods. For details, see "Search and Browse Overview" on the previous page.

  - Visualize your Catalog content using the Graphical Navigator. For details, see "Navigator" on page 71.

  - Collaborate with your colleagues. For details, see "Collaboration Overview" on page 94.

- **Governance**

  Systinet is primarily a governance platform and provides extensive support for publishing, managing and distributing your content.

  - Author and publish your data content in the Catalog. For details, see "Authoring Overview" on page 104.

  - Edit, categorize, and manage your content. For details, see "Artifact Management Overview" on page 153.

  - Maintain multiple versions of your content. For details, see "Versioning Overview" on page 213.

  - Control the consumption and re-use of your content using Contract Management. For details, see "Contracts Overview" on page 223.

  - Ensure that content conforms to your business policies. For details, see "Policy Management" in the *Concepts Guide*.

  - Control the development of your catalog content using Lifecycle Governance. For details, see "Lifecycle Overview" on page 238.

- Utilize other products to propagate and enhance your content. For details, see "Product Integration Overview" on page 291.

The content of the Catalog conforms to the Service Definition Model. The SDM Model defines an extensible hierarchy of artifact types, with defined properties and relationships. For more details, see "Service Definition Model (SDM)" below.

Users access the Catalog through the following features of Systinet:

- Access and work with Catalog content using the Catalog Tab. For details, see "Catalog Tab" on page 22.

- Visualize Catalog content using the Navigator Tab. For details, see "Navigator Tab" on page 82.

- Administrators access and work with system data using the Administration Tab. For details, see the *Administration Guide*.

- Developers can access and work with Catalog content remotely using WebDav clients or Systinet IDE Integrations. For details, see "WebDAV Compliant Publishing in the *Developer Guide* and *Plugin for Eclipse* and *Plugin for Visual Studio*.

Service Definition Model (SDM)

At the core of Systinet is the Service Definition Model (SDM). The SDM defines the Catalog data schema, reflecting the actual semantics of the data in your enterprise.

The flexibility and extensibility of the SDM ensure that Systinet can keep up with the evolution of your data.

The important concepts of the model are the artifacts types, their properties, and the relationships between artifact types.

Each artifact type represents an abstraction of a particular entity in your organization, for example, a service or organizational unit. The properties of an artifact describe it in business terms and its relationships describe how it fits with the other entities in the Catalog, and by extension, your business.

For example, the following diagram shows artifacts and relationships relating to a service.
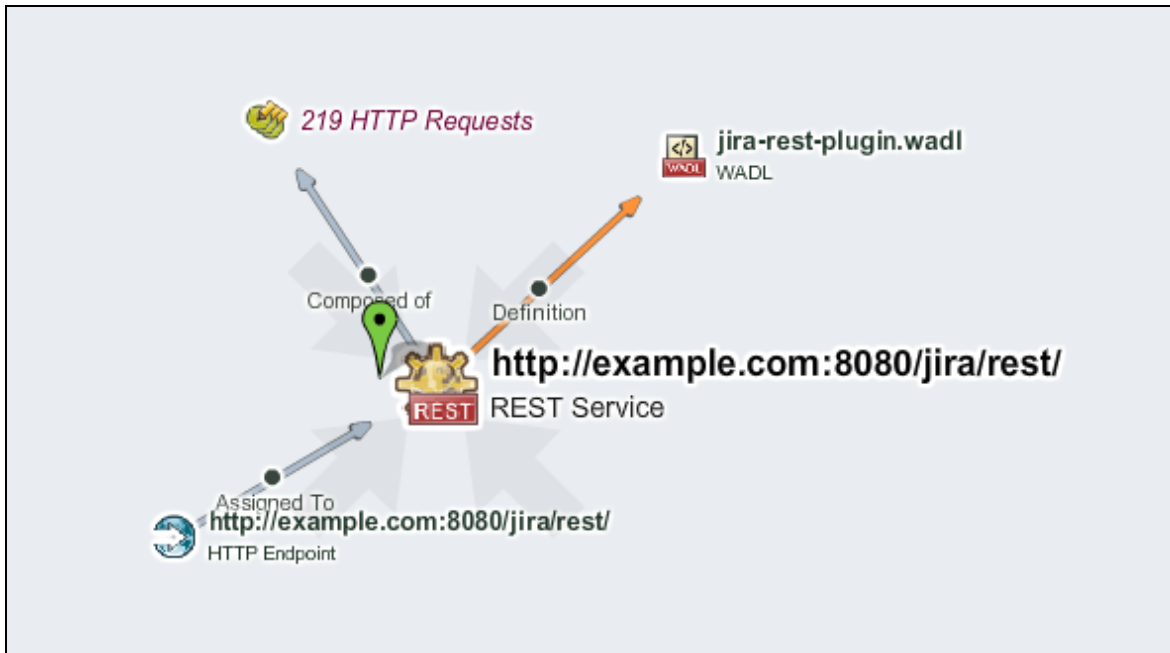
For more details about these concepts, see "Artifacts and Properties" below.

For a more detailed description of the use of artifact types and relationships in a service context, see "Application Modeling" on page 105.

> **Tip:** You can customize the SDM to meet your requirements using Customization Editor. For details, see the *Customization Editor Guide*.

For details of the structure of the model, see the SOA Definition Model Documentation available at `http://host:port/soa/web/doc/sdm/index.html`.

Artifacts and Properties

Artifacts are the entities in the Catalog that represent objects in your business. Each artifact is an instance of an artifact type defined in the SDM Model.

For each artifact type, the SDM Model defines a set of properties that describe the artifact, categorize it, and define its relationships with other artifacts.

This document describes artifacts in the following ways:

- **Artifact Instances**

  Artifact instances are entities in the Catalog. Each artifact instance corresponds to a business object, such as a service, an organizational unit, or a contract.

- **Data Artifacts**

Data artifacts are artifact instances that are associated with specific data content or documents, such as documentation, WSDLs, and XML Schemas.

- **Abstract Artifact Types**

  Abstract artifacts types do not have artifact instances. Their purpose is to define generic artifact types in the SDM Model hierarchy. For example, the SDM Model defines an abstract Implementation artifact type with a set of properties that apply to all implementation artifact types; SOAP Services, XML Services, and Web Applications.

The SDM Model defines a set of properties for each artifact type.

Systinet uses the following property types:

- **Primitive and Complex Properties**

  Primitive and complex properties are attributes of artifact instances. Examples of primitive properties are name, and description. An example of a complex property is address, which consists of further sub-properties; address lines, city, country, etc. These properties describe the artifact instance in terms of the business object it represents.

- **Categorization Properties**

  You can classify artifacts using defined categorization sets. Examples, include geographical location, and business impact. For more details, see "Categorization" below.

- **Relationships**

  Relationships capture logical as well as physical dependencies between artifacts (enabling impact and what-if analysis). Relationship-based properties are either explicitly created by users or implied by the model for a type of source document and related target document. An example of a relationship is the provider relationship between a service and the entity that provides it. The most important concept for relationships is aggregation which enables a single operation on a single artifact to propagate to related artifacts. For details, see "Aggregation" on page 154.

- **System Properties**

  Systinet uses system properties to track information about an artifact such as its revision, access rights, unique identifier, etc. These properties describe the artifact instance in terms of the functionality of Systinet, for example, for security, lifecycle, and artifact history.

For a more detailed description of the use of artifact types and relationships in a service context, see "Application Modeling" on page 105.

Categorization

Categorization provides a way of classifying artifacts according to known and controlled value sets. Artifacts can have properties that are classed as categorization values which can only hold a value from the fixed set of values. Artifacts also use a category bag, enabling them to be classified by a particular categorization even if it is not defined as a property for that artifact type in the SDM model.

Systinet typically uses taxonomies to define sets of fixed values. These are usually referred to as categories. The categories can be organized either hierarchically or as a flat list. For example, the criticality categorization defines a set of values for classifying the business impact of a service failure, with possible values, High, Medium, and Low. Systinet also uses widely recognized categorizations, such Countries (ISO 3166) used to standardize categorization by country or even regions within countries.

Users with the appropriate permissions can categorize artifacts. For details, see .

The Catalog tab provides functionality for browsing artifacts by their categorizations and search enables the use of categorization properties. For details, see .

Lifecycle management can use categories to assign artifacts to different lifecycle processes according to their categorization. For details, see "Automatic Assignment" on page 246.

You can customize the set of categorizations defined in Systinet. For details, see the *Taxonomy Editor Guide*.

How to Search the Catalog

The Homepage Search component in the Catalog tab is the simplest way to find content in the repository.

Screenshot: Homepage Search Component

As you type your search term, Systinet returns a dynamic set of results based on artifacts with matching names, descriptions or keywords. Matched Application Interfaces and Applications are ordered to the top of this list. Either select an item from this list, or complete your search term and click **Go** to open the Search page showing matching results with additional search criteria and filtering options.

Alternatively, click **More Options** to open the Search page enabling more detailed search criteria to be set before applying your search term.

Screenshot: Search Page

The search page provides property-based search within artifact properties and full-text search within attachments to data artifacts and all properties of artifacts.

Property search interprets your search text exactly as you input it. For example, searching for `banking application` locates artifacts with properties containing the whole phrase including the space. Full-text search interprets some characters in a different way. For example, `banking application` interprets `banking` and `application` as separate terms and returns artifacts which contain both these terms which may be separated in different properties or in different places in the attachment.

> **Note:** If you require full-text search, the administrator must enable it. If full-text search is not enabled, only property search is available. For details, see "How to Manage Basic Configuration Options" in the *Administration Guide*.

To Search the Catalog by Properties:

1. In the Search page, specify any of the following criteria for your search:

   - Type in the text to search for. The search returns matching artifacts where the text is a substring of artifact name, description, or keywords.

   - Select an artifact type.

   - Select Keywords to search for.

     > **Note:** You can enter multiple comma separated keywords. Systinet searches for multiple keywords using the AND operator, so the results include artifacts matching all input keywords.

   - Set a Last Modified time.

   - Select the domain that the artifact belongs to.

- Click **Add Criteria** to use additional properties in your search.

  Table: Available Search Criteria Types

Search Criteria Type	Description
Basic	Criteria applicable to all or most artifact types.
Lifecycle	Criteria specifying lifecycle details for artifacts.
Ownership	Specify the owner of artifacts.
Others	Specify properties applicable to specific artifact types.

  For details of specific properties, see "Authoring UI Reference" on page 139.

2. Click **Search** to return artifacts matching your criteria.

Note: Search collates all the specified criteria and returns results using the AND operator. If you use the same criteria twice in your search, the search uses the OR operator for those inputs.

To Perform a Full-Text Search:

1. Select **Full-text** to perform a search of all artifact properties and data attachments.

2. Type in a full-text query which should conform to the following rules:

   Note: Full-Text search performs search on configured database and so follows the rules of the configured database types.

   Basic Search Rules:

   - A full-text search query is a list of words, phrases, and subqueries (collectively called terms) separated by spaces or logical operators AND, OR, and NOT.

   - A word is a string of characters, numbers, and some special characters.

   - A phrase is a list of words enclosed by quotation marks.

   - Enclose search terms including special characters in quotes.

     For example, `"Adam's"`.

   Wildcard Rules:

- Wildcards % and * can be used to represent zero or more arbitrary characters.

  > **Note:** MSSQL only treats % and * as wildcards at the end of words.

- Wildcard _ represents one arbitrary character.

  > **Note:** MSSQL does not support _ as a wildcard.

Logical Operators:

- Logical operators AND and OR, which must be in uppercase, can be placed between terms. A space between terms is a default AND.

- Logical operator NOT, which must be in uppercase, can only be placed before a term if there is at least one preceding term connected by AND.

  For example, `a AND NOT b` and `a NOT b` work, but `NOT b` and `a OR NOT c` do not work.

Subqueries:

- A subquery is a query enclosed in brackets. For example, `a AND (b OR c)`.

Exceptions and Special Cases:

- To search for text containing brackets, wildcards, or other special characters enclose the text in double quotes (") to identify it as text instead of using the special meaning of these characters.

  > **Note:** Some databases do not index special characters and common words (for example, a, the, and, etc.), so it may not be possible to search for them as text.

  > **Note:** In Oracle database, a lexer preference can be created in order to let the database know which special characters are indexed as normal texts. For more details, refer to **Create a custom LEXER preference documentation** in *Installation Guide*.

- For MSSQL, avoid searches using ` (backward apostrophe).

- For MSSQL, to search for words containing < or > in XML/HTML documents, use < or > instead.

Systinet performs your search and matching services are listed in the results table.

Click an artifact name in the table to open its details page.

After entering your search criteria, you can save the search for later use. For details, see

How to Use Saved Searches

After performing a search you can store it for later use.

To Create Saved Searches:

1. After entering your search criteria in the Search page, click **Save** to open the Save Search dialog box.

2. Enter a name for the search, review the search criteria, and click **Save** to create the Saved Search.

> **Tip:** The Saved Search provides a URL that you can use in an Atom feed to track changes. For more details, see "How to Use Feeds" on page 101.

Screenshot: Saved Search with Feed URL
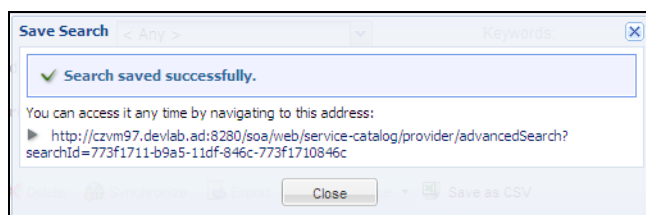
Saved searches are then available in the Search page.

If the option to share the search is selected, the search will be available to other users.

To Access Saved Searches:

1. In the Search page, click **Load** to expand the Saved Search options.

2. Click the name of the Saved Search you want to use to refresh the page with the current set of search results for the saved criteria. If your search is not available in the list, click **Managed Saved** to view all your saved searches.

After creating saved searches, you can edit, delete, or execute them.

To Manage Saved Searches:

1. In the Search page, click **Load** to expand the Saved Search options.

2. Click **Manage Saved** to open the Saved Searches dialog box.

3. Click the name of a search to view its details.

4. Do any of the following:

- To change the name of the search, input a new name and click **Save**.

- To delete the search, click **Delete**.

- To execute the search, click **Load**.

- To share the search select the **Shared search** option.

How to Browse by Artifact Type

The Browse pages provide an entry point for finding an artifact of a particular type.

Access the Browse page name by clicking the artifact type name in the **Catalog** tab Artifacts menu which shows the most important artifact types. If an artifact type is not in the list, click **More…** to open the Catalog Browser which displays all the artifact types you can access and also enables you to customize the Artifacts menu. For details, see "How to Use the Catalog Browser".

Screenshot: Services Page

Browse pages consist of the following components:

- **Filters**

  Select a filter to reduce the content of the results table to those services matching the selected filter.

  The default filters for Services are **All**, **My**, and **Favorites**. The selected filter controls both the content of the results table and the criteria available in the search component.

> **Note:** The available filters vary according to the artifact type.

- **Results**

  A table of services matching the selected filter, search, and facet criteria.

  The results table may still contain many service artifacts, even after selecting a filter or using search criteria. Depending on the artifact type, the table may contain additional criteria, known as facets, to enable you to further refine your search.

  For example, the Services page results table enables you to select which types of service to show (Business, Application, and Infrastructure).

  Click a service name in the results table to open its details page.

- **Search**

  Further reduce the services shown in the table using the search component.

  For any selected browse filter, the results table may list many services. The search component enables you to find a particular service of interest.

  To Perform a Browse Page Search:

  a. Click **Search** to expand the browse page search component.

  b. Optionally, click **Add Criteria** to show additional search parameters and select additional criteria if they are required for your search. For example, Version or Categories.

  c. Enter your search criteria, and click **Search**.

  > **Tip:** You can enter multiple comma separated Keywords. Multiple Keywords are searched using the AND operator, so the results only include artifacts matching all Keywords.

  The results table refreshes with the services matching your inputs.

How to Use the Catalog Browser

The Artifacts menu only shows the most important artifact types. The Catalog Browser enables you to browse and create all the artifact types that your role can access.

Screenshot: Catalog Browser
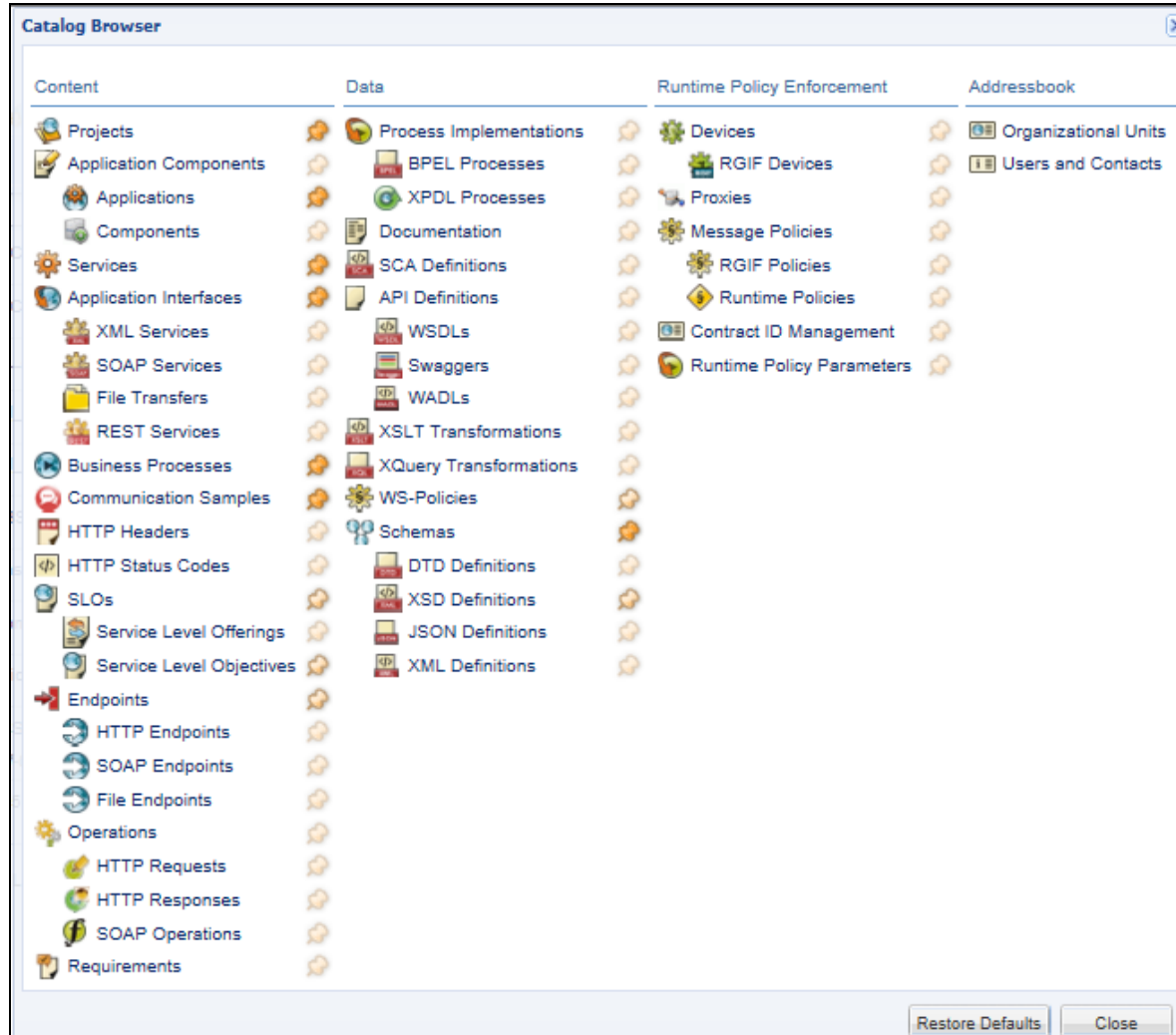
To Access the Catalog Browser:

- Do one of the following:

  - Click **More...** in the Artifacts menu to open the Catalog Browser in browse mode.

  - Place the cursor over **Create** to open the create artifact listing, and then select **More...** to open the Catalog Browser in create mode.

In both modes, the Catalog Browser displays a list of all the artifact types that you can browse or create split into Content, Data, and Address Book categories.

Click an artifact type name to open the relevant browse or create page for that artifact type. For details, see "How to Browse by Artifact Type" on page 65 and "How to Create Artifacts" on page 117.

The Catalog Browser enables you to customize the Catalog tab Artifacts menu.

To Customize the Artifacts Menu:

1. In the Catalog tab Artifacts menu, click **More...** to open the Catalog Browser.

   The pin icon next to each artifact type controls the appearance of that artifact type in the Artifacts menu for browsing and the list of artifact types in the create list.

2. Do any of the following:

   ■ To add an artifact type to both listings, click a pin to highlight it.

   ■ To remove an artifact type from both listings, click the pin to remove the highlight.

   > **Note:** Abstract package artifact types, such as Implementation, do not appear in the create artifact list. Only artifact types that can have instances are listed.

How to Browse by Categories

Alternatively, click the **Categories** link in the left menu to open the Categories browsing page.

Screenshot: Categories Browsing Page

The Categories page lists all taxonomies used to categorize services in the repository with a breakdown by the number of services assigned to each category.

Select a taxonomy or category to open a page for that taxonomy or category with a table listing all the matching services in the repository.

Screenshot: Artifacts in Categories Page

Select an artifact to open its details page.

How to Access Recent Documents

Systinet tracks the artifacts you look at and enables you to easily access them again.

Whenever you open an Artifact Details page, Systinet adds the artifact to the Recent Documents menu in the Catalog tab.

Screenshot: Recent Documents Menu

Click an artifact name in the Recent Documents menu to open its Artifact Details page.

The Graphical Navigator provides the same functionality for recent documents. For details, see "How to Access Recent Artifacts in Navigator" on page 92.

How to Use Favorites

Systinet enables you to mark artifacts of interest as favorites. This enables easy access to those artifacts using favorites functionality in the Catalog Homepage, Browse Pages, in the Graphical Navigator, and in tables of artifacts.

To Add or Remove Artifacts from Your Favorites:

- Do one of the following:

  - In the Artifact Detail page Overview or Detail tab, click the **Add to Favorites** context action to add or the **Favorite** context action to remove the artifact from your favorites.

  - In tables of artifacts with a favorites column, for example in Browse pages, click the star icon for the artifact you want to add or remove from favorites.

    Note: Artifacts marked as favorites are marked with a gold star, non-favorites with a grey star.

  Artifacts marked as favorites appear in the Favorites page.

Screenshot: Favorites Page

The Favorites page lists all your favorite artifacts with selectable artifact type facets to filter the table. Click an artifact name to open its details page.

To remove an artifact from your favorites, click the gold star.

The Graphical Navigator provides additional functionality for Favorites. For details, see "How to Access Favorites in Navigator" on page 91.

Chapter 3: Catalog Visualization

This part contains the following topics:

Concepts

References

Tasks

Navigator

The Navigator tab provides visual representation of the content of the Catalog. Instead of focusing on the details of single artifacts, it helps to understand the structure and dependencies of applications, services, and processes in the Catalog. As its name suggests, it is not only an alternative view but also a navigator that enables the exploration, browsing, and traversing of your Catalog content in a graphical way.
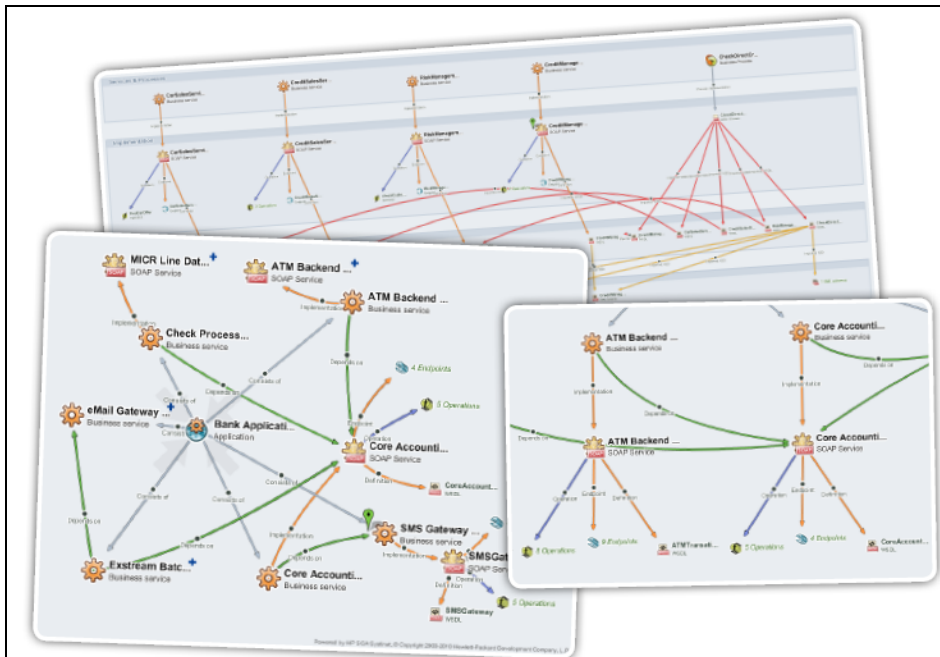
The main features of the Graphical Navigator are:

- **Catalog Browsing**

  Simple and rapid navigation of the Catalog content. For details, see "Navigator Tab" on page 82.

- **Navigator Layouts**

  The navigator enables you to view artifact graphs using different layouts, organizing the artifacts into various structures to demonstrate a particular aspect of their relationships. You can also create customized layouts to focus on particular relationship structures. For details, see "Navigator Layouts" on page 78.

- **Filters**

  The navigator provides a set of role-based filters focusing on the artifacts of interest to each role, as well as the ability to create custom filters. For details, see "Navigator Filters" on page 76.

- **Saved Graphs**

  The graph is interactive, enabling you to modify its appearance by moving, expanding, and collapsing nodes. You can add a particular graph layout to your saved graphs, enabling you to return at a later time to a particular layout for a particular graph. For details, see "How to Use Saved Graphs" on page 91.

- **Impact and Domain Highlights**

The navigator enables you to view the impact of change to an artifact within its graph of relations, or to view the artifacts within a particular domain. For details, see "How to Highlight Impact and Dependency" on page 93 and "How to Highlight Domains" on page 93.

- **Screenshots**

  Create an image file copy of particular graph layouts. For details, see "Navigator Toolbar" on page 83.

The navigator can also help collaboration between Systinet users. For example, an architect can demonstrate the structure of an application to the developers who implement it, and the developer can explain dependencies the application has on corporate schemas to QA engineers.

The main benefits of the navigator vary according to the role of the user:

- *Architects* can analyze the structure and dependencies of applications, services, and business processes.

- *Developers* can quickly understand the complex structure of service interfaces defined by interconnected WSDLs and XSDs, and understand the business context of service implementations.

- *QA Engineers* can understand the context and structure of dependencies for tested services and review the impact of any changes in order to focus only on those services affected by a change.

- *Operation Engineers* can assess the purpose and business context of services and applications.

- *Managers* can use the navigator to improve communication with architects and business analysts.

Your application infrastructure can quickly become extremely complex, so it is essential to have the right tools to help you to understand, present, discuss, and optimize your content.

HP Software also provides a set of IDE plugins with integrated functionality that opens the Navigator view within the IDE. For details, see *Plugin for Eclipse* and *Plugin for Visual Studio*.

Example: Banking Application

Consider a Banking Application that consists of the following:

- Six interrelated banking services.

- Application documentation.

Each service can be examined in more depth. For example, the Core Accounting Service consists of the following:

- Three documents about the service.

- A SOAP Service implementation.

- A WSDL defining the implementation.

- Five operations of the implementation defined by the WSDL.

- Four endpoints of the implementation defined by the WSDL.

- Five documents about the implementation.

- Two sets of service level objectives which are applicable to the implementation.

This description is rapidly becoming more and more complex, even though it is only the structure of a single application and one of its services.

The Navigator provides an interactive overview of this complexity in a single image.
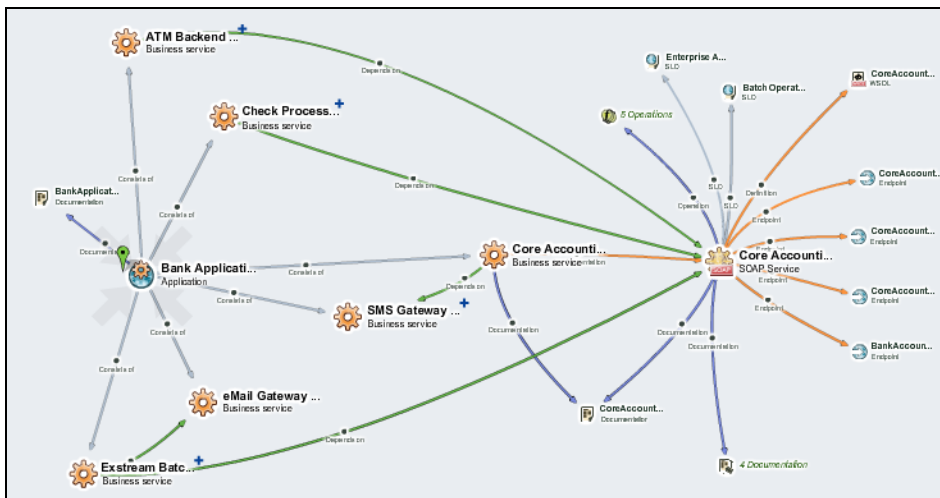
Understanding the Navigator Graph

Each node in the artifact represents an artifact or a group of artifacts, denoted by an artifact icon.

> **Tip:** The icon for each artifact type is customizable. For details, see the *Customization Editor Guide*.

The content of the graph is determined by the filter. The navigator contains a number of default filters targeted at particular user roles and you can create your own custom filters. For details, see "Navigator Filters" on page 76.

The artifact and relationship annotations are determined by the Settings control. For details, see "Navigator Toolbar" on page 83.

The structure of the graph is determined by the layout. The navigator contains a number of default layouts highlighting particular relationship structures and you can create your own custom layouts. For details, see "Navigator Layouts" on page 78.

An artifact node can display in the following ways:
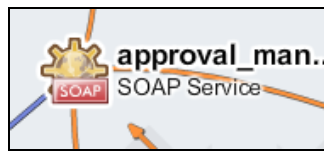
approval_man.. SOAP Service	A fully expanded node representing a single artifact with all its relationships expanded and shown in the graph. This diagram shows a SOAP Service with several outgoing and incoming relationships. The relationship types are labeled and color-coded as shown in the Relationships information box.
A UDDI Node Organizational Unit	A node marked with a green flag. This node is the subject of the Artifact Info box. Click a node to make it the subject for the Artifact Info box. For details, see "Navigator Information Boxes" on page 85.
Account_Soap... Business service	A collapsed node representing a single artifact with further relationships collapsed and hidden in the graph. This diagram shows a Service with an outgoing relationship. Further relationships are collapsed and represented by **+**. Double-click the node, use the **Expand Tree** node context action, or the **Expand Tree** action in the Artifact Info box to view these relationships. **Note:** If an artifact node contains many collapsed relationships, expanding it may take some time.
8 Operations	A collapsed node representing a group of artifacts. This diagram shows a set of operations collapsed into a single group. Double-click the group, use the **Expand** node context actions, or the **Expand** actions in the Artifact Group Info box to view the individual artifacts in the group. **Note:** If a group contains many artifacts, expanding it may take some time.
ApprovalMan... Endpoint	A node marked with an orange flag. This node is the subject of the current impact view. For details, see "How to Highlight Impact and Dependency" on page 93.
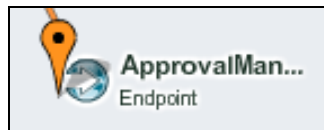
The following node context actions are available:

Node Context Actions

Context Action	Description
View in Catalog	Exit the Navigator tab and open the artifact detail page.
Impact Highlight	Highlight the impact tree for the artifact. For details, see "How to Highlight Impact and Dependency" on page 93.

Node Context Actions, continued

Context Action	Description
Domain Highlight	Highlight artifacts in the selected domain. For details, see "How to Highlight Domains" on page 93.
Center Artifact	Make the artifact the focus of the view and show its relationship graph. (Concentric Radial layout only).
Collapse Tree	Collapse outward relationships from this artifact.
Expand Tree	For collapsed nodes, expand the relationships to hidden artifacts.
Expand	Open the Expand Artifacts from Group dialog box to select the artifacts to view separately from the group.
Expand All	Expand all the artifacts in a group.

You can interact directly with the graph and artifact nodes using the following functionality:

Graph Functions

Function	Description
Move Graph	Move the entire graph by dragging and dropping in an empty part of the graph.
Move Node	Move a node and its related sub-graph by dragging and dropping the artifact node. Hold **CTRL** and drag and drop to only move the selected node.

Navigator provides additional functionality to change the format and content of the graph in the toolbar and artifact info box. For details, see "Navigator Toolbar" on page 83 and "Navigator Information Boxes" on page 85.

Navigator Filters

The Navigator provides a set of filters enabling you to focus on the artifacts that matter to your role. The filter determines the content of the graph by defining which artifacts display, when to group a set of artifacts, which relationships to show, and what color each relationship uses.

Access Navigator Filters, using the Filters toolbar control.

Screenshot: Navigator Filters Control
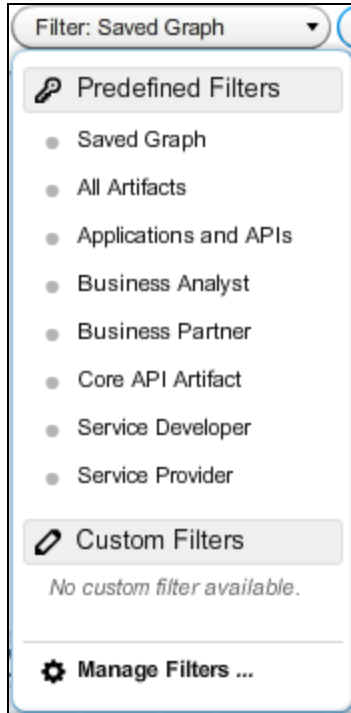
Select from the following predefined filters which the administrator can edit:

- **All Artifacts**

  Displays all artifacts that your role has permission to view.

- **Applications and APIs**

  Focuses on applications and application interfaces related artifacts.

- **Business Analyst**

  Focuses on applications, projects, services, processes, and their implementations, and contract related artifacts.

- **Business Partner**

  Focuses on services and their implementations, and contract related artifacts.

- **Core API Artifact**

  Focuses on core API artifacts.

- **Service Developer**

Focuses on the implementation of services, showing services, their implementations, and service description documents.

- **Service Provider**

  Focuses on the provision of services, showing services, their implementations, endpoints, and operations.

> **Note:** The choice of filters is restricted to those appropriate to your role.

In addition to the predefined filters, you can create your own custom filters. For details about creating, copying, and editing filters, see "How to Manage Navigator Filters" on page 86.

Navigator Layouts

The Navigator offers a number of alternative layouts providing different ways to view Catalog data, enabling you to visualize your data in the way that best suits your needs.

Access Navigator Layouts, using the Layouts toolbar control.

Screenshot: Navigator Layouts Control
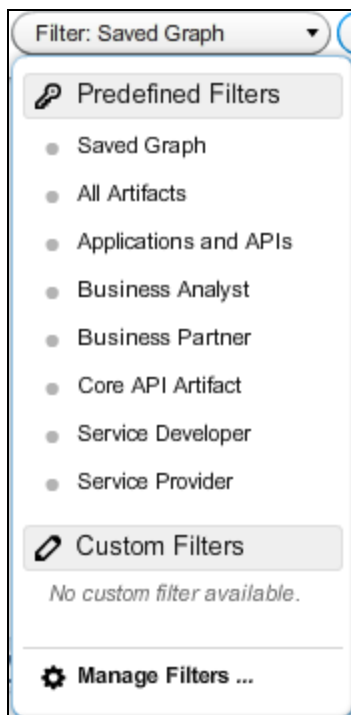
Select from the following predefined layouts:

- **Concentric Radial** - The default graph layout places the current artifact in the center of the graph and the related artifacts radiate outwards in all directions. This layout enables you to explore the immediate neighborhood of an artifact.

- **Hierarchical Layout** - This layout organizes the graph into a vertical artifact hierarchy according to the artifact relationships and any defined artifact aggregation structures.

The administrator can configure aggregations and there are a set of default aggregations in Systinet. For details about Aggregation, see "Aggregation" on page 154.
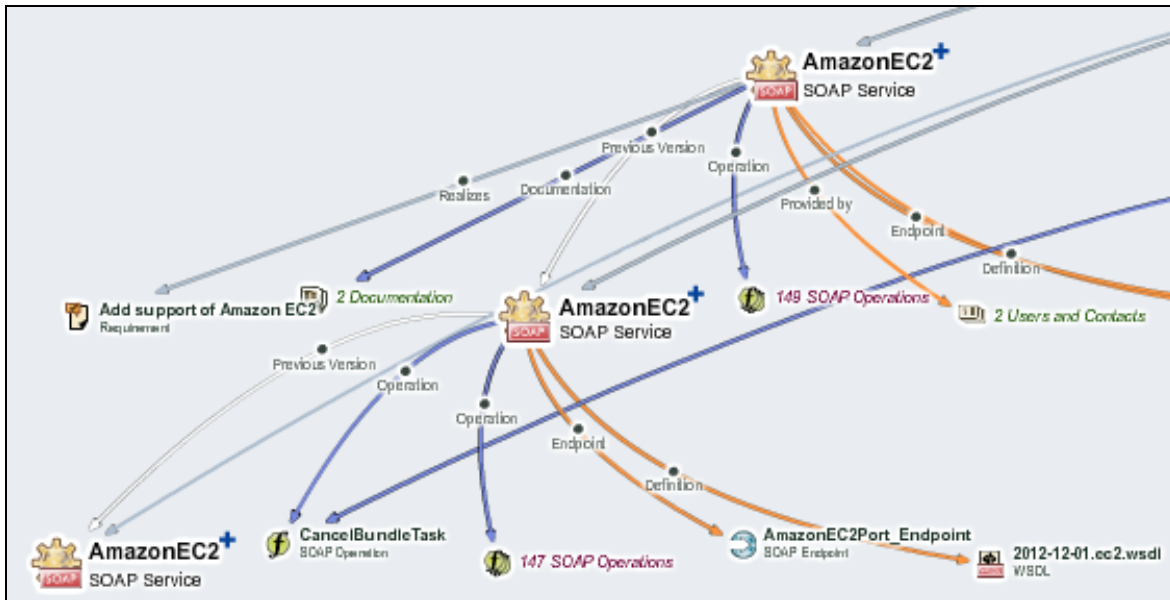
Layered Layouts

Layered layouts organize the hierarchy of artifacts into identified layers, enabling you to focus on important abstractions and analyze dependencies between layers. Each expandable layer can consist of multiple artifact types with relationships within a layer and between them.

Systinet provides the following default Layered Layouts:

- **API Implementation Layer** - provides the implementation layer of the APIs. It divides artifacts into Implementation, Interfaces, Request and Response layers.
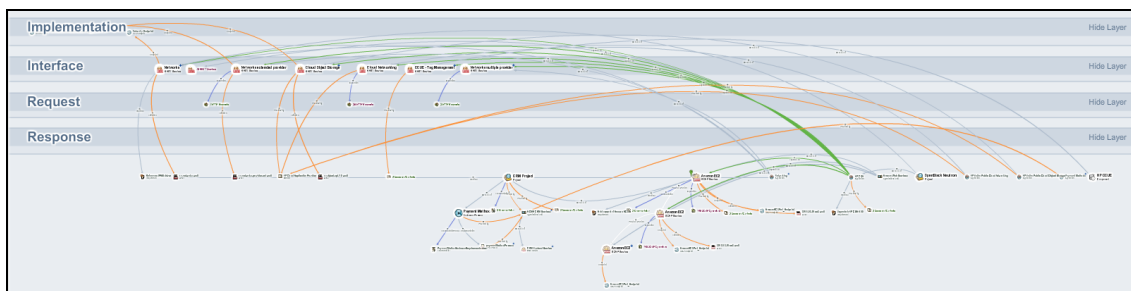
- **Application and APIs Layers** - provides the view of applications and API layers. It divides artifacts into Application and API layer.
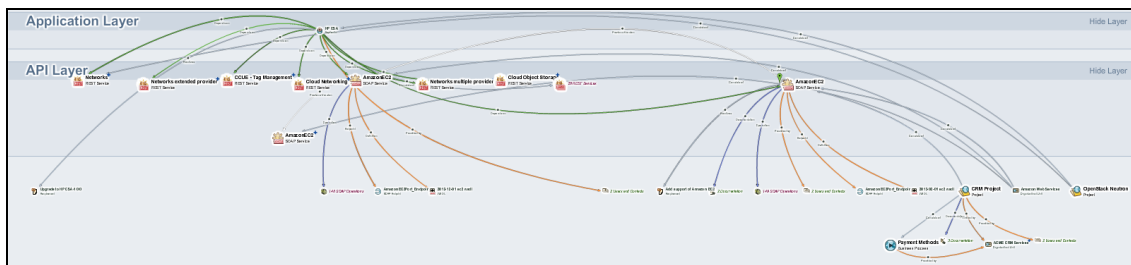
- **Architectural Layers**- provides a holistic view of your SOA architecture. It divides artifacts into Application, Process, Service, and Infrastructure Layers.
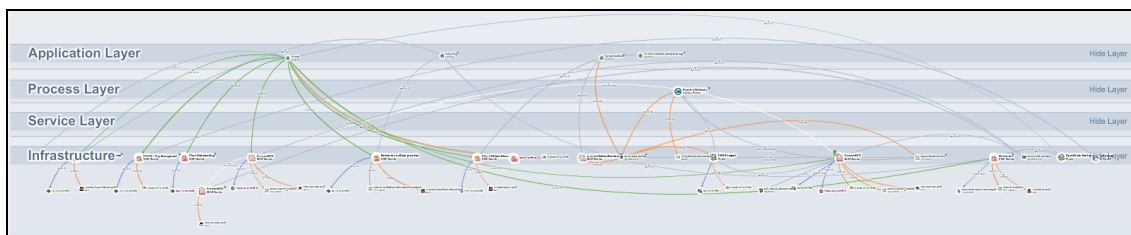
- **Implementation Layers** - renders the internal architecture of services and their implementations. It divides artifacts into Services & Process, Implementation, WSDLs, and XML Schemas layers.
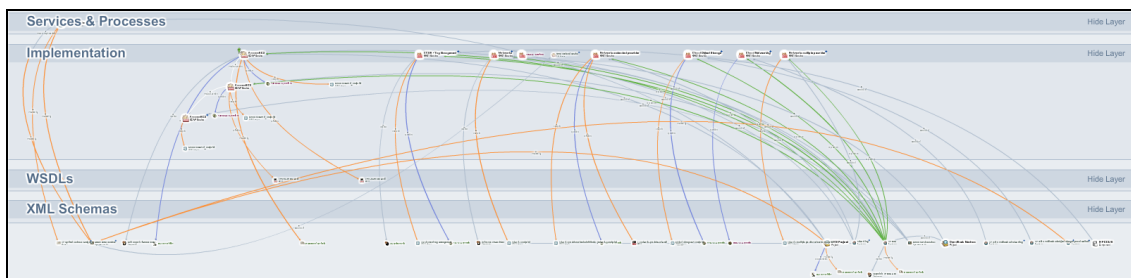
- **Project View** - provides a view tailored to projects and their content. It enables you to analyze the dependencies between projects, their content, and used data types.
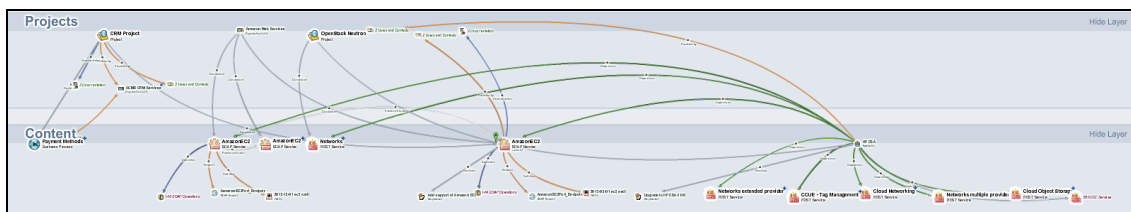
Systinet enables you to create your own customized layered layouts. For details about managing custom layouts, see "How to Manage Navigator Layouts" on page 88.

In addition to the various layout schemes, you can highlight the impacts between artifacts and artifacts in specific domains. For details, see "How to Highlight Impact and Dependency" on page 93 and "How to Highlight Domains" on page 93.

Navigator Tab

The main purpose of the Navigator tab is to provide an overview of an artifact and its relationships.

To View a Navigator Graph for an Artifact:

- Follow the instructions given below:

    - In the Artifact Details page Overview or Details tab, click the **Navigator** context action. The Navigator tab opens showing the graph for that artifact.

    - Click the **Navigator** tab and click ⌂ to open the Select Root Artifact dialog box.

        > **Note:** If you have not opened the Navigator previously, the Select Root Artifact dialog opens by default. The default behavior of the Navigator tab is to open the previously viewed graph.

        Use the Search, Saved Graphs, Favorites or Recent Artifacts tabs to locate the artifact or graph you want to view. For more details, see "How to Use Navigator Search" on page 90, "How to Use Saved Graphs" on page 91, "How to Access Favorites in Navigator" on page 91, and "How to Access Recent Artifacts in Navigator" on page 92.

        The Navigator tab opens showing the graph for the selected artifact.

Screenshot: Navigator Tab
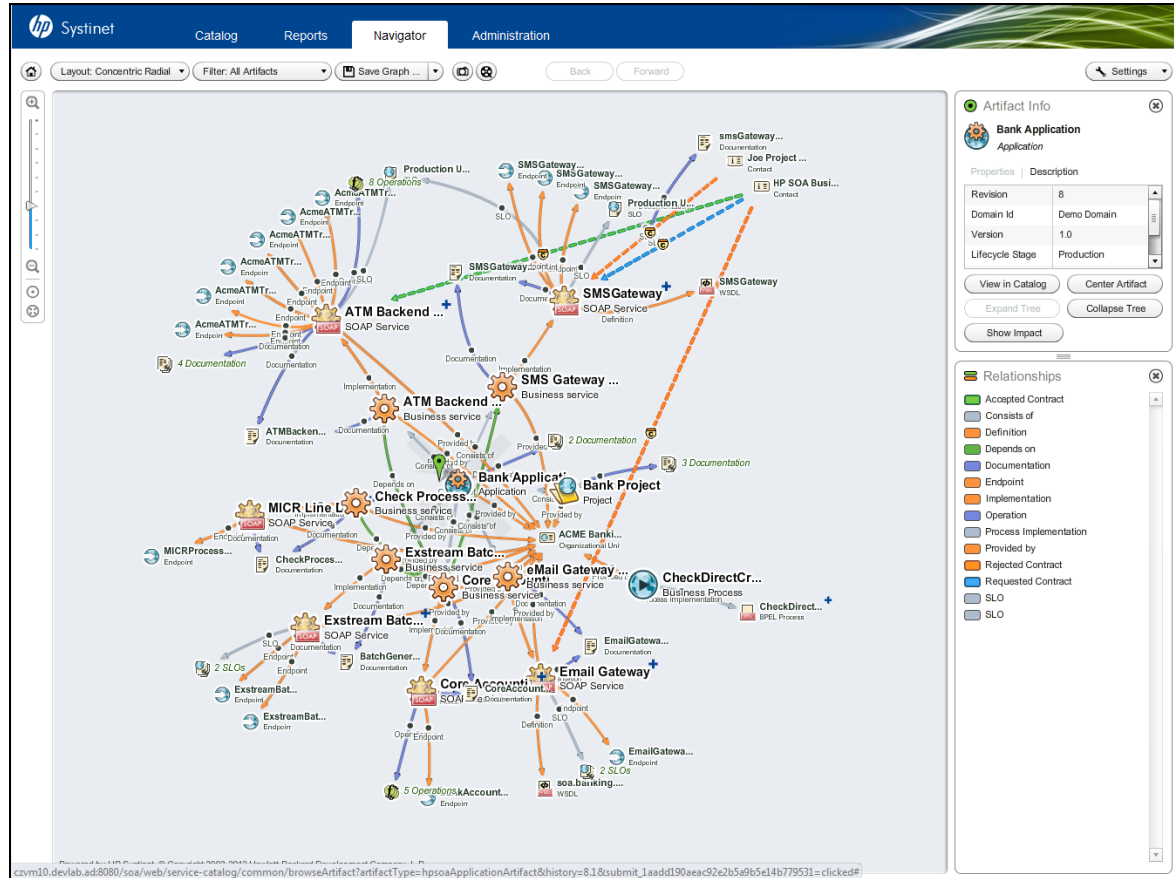
The Navigator Tab consists of the following elements described in more detail in the following topics:

- "Understanding the Navigator Graph" on page 74

- "Navigator Toolbar" below

- "Navigator Information Boxes" on page 85

Navigator Toolbar

The Navigator toolbar contains the following functionality:

Navigator Toolbar Functions

Function	Description
Zoom	Use slider or the **[+]** or **[-]** controls to the left of the graph to change the size. The buttons below the slider enable you to reset the graph to its default layout and fit the whole graph on the screen.

Navigator Toolbar Functions, continued

Function	Description
⌂	Open the Select Root Artifact dialog and use the Search, Saved Graphs, Favorites, or Recent Artifacts functions to select a new artifact to view in Navigator. For details, see "How to Use Navigator Search" on page 90, "How to Use Saved Graphs" on page 91, "How to Access Favorites in Navigator" on page 91, and "How to Access Recent Artifacts in Navigator" on page 92.
Layout	Change the layout of the graph, selecting from the default layouts or your own custom layouts. For details, see "Navigator Layouts" on page 78.
Filter	Change the graph filter, selecting from the default filters or your own custom filters. For details, see "Navigator Filters" on page 76.
Save Graph	Add the current graph layout to your saved graphs or select from previously saved graphs. For details, see "How to Use Saved Graphs" on page 91.
(camera icon)	Create and save an image file for the current graph.
(full-screen icon)	Open a full-screen view of the current graph. Click the button again or press **Esc** to return to the default navigator view. **Note:** According to Adobe Flex full-screen security policy, users cannot enter text in text input fields while in full-screen mode. All keyboard input and the keyboard-related ActionScript is disabled while in full-screen mode, with the exception of the keyboard shortcuts (such as pressing the Esc key) that return the application to normal mode. The navigator automatically exits full-screen mode for most user-dialogs where user input may be required, except for the Artifacts to Expand dialog.
Back Forward	When re-centering or re-laying out a chart, the current graph is saved and a user can to go backwards and forwards to view the versions.
Settings	• Select to display the Artifact Info and Relationships. For more details, see "Navigator Information Boxes" on the next page. • Select to display the **Artifact Type**, **Lifecycle Stage**, or **Version** as the artifact node annotation. • Select whether to display relationships annotation.

Navigator Information Boxes

On the right of the graph there are a number of information boxes displaying information about the currently selected artifact, the relationships, and optional highlighting information for domains and impact.

Screenshot: Navigator Information Box
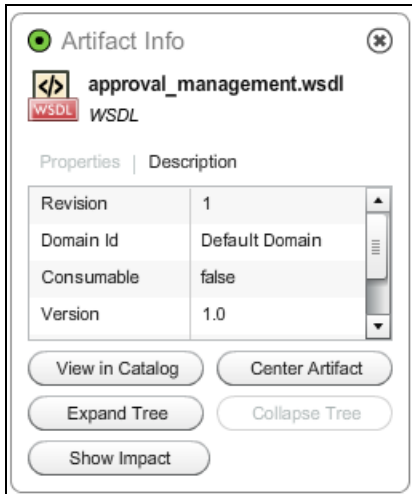
The Artifact Info box displays information about the currently selected node highlighted with the green flag. Depending on the node there are the following types of Information box:

- **Artifact Info**

  For a single artifact, the Information box shows some artifact details, with the following options:

  - **View in Catalog**

    Exit the Navigator and view the artifact details in the Systinet UI.

  - **Center Artifact**

    Make the artifact the focus of the graph.

  - **Expand Tree**

    For collapsed nodes, expand the relationships to hidden artifacts.

  - **Collapse Tree**

    Collapse outward relationships from this artifact.

  - **Show Impact**

Highlight the impact tree for the artifact. For details, see "How to Highlight Impact and Dependency" on page 93.

- **Artifact Group Info**

  For a group of artifacts artifact, the Information box displays the following options:

  - **Expand All**

    Expand all the artifacts in a group.

  - **Expand**

    Open the Expand Artifacts from Group dialog box to select the artifacts to view separately from the group.

Screenshot: Relationships Legend
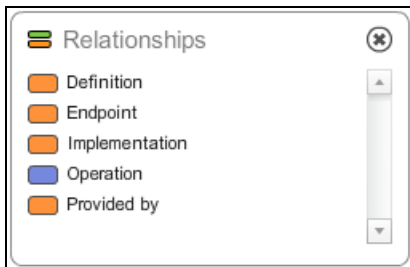
The Relationships information box displays the color coding identifying the relationships in the graph. You can customize the relationship colors by creating your own filter. For details, see "Navigator Filters" on page 76.

The Impact and Domain information boxes display when you select to highlight these aspects of the graph. For more details, see "How to Highlight Impact and Dependency" on page 93 and "How to Highlight Domains" on page 93.

How to Manage Navigator Filters

You can create and manage your own set of custom navigator filters. For details about the concept of filters, see "Navigator Filters" on page 76.

To Create, Copy, or Edit Navigator Filters:

1. Expand the Filter toolbar control, and select **Manage Filters** to open the Manage Filters dialog box.

2. Do one of the following:

- Click **New** to open the Create New Filter dialog box.

  The default settings are the set of all artifacts visible to you.

  > **Note:** If you want to share a created filter that all users can see, select the filter and click **Share** in the Manage Filters dialog. Click **Yes** to confirm you want to share the filter. Click **Unshare** to hide the shared filter from other users.

- Select the filter to edit, and click **Edit** to open the Edit Filter dialog.

  > **Note:** For built-in filters, this option is only available to the administrator. Systinet prompts other users to copy the selected filter.

- Select the filter to copy, and click **Copy** to open the Copy Filter dialog.

  The default settings match the selected filter.

Screenshot: Create New Filter Dialog Box

3. Input a name for your filter if you are creating a new filter or copying a filter.

4. In the **Artifact Types** tab do any of the following:

- Select the artifact types to display using the individual artifact and artifact collection checkboxes.

- Use the **Grouping** slider bar for each selected artifact type to determine how the navigator displays multiple instances of an artifact type. To show all artifacts of that type which branch from the same related artifact node as individual nodes, slide the bar all the way to the right (if there are 20 or more individual artifact instances branching from the same related artifact node, the navigator always displays them as an expandable group). To always display multiple instances of the artifact type branching from the same related artifact node as a group, forcing the user to expand it to see individual artifact nodes, slide the bar all the way to the left.

- Use **Filter** to search for a particular artifact type.

5. In the **Relationships** tab, select the relationships to display, the direction of the relationship (Incoming, Outgoing or both) and select a color for each relationship type. Use **Filter** to search for a particular relationship type. The available relationships are determined by the selected artifact types.

6. If you want to share a filter that all users can see, select the filter and click **Share** in the Manage Filters dialog. Click **Yes** to confirm you want to share the filter. Click **Unshare** at any time to make the shared filter hidden from other users.

7. Click **Save** to save your filter.

To delete a custom navigator filter, select it and then click **Delete**.

How to Manage Navigator Layouts

Systinet enables you to create and manage your own set of custom navigator layered layouts. For details about the concept of layouts, see "Navigator Layouts" on page 78.

> **Note:** You cannot edit the concentric radial and hierarchical layouts.

To Create, Copy, or Edit Navigator Layouts:

1. Expand the Layout toolbar control, and select **Manage Layouts** to open the Manage Layouts dialog box.

2. Do one of the following:

   - Click **New** to open the Edit Layout dialog box to create a new layout.

     > **Note:** If you want to share a created layout that all users can see, select the new layout

and click **Share** in the Manage Layouts dialog. Click **Yes** to confirm you want to share the filter. Click **Unshare** to hide the shared layout from other users.

- Select the layout to edit, and click **Edit** to open the Edit Layout dialog.

  Note: For built-in layouts, this option is only available to the administrator. Systinet prompts other users to copy the selected layout.

- Select the layout to copy, and click **Copy** to create a copy of the selected layout. Select the copy and click **Edit** to open the Edit Layout dialog box.

Screenshot: Edit Layout Dialog Box

3. Control the number of layers with the **Add Layer** button and the **Delete Layer** control in each layer.

4. To change the order of layers, use the up and down buttons in each layer.

5. To edit layer names click **Rename Layer**, input a new name and click **Save**.

6. Select a layer to control the artifact types in each layer:

   - Click **Add Rule** to open the Select Artifact dialog.

     Use the text filter to locate a particular artifact type, select an artifact type, and click **Select** to

add it to the layer.

- Click **X** in the right-column of the artifact table to remove an artifact type from the layer.

7. Click **Save** to make your changes.

To delete a custom navigator layout, select it and then click **Delete**.

How to Use Navigator Search

The Navigator enables you to search for artifacts you want to view.

Click ⌂ to open the Select Root Artifact dialog box and select the Search tab.

Screenshot: Navigator Search Dialog
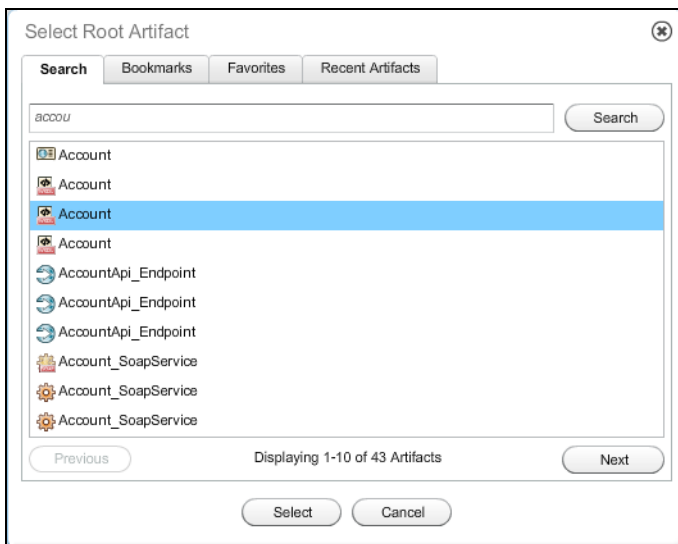
To Perform a Navigator Search:

1. Start typing your search term in the input box. As you type Systinet updates the dialog with the artifacts matching your search term.

   Note: Search uses the name, description, and keyword properties to find matching artifacts.

2. Select an artifact and click **Select** or double-click an artifact to open its graph. Use **Next** and **Previous** to locate the artifact if there are too many to show in the dialog.

How to Use Saved Graphs

The Navigator enables you to save particular graph layouts for later reference.

To Create Saved Graphs:

1. Do one of the following:

   ■ Click **Save Graph**.

   ■ Expand the Save Graph menu, select **Manage Saved Graphs** to open the Manage Saved Graphs dialog box and click **Save**.

2. Enter a name, and click **Save** to create a saved graph for the current layout.

To Access Saved Graphs:

- Do one of the following:

   ■ Expand the **Save Graph** menu and select a saved graph from the list to open the saved graph layout.

   ■ Expand the **Save Graph** menu and select **Manage Saved Graphs** to open the Manage Saved Graphs dialog box. Select a saved graph and click **Load** to open the saved graph layout.

   ■ Click ⌂ to open the Select Root Artifact dialog box and select the Saved Graphs tab. Select a saved graph and click **Select** or double-click a saved graph to open the saved graph layout.

How to Access Favorites in Navigator

The Navigator enables you to quickly access the graphs for artifacts you have marked as favorites.

Click ⌂ to open the Select Root Artifact dialog box and select the Favorites tab.

Screenshot: Navigator Favorites Dialog
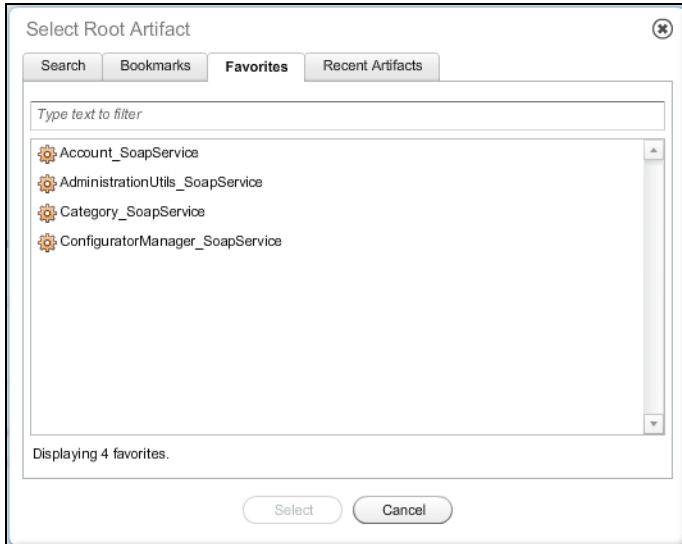
Select an artifact and click **Select** or double-click an artifact in the Favorites dialog to open the graph for that artifact.

How to Access Recent Artifacts in Navigator

The Navigator enables you to quickly access the artifacts you have recently viewed in the Catalog.

Click ⌂ to open the Select Root Artifact dialog and select the Recent Artifacts tab.

Screenshot: Navigator Recent Artifacts Dialog
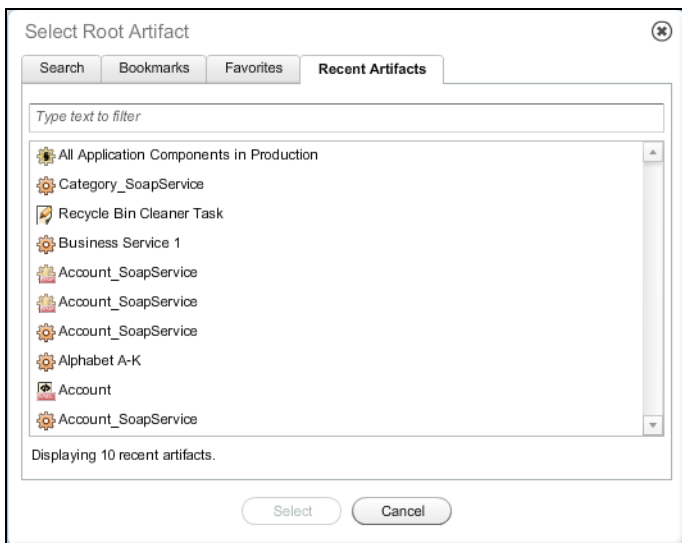
Select an artifact and click **Select** or double-click an artifact in the Recent Artifacts dialog to open the graph for that artifact.

How to Highlight Impact and Dependency

To assist you in assessing the effect of change, the navigator enables you to highlight the impact and dependencies between artifacts in the context of all the displayed relationships.

To View the Impact of an Artifact:

- Do one of the following:

    - In the Navigator graph, open the context menu for the artifact node and select **Impact Highlight**.

    - In the Artifact Info box, click **Show Impact**.

The artifact is highlighted with an orange flag and an Impact legend box opens displaying the selected artifact. All related artifacts that are potentially impacted by a change to this artifact are highlighted, and all other artifacts are dimmed.

Remove the impact highlighting by closing the Impact legend box, or open the context menu for the highlighted artifact and select **Impact Highlight**.

> **Note:** Systinet determines the impact and dependency on related artifacts using the default aggregate structures and any custom aggregates defined by the administrator, For details, see "Aggregation" on page 154.

How to Highlight Domains

The navigator enables you to highlight artifacts in a particular domain in the context of all the displayed relationships.

To View the Artifacts Within a Domain:

1. Open the context menu for graph and select **Domain Highlight** to open the Domain Highlight information box.

2. Select the domain to highlight from the drop-down list.

All artifacts in the selected domain are highlighted and all other artifacts are dimmed.

Remove the domain highlighting by closing the Domain Highlight information box, or open the context menu for the graph and select **Domain Highlight**.

Chapter 4: Collaboration

This part contains the following topics:

Concepts

- "Collaboration Overview" below

Tasks

- "How to Use Comments" on the next page

- "How to Use Notifications" on page 96

- "How to Use Events" on page 98

- "How to Use Tasks" on page 100

- "How to Use Ratings" on page 101

- "How to Use Feeds" on page 101

- "How to Share Artifacts" on page 102

Collaboration Overview

Systinet is a collaborative platform enabling you to actively participate and provide information to other users throughout the service lifecycle.

Systinet provides the following collaboration features:

- **Comments**

  The Artifact Details page provides a Discussion area where you can start threads and respond to other comments. For details, see "How to Use Comments" on the next page.

- **Notifications**

  You can send e-mail notifications about an artifact to specified stakeholders. For details, see "How to Use Notifications" on page 96.

- **Events**

  Events keep you up-to-date with changes to artifacts that you are a stakeholder in and user actions that may impact you. For details, see "How to Use Events" on page 98.

- **Tasks**

  Tasks provide you with updates for lifecycle and contract tasks assigned to you. For details, see "How to Use Tasks" on page 100.

- **Ratings**

  You can apply an individual rating to an artifact, contributing to an overall rating that enables other users to select the best artifacts. For details, see "How to Use Ratings" on page 101.

- **Feeds**

  Systinet provides artifact and search feeds that enable you to track changes and discussions about artifacts in your feed readers and in the Reports tab. For details, see "How to Use Feeds" on page 101.

- **Sharing**

  Systinet provides a simple action to provide all users with read access to an artifact, making it visible across the whole user-base. For details, see "How to Share Artifacts" on page 102.

Systinet is also a collaborative platform in combination with your development environments. HP Software provides plug-ins for widely used IDEs that enable your developers to collaborate using the discovery and publishing features of Systinet directly from their IDE. For details, see *Plugin for Eclipse* and *Plugin for Visual Studio*.

How to Use Comments

Systinet provides an artifact discussion forum where you can make comments about particular artifacts.

The Artifact Detail page Overview tab includes a Discussion component showing the last few comments.

Screenshot: Artifact Detail Discussion Component
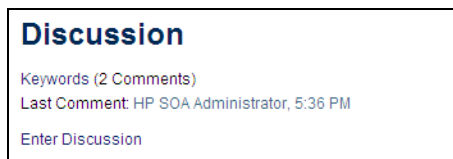
Click a thread title to enter that discussion or click **Enter Discussion** to open the Artifact Details Discussion tab.

You can browse all comments in the Artifact Details Discussion tab.

Screenshot: Artifact Detail Discussion Tab
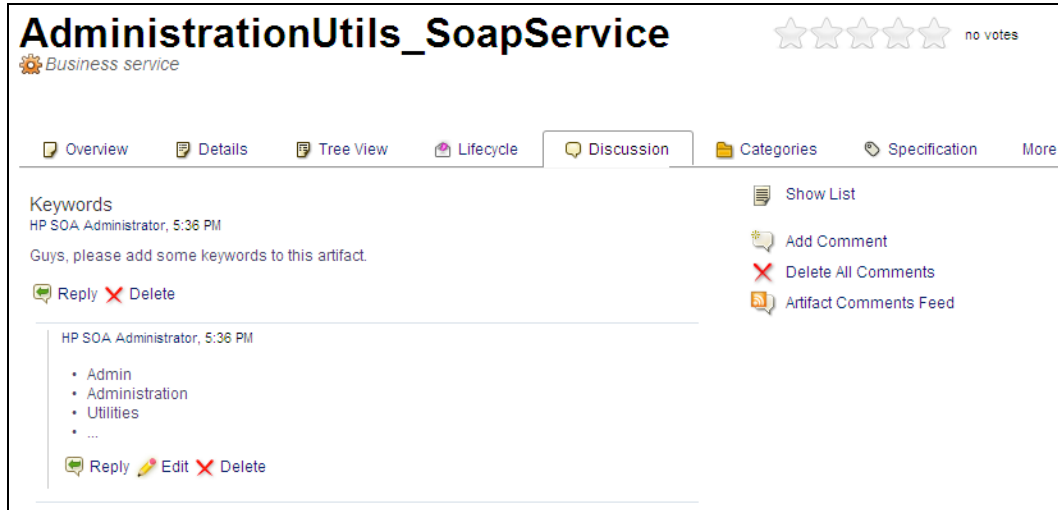
You can organize the Artifact Details Discussion tab using **Show Thread** and **Show List** with an option to sort the list view by oldest or newest comment first.

Start a new discussion thread using **Add Comment**.

> **Tip:** In the Add Comment dialog, click ⊞ to provide a reference link to another artifact in your comment.

Respond to a particular comment using by clicking **Reply** for that comment.

If these are no replies to your thread or comment, you can **Edit** and **Delete** your comments.

The Artifact Comments Feed context action provides a URL that you can use in feed readers or in the Reports Tab to keep up-to-date with any new comments about the artifact. For details, see "How to Use Feeds" on page 101.

How to Use Notifications

Systinet enables you to send e-mail notifications to artifact stakeholders.

To Send Manual Notifications:

1. In the Artifact Details page Overview or Details tab for the artifact you want to send the notification for, click the **Send Message** context action to open the Send a Message page

2. Select the recipients to notify.

Notification Recipients

Recipient	Description
Owner	The user, group, or role that owns the artifact. For details, see "How to Edit Artifact Ownership" on page 179.
Maintainers	Users, groups, and roles groups with write permission for the artifact. For details, see "How to Edit Access Rights" on page 185.
Contacts	Users and organizational units associated with the artifact by the contact relationship or listed in the artifact stakeholder property. For details, see "How to Manage Contacts" on page 119. The Contacts recipient group can be refined into selectable contact roles.
Consumers	People and organizational units with contracts to use the artifact. For details, see "Contracts Overview" on page 223.
Providers	People and organizational units responsible for contracts associated with the artifacts. For details, see "Contracts Overview" on page 223.
Other Recipients	Click **Add Other Recipients** and use the user, group, and role search feature to add any other required recipients.
Previous Stakeholders	Expand **Show Advanced Options** and select **Include Recipients from Previous Versions** to notify stakeholders from previous versions of the artifact about changes to a newer version. The stakeholders notified by this option match those for the current artifact version. For example, if **Consumers** is selected, then the consumers of previous versions are notified.

> **Note:** The recipient options vary depending on the status of the artifact.

3. *Optional* Change the subject for the mail.

4. *Optional* Select **Lifecycle Status** include a section containing the current lifecycle status in the mail.

5. Enter the message text for the mail.

6. Click **Preview** to view a draft of the mail message.

7. Click **Send** to send the notification to the selected recipients.

The administrator can set up automatic notifications as part of a lifecycle process. For details, see "How to Define Automatic Actions" in the *Administration Guide*.

How to Use Events

Systinet keeps you up-to-date with all actions on artifacts that are relevant to you. Actions may appear in the Events lists because you performed them, or because they affect artifacts that you are a stakeholder for.

Events information appears in the following places in the UI:

- New Events notification in the Profile Status section of all pages.

- Actions on individual artifacts displayed in the Artifact Details page Events component and the Artifact Details page Events tab.

Screenshot: Artifact Overview Events Components
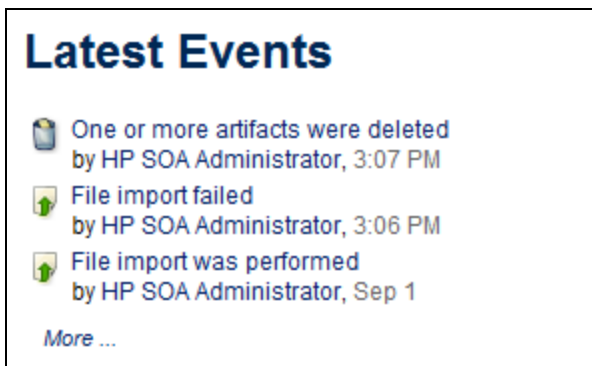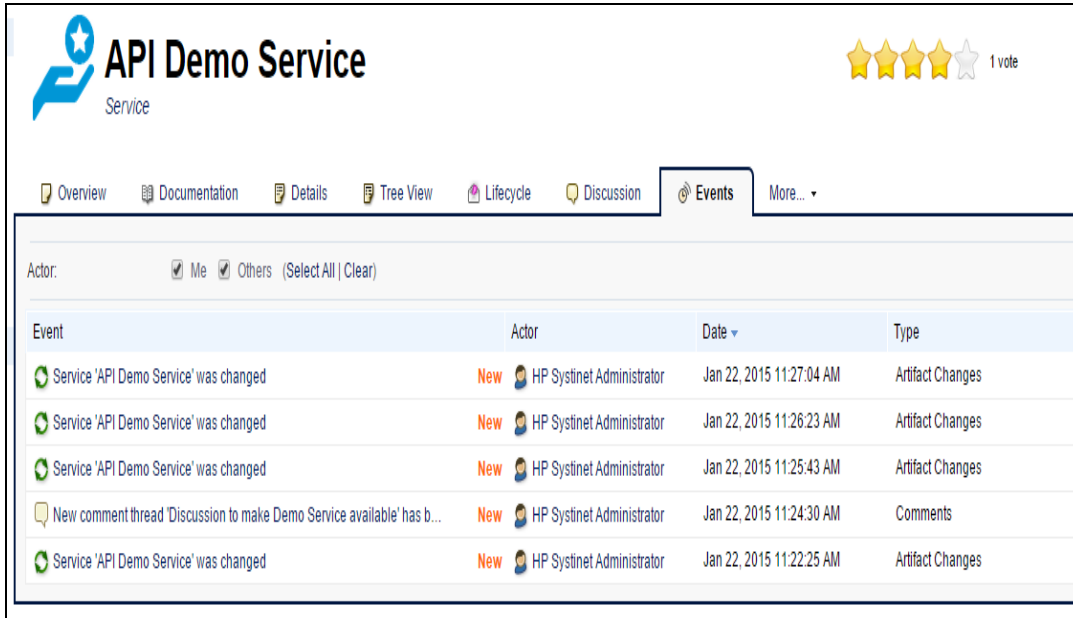
The Events components in the Artifact Detail page Overview tab display the latest actions performed in Systinet that are relevant to you.

Screenshot: Events Page
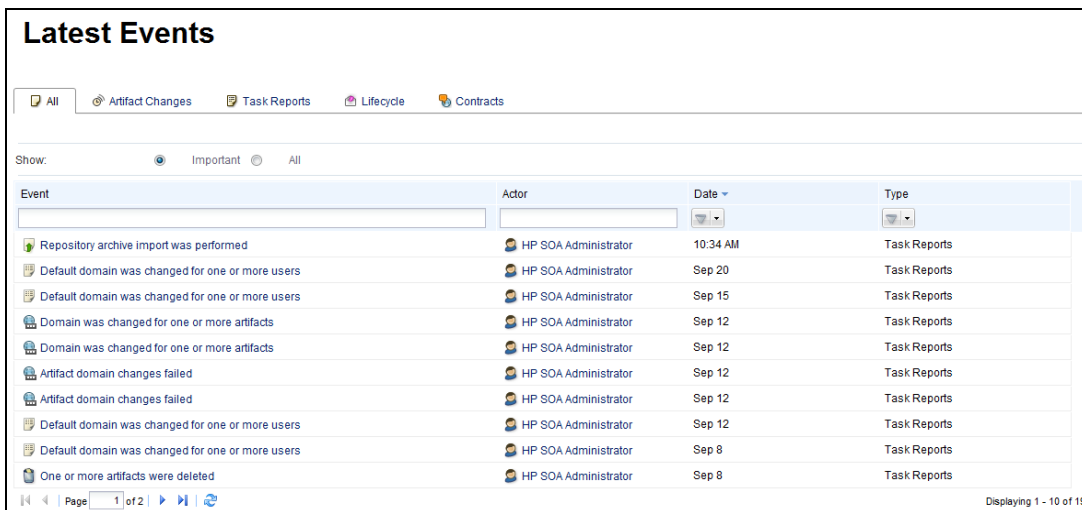
The Artifact Overview Events component displays actions performed on the artifact with a **More...** link to the Artifact Events tab.

Screenshot: Artifact Details Events Tab

You can customize the information the Events component displays.

To Customize the Events Component:

1. Click the tool icon in the Events component on the home page to open the Edit Configuration - Events dialog box.

2. Customize the following events parameters:

3. Click **Save** to apply your changes.
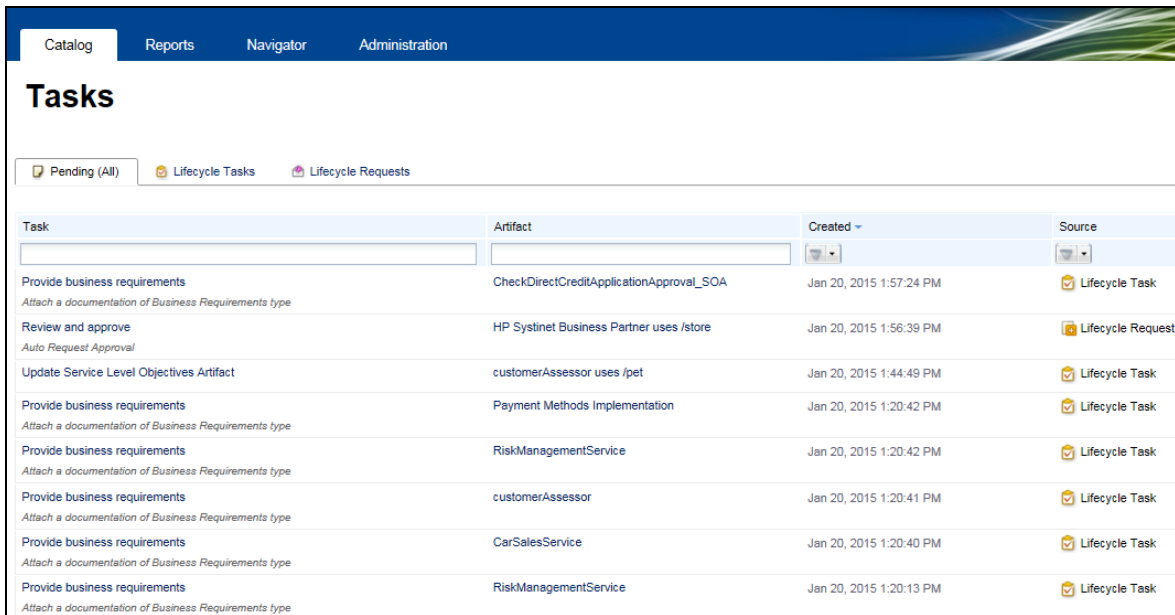
How to Use Tasks

As a user in Systinet you may be required to perform actions related to contracts and the lifecycle of artifacts. Tasks may be assigned to you as an individual, a member of a group, as a user in a particular role, or because you are the owner of an artifact.

Task assignments come from lifecycle or as a result of a contract request. Click the task name to open the relevant page to view the lifecycle or contract action.

To view all your outstanding tasks click **Tasks** in the left menu of the homepage to open the Tasks page.

Screenshot: Tasks Page

The Tasks page shows more details about each tasks and enables you to sort them by type or filter to find a particular task or artifact.

In addition to viewing all tasks assigned to you, Artifact Details pages contains task information specific to the artifact in the My Tasks component.

Screenshot: Artifact Details Page Overview Tab My Tasks Component
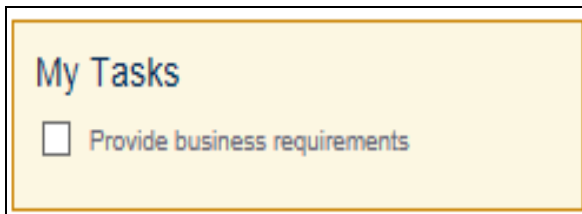
The My Tasks component enables you to complete a lifecycle task.

For all these task displays, clicking the task opens the appropriate page to perform the required actions depending on the task type.

- **Lifecycle Tasks**

  Perform a task assigned to you, your group, or your role as part of a lifecycle process. For details, see "How to Review/Close Lifecycle Tasks" on page 248.

- **Lifecycle Requests**

  Process requests for approval of the completion of a lifecycle stage for artifacts. For details, see "How to Approve/Reject Approval Requests" on page 251.

How to Use Ratings

Systinet enables you to apply a rating score to artifacts. You can apply a rating to each artifact once and change your rating at any time.

The average rating is available in Artifact Detail pages and in tables of artifacts, enabling you to choose services based on their collective rating.

Screenshot: Service Rating

To Rate an Artifact:

1. Open the Artifact Details page.

2. Move the cursor over the stars and click a star to set your rating. The status next to the rating stars indicates the current status and indicates if you have previously rated the service.

Average ratings are available to filter and sort tables of artifacts. For details, see "Tables in the User Interface" on page 24.

How to Use Feeds

Systinet provides Atom feed support to enable you to track changes either with your own feed reader or using the Content Feed feature of the Reports Tab.

You can access the following feeds:

- **Individual Artifact Feeds**

  Track changes to a specific artifact. Use the **Feeds>Atom Feed** context action from the Artifact Details page Overview or Details tabs to obtain the URL for the feed.

- **Individual Artifact Comment Feeds**

  Track the discussion about a particular artifact. Use the **Feeds>Artifact Comments Feed** context action from the Artifact Details page Overview or Details tabs or the **Artifact Comments Feed** context action from the Artifact Details page Discussion tab to obtain the URL for the feed.

- **Search Result Feeds**

  Use the Search URL provided with saved searches to create a custom search feed. A typical example would be to create a search feed for all services. This feed would then notify you about the creation of new services. For details about Saved Searches, see "How to Use Saved Searches" on page 64.

For details about using Content Feeds in the Reports tab to track changes, see "How to Manage Content Feeds" on page 357.

How to Share Artifacts

By default, the separation of the Catalog into domains and default security mean that artifacts are only visible to users who need to see them. However, there are situations where you want to make an artifact visible to all users, for example, when a service goes into Production.

Systinet provides a simple share mechanism which enables the administrator or the owner of an artifact gives all users read access to an artifact, making it visible to everyone. Share may be performed on an individual artifact or in bulk.

> **Note:** The default functionality of sharing uses the `system#registered` group which represents all users who access Systinet. The administrator can change the users that can access shared artifacts. For details, see "How to Change the Sharing Principal" in the *Administration Guide*.

To Share an Artifact:

1. Open the Artifact Details page for the artifact and select the Access Rights tab.

2. Click the **Share** context action to open the Share dialog box.

3. *Optional*: Expand **Advanced Options** and select **Include Subartifacts** to also share sub-artifacts in the artifact aggregate.

4. Click **Share** to make the artifact visible to all users.

To Share a Set of Artifacts:

1. From a table of artifacts, select the artifacts to change, open the **Edit** context menu, and select **Share** to open the Share dialog box.

2. *Optional*: Expand **Advanced Options** and select **Include Subartifacts** to also share sub-artifacts in the artifact aggregates.

3. Click **Share** to make the selected artifacts visible to all users.

Chapter 5: Authoring

This part contains the following topics:

Concepts

Tasks

Reference

Authoring Overview

Authoring is the process of adding content to the Catalog for the purposes of governance and management.

Authoring consists of the following overlapping areas of functionality:

- Manually creating artifacts to represent entities that you want to develop, govern, and publish.

- Manually enhancing these artifacts with additional properties that represent important information about these entities.

- Manually organizing the content by creating relationships between artifacts.

- Publishing data artifacts with attached content. The publishing process may automate some of the manual authoring steps.

This topic describes the following authoring concepts:

- "Application Modeling" below

Different organizations model their application infrastructure in different ways. This topic provides a concept reference to the various *service artifacts* in Systinet and ways to use them to meet your requirements.

- "Environments" on page 111

Most organizations deploy services and applications to multiple environments during their lifecycles. This topic describes the concept and implementation of environments in Systinet.

- "Publishing Overview" on page 112

Systinet provides extensive support for publishing service definition documents, such as WSDLs. This topic describes the various options available for streamlining your content publishing.

- "Service Provision Workflow" on page 116

This topic describes the basic steps involved in taking a service from a design document, to a service in production and ready for consumers to use.

The concepts described above involve using the following authoring procedures:

In addition to these authoring processes, "Artifact Management Overview" on page 153 describes additional processes that you may need as part of your overall authoring process.

HP Software also provides a set of IDE plugins with integrated publishing functionality. For details, see *Plugin for Eclipse* and *Plugin for Visual Studio*

Application Modeling

Different organizations model their application infrastructure in different ways. The open architectural model of Systinet enables you to choose how you create, govern, and provide your application infrastructure in a way that matches your needs.

There are a number of different definitions for *service artifacts* that recognize the various ways that different organizations describe their service infrastructure. Systinet provides the following artifact types with suggestions of how you may want to use them:

Service

A service is a set of functionality that you can provide or use as a single entity. The purpose of the service artifact is to provide an abstraction in Systinet that you can use as a central point of governance, provision, and use that describes the functionality and the users, groups, and contacts responsible for its maintenance and support. For more details, see "Service Artifacts" on the next page.

Business Process

A business process is a set of functionality that you can provide or use as a single entity. The purpose of the process artifact is to provide an abstraction in Systinet that you can use as a central point of governance, provision, and use that describes the functionality and the users, groups, and contacts responsible for its maintenance and support. For more details, see "Business Process Artifacts" on page 108.

Application Interface

An implementation represents the technical and functional aspects of a service or business process. Each service or process may have a number of implementations to different endpoints or with different versions. The implementation artifact provides a central reference point for all these technical and functional aspects of the service or process. An implementation typically consists of an endpoint that the implementation is deployed to, the operations that the implementation performs, and a set of definitions documents that describe the service in technical terms. The elements of an implementation are described in more detail in "Implementations" on page 108.

Application

An application typically provides a set of services and processes. The application artifact enables you to collect a set of services and processes together to govern and provide them as a single entity. Use the generic add relationship functionality to add artifacts to an application. For details, see "How to Manage Relationships" on page 120.

Project

A project typically contains a set of artifacts with a shared development lifecycle. The project artifact enables you to collect these artifacts together and govern them as a single entity. Use the generic add relationship functionality to add content to a project or use specific project functionality. For details, see "How to Manage Relationships" on page 120 and "How to Use Projects" on page 127.

Using these artifact types and the relationships between them provides a platform to manage, govern, and provide your content at the most appropriate level for your organization. For more details about relationships, see "Relationships" on page 110.

An important concept for artifact management is aggregation. Aggregation enables you to perform actions on one artifact which then propagate to all the related sub-artifacts. For more details, see "Aggregation" on page 154.

Consider the following factors when you choose the most appropriate artifact types to govern and provide your infrastructure:

Provision and Use

For the purpose of forming contracts, the model contains an Service Level Objectives artifact type to represent the terms that you offer for a piece of functionality. The default functionality of Systinet

enables an SLO to be associated with Applications, Services, Implementations, Endpoints, and Operations.

How you want to offer your business functionality to consumers may determine your choices when you create, govern, and provide your service infrastructure. For more details, see "Contracts Overview" on page 223.

Lifecycle Governance

For the purpose of governing a lifecycle, Systinet provides default lifecycle processes for Applications, Services, Implementations, Projects, Processes, and Process Implementations.

How you want to govern the lifecycle of your business functionality may determine your choices when you create, govern, and provide your service infrastructure. For more details, see "Lifecycle Overview" on page 238.

Versioning

During their lifetime services and processes evolve, so there may be different versions that require management. Systinet supports various versioning strategies enabling you to introduce new versions in a controlled way, which you can then manage, govern, and provide separately.

How you want to support versioning may determine your choices when you create, govern, and provide your service infrastructure. For more details, see "Versioning Overview" on page 213.

Service Artifacts

Artifacts in Systinet represent an abstraction of your services enabling you to visualize and manage your service development, governance, and provision.

The service artifact is usually the central element of a set of artifacts representing a particular set of functionality that you provide or use.

In practice, a service cannot be represented by a single artifact and consists of a set of the following related artifacts:

- **Service**

  The key artifact representing the concept of the service. This artifact contains properties identifying the owners, contacts, and details about the set of functionality that it represents. Other details are represented in related artifacts.

- **Implementation**

  Services typically have at least one implementation. Each implementation represents functional elements of the service. There are a number of different kinds of implementation and they relate to sets of technical artifacts representing the operations the implementation performs, the endpoints representing the servers or environments where the implementation is available, and definition documents such as WSDLs and XSDs that describe the service and its implementation. For more details, see "Implementations" on the next page.

- **Documentation**

Services are often described by different documents such as specifications and user guides.

- **Service Level Objectives**

  Providers of services can offer sets of terms to service consumers describing the service levels that they expect to meet. When a consumer requests the use of a service, the service level objective defines the terms of the contract between the provider and the consumer. For details about contracts, see "Contracts Overview" on page 223.

Systinet enables associates these different artifacts together as a single service aggregate enabling you to manage, govern, or use them all as a single object. For more details about aggregates, see "Aggregation" on page 154.

Business Process Artifacts

Systinet represents an abstraction of your processes enabling you to visualize and manage your process deployment and governance.

The Business Process Artifact is usually the central point of a process infrastructure representing a particular set of functionality that you provide or use.

In practice, a business process cannot be represented by a single artifact and consists of a set of the following related artifacts:

- **Business Process**

  The key artifact representing the concept of the process. This artifact contains properties identifying the owners, contacts, and details about the set of functionality that it represents.

- **Process Implementation**

  The entity representing the process implementation. A Business Process typically has one BPEL Process Implementation, which refers to related (consumed and provided) services and their implementations using an aggregation of WSDL documents. BPEL Process implementation is typically accessible and controlled by one service implementation, so a process implementation is also visible as a service implementation, and can be reused by another service or process. For details, see "Implementations" below.

- **Documentation**

  Processes are often described by different documents such as specifications and user guides.

Systinet associates these different artifacts together as a single process aggregate enabling you to manage, govern, or use them all as a single object. For more details about aggregates, see "Aggregation" on page 154.

Implementations

Implementations represent the technical and functional elements of a service or process. Systinet supports the following implementation artifact types:

- **Application Interfaces**

  - SOAP Service

  - XML Service

  - REST Service

  - File Transfer

- In the earlier version of the guide Application Interfaces was called as Service Implementations.

- **Process Implementations**

  - BPEL Process

  - XPDL Process

Implementations are typically associated with the following artifact types:

- **Endpoints**

  Endpoints represent URLs used to access a service or process implementation. Throughout the lifecycle of a service or process, there are likely to be a number of different environments hosting the service or process, for example, development, testing, and production servers. Use a different endpoint to represent each of these environment implementations.

- **Operations**

  Operations represent a method defined in an implementation.

- **WSDLs**, **XML Schemas**, **BPELs**, and **XPDLs**

  Services, processes and their implementations are often described by definition documents such as WSDLs or BPELs. Systinet provides publishing functionality which decomposes definition documents, analyzes the service infrastructure it defines, and then creates or updates the relevant artifacts in the Catalog. For more details, see .

  > **Note:** BPEL and XPDL definition documents are directly attached to the relevant BPEL and XPDL Process artifacts.

  > **Note:** After the publication of a BPEL/XPDL file, there must be a "consists of" relationship from the BPEL/XPDL process artifacts to the SOAP Service artifacts which define the imported WSDLs of BPEl/XPDL.

Implementations and Endpoints can use different messaging transport methods. Systinet supports HTTP, HTTPS, SMTP, and JMS transport methods.

Typically, implementations are deployed to several endpoints during their lifecycle which may represent different environments. Systinet provides environment support which reduces the overhead involved with maintaining artifacts for multi-environment deployment. For details, see "Environments" on the next page.

Relationships

The relationships between artifacts represent one of the most significant aspects of authoring and artifact management.

For each artifact type, the SDM Model defines a set of relationships where each named relationship defines a link between that artifact type and another artifact type which is expected to be associated with it. For details about the model, see "Service Definition Model (SDM)" on page 57.

Every relationship type is directional from one artifact type to another and has an inverse relationship in the opposite direction. For example, the relationship from a Service Artifact to a Documentation Artifact is the *Documentation* relationship and the inverse relationship from the Documentation Artifact to the Service Artifact is the *DocumentationOf* relationship.

Relationships can be broadly divided into the following types:

- **Dependencies**

  A *Dependency* is a directed relationship indicating that an artifact uses another artifact. For example, Service A may have a *depends on* relationship to Service B to indicate that Service A uses Service B and that any change to the Service B should take into account the impact on Service A. Contract relationships are a specific example of dependency relationships indicating the contractual use of an artifact according to specified usage terms. Various functions in the UI offer an **Include Dependencies** option (for example, Export to ZIP Archive) which takes these dependencies into account when performing actions on an artifact.

  The inverse of a dependency is an *Impact*. Impact enables you to view the dependent artifacts which may be affected by a change to an artifact. The most useful way to view impact and dependencies is in the Navigator view. For details, see "How to Highlight Impact and Dependency" on page 93.

- **Aggregations**

  Aggregation relationships enable a set of related artifacts to be represented as a single abstraction in order to perform operations on the entire set. For example, a service aggregation consists of not only the service itself, but also its implementations, operations, endpoints, SLOs, documentation, etc. The concept of aggregation has a significant impact on the functionality of Systinet and is discussed in more detail in "Aggregation" on page 154.

- **Unclassified Associations**

  In addition to relationships in the public model, there are additional system relationships which Systinet automatically applies in specific circumstances. For example, when you create a new version of an artifact, a previous version relationship links the new version to the old one. This relationship enables you to notify users of the old version when the new version is ready to use.

The user interface provides extensive support for named relationship types, showing the most important relationship in the Artifact Details page Overview tab, and all the expected relationships in the Details tab. The Relationships tab provides an alternative view with the relationships organized into Outgoing and Incoming Relationships. For details, see "Artifact Details Page Overview Tab" on page 28, "Artifact Details Page Details Tab" on page 32, and "Artifact Details Page Relationships Tab" on page 49.

In addition to the relationships defined by the model, there are use cases where a more generic relationship may be useful. Examples include defining project content, where the content may be any artifact type, or composite services where a particular service may be made up of other services. To support these use cases, Systinet includes generic **Consists Of** and **Depends On** relationships, representing sub-artifacts and parent artifacts respectively. Use these relationships to associate artifact types together when a specific named relationship does not exist.

Manage both named and generic relationships from the Artifact Details page. For details, see "How to Manage Relationships" on page 120.

Environments

Most organizations use several environments to clearly separate production from testing and development. This means that when a service moves through its lifecycle, it is likely to be deployed multiple times in a number of different environments.

Systinet makes use of an Environment categorization to represent these environments. Each environment is represented by an environment category value which can be applied to endpoints and also to server artifacts for other products integrated with Systinet (for example, UDDI and BSM servers).

This categorization ensures that only the appropriate information is automatically exchanged between Systinet and integrated servers. For more details about integrated servers, see "Product Integration Overview" on page 291.

This approach avoids the constant republishing of WSDLs defining services to reflect these additional endpoints as the service moves through its lifecycle. Systinet provides environment specific WSDLs that automatically contain the appropriate endpoints whenever a WSDL is required.

A typical example for a service could be described as follows:

- **Development**

  A service is likely to be defined by publishing a WSDL before it reaches the development stage. When it reaches the development stage it requires an endpoint representing a development server. Instead of amending and republishing the WSDL, Systinet enables you to add an endpoint and categorize it as `Development`.

- **Testing**

During the testing stage, the service may require an additional endpoint representing test deployment of the service and this is also achieved by adding a new endpoint categorized as `Testing`.

Systinet may be integrated with *Application Lifecycle Manager* (ALM). When a service is registered in ALM, the associated WSDL for the test environment is exported to ALM. By categorizing the ALM server as `Testing`, you ensure that only the Testing endpoints are visible to the ALM server. For more details about ALM integration, see "ALM Integration" on page 332.

- **Staging**

  A staging environment is intended to duplicate the production environment as closely as possible. Staging environments are often used for final pre-production testing, or for demonstration and training purposes. A service in a staging environment requires an additional endpoint representing the staging deployment categorized as `Staging`.

- **Production**

  Similarly in production, the service requires an additional endpoint representing the production deployment of the service and this is also achieved by adding a new endpoint categorized as `Production`.

  When a service reaches production you may wish to export it to a UDDI registry. By categorizing the registry server as `Production` you ensure that the service is exported with the appropriate production endpoints. For more details about UDDI Registry introduction, see "UDDI Registry Integration" on page 295.

Publishing Overview

Files can be uploaded to Systinet with the Import File wizard accessible from the Import menu in the Catalog tab, or as Upload actions in Artifact Detail page Details tabs.

Systinet does not change uploaded files, they are stored and available as they were uploaded. You can upload local files or remote files exposed at a URL. Moreover, you can upload a ZIP archive containing multiple files. As well as the file to upload, you can select the server folder and set additional options.

Import File functionality offers the following options:

- **Server Folder**

  The server folder determines the path in the repository workspace where Systinet stores uploaded files.

  The correct use of server folders is crucial for definition data management. As mentioned above, definition data files usually refer to each other via relative links and so it is important to set server folders so that relative links work. For more details and examples, see "Server Folders" on page 114.

Each uploaded file is also accessible using the REST location space in the following URL format, `http://hostname:port/systinet/platform/rest/location/serverFolder/fileName`. For example, if you upload `a.wsdl` to server folder `/accounting` then the WSDL document is accessible using `http://hostname:port/systinet/platform/rest/location/accounting/a.wsdl`. For more details, see "Publishing Location Feeds" in the *Developer Guide*.

- **Authentication**

  This option applies when you upload remote documents and the remote server requires credentials. Import provides the option to input credentials and to store them for future use. For details, see "How to Manage Your Credentials".

- **Synchronization**

  This option applies when you upload remote documents. Systinet provides the option to synchronize the content of the repository with remote files. This feature is important when Systinet is not the master store for definition files. The administrator can schedule a synchronization task that synchronizes all documents in Systinet with remote master files.

  The behavior of synchronization depends on the following options:

  - **Automatic**

    Automatically update the repository file if the remote file is updated.

  - **Manual**

    If the remote file is updated, the repository file is marked as *out-of-sync*. Systinet provides functionality to update each out-of-sync repsoitory file manually.

  - **Disabled**

    No synchronization takes place.

- **Process Archive Content**

  This option applies when you upload a ZIP archive. If selected (default), Systinet extracts the archive and uploads all the content. Otherwise, Systinet uploads the archive as documentation content attached to a documentation artifact. ZIP, JAR, WAR, EAR, and BPR archives are supported by default, but you can extend the predefined set of supported archives.

- **BPEL Decomposition**

  This option applies when you upload BPEL or SCA documents. It is similar to WSDL decomposition. You can select whether to create only data content artifacts, such as SCA or BPEL artifacts (None option) or whether to create a Business Process for each BPEL artifact.

- **WSDL Decomposition**

This option applies when you upload a WSDL. You can select from the following WSDL decomposition options to enable the publishing process to create the artifacts defined by the WSDL:

- **None**

  Create only data content artifacts, such as WSDL and XSD artifacts, with attached data content.

- **Implementations**

  Create data content artifacts and also parse the WSDLs and create appropriate implementation artifacts, such as SOAP Service, Operation, and Endpoint artifacts.

- **Services**

  In addition to the implementation artifacts, create appropriate service artifacts.

- **Service Type**

  This option applies if you upload a WSDL document and create services when WSDL decomposition is set to Services. You can specify the service type (infrastructure, business, or application service) for the created service.

> **Note:** Systinet controls the functionality of publishing with a set of configurable properties. For details of these properties and how to change them see the Publishing table in "System Configuration Properties" and "How to Manage System Settings" in the *Administration Guide*.

For more details about the process of publishing, see "How to Publish Content" on page 127.

Publishing includes support for versioning. For details about using publishing to control versioning, see "How to Upload Versioned Data Files" on page 217.

Server Folders

Server folders specify data file locations in the Systinet repository workspace.

Systinet offers the repository workspace as a primary storage location for data definition files. By specifying server folder you can determine the storage location in the workspace which is crucial for retaining valid relative links to already uploaded documents. The workspace can be browsed with the select server folder dialog in the File Import page or with the REST Interface.

HP recommends using relative links and to design server folders in the workspace appropriately. For more details, see "Definition Data Best Practice" in the *User Guide*.

Systinet exposes the workspace using REST as a public interface. You can browse the workspace at `http://hostname:port/systinet/platform/rest/location`. This URL opens a list of top level server folders and files uploaded directly to the root folder.

You can download a selected folder as an archive. For example, you can download all data definition files in a server folder to your local disk as a ZIP archive, change them according to your needs and then upload them again to the repository. Alternatively, you can use Systinet IDE Integration Plug-Ins or WebDAV for this purpose. For more details, see *Plugin for Eclipse*, *Plugin for Visual Studio*, and "WebDAV Compliant Publishing" in the *Developer Guide*.

> **Note:** Systinet does not support the upload of ZIP files using WebDav. If you need to upload a ZIP file, use the import functionality of the UI or upload functionality of the IDE Plug-Ins.

You can browse a selected folder by clicking its name. When you view the content of specific folder, the URL changes. For example, when you browse the `/accounting` folder then the URL is `http://hostname:port/systinet/platform/rest/location/accounting`. For more details, see "Publishing Location Feeds" in the *Developer Guide*.

The following examples describe WSDL and XML schema documents, but are also valid for all data definition files with relative references:

Uploading a WSDL Document and Creating a New XML Schema

In this example you upload a ZIP archive including `account.wsdl` and `account.xsd`. The WSDL document contains relative reference to `account.xsd`. The following diagrams describe the relative reference and the ZIP structure. Notice that the relative links correspond to ZIP structure.


```
<?xml version="1.0" encoding="UTF-8"?>

<definitions
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns="http://schemas.xmlsoap.org/wsdl/"
    xmlns:tns="http://systinet.com/uddi/account/5.0"
    xmlns:account="http://systinet.com/uddi/account/5.0"
    targetNamespace="http://systinet.com/uddi/account/5.0">

    <types>
        <xsd:schema targetNamespace="http://systinet.com/uddi/account/5.0" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <xsd:include schemaLocation="./schemas/account.xsd"/>
        </xsd:schema>
    </types>
```

WSDL document refers XML Schema document relatively via ./schemas/account.xsd
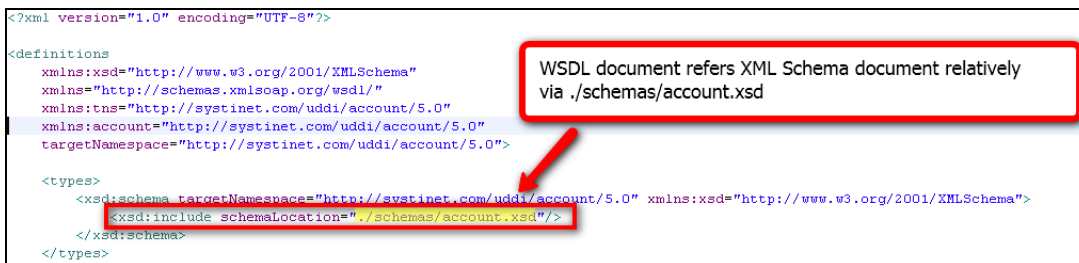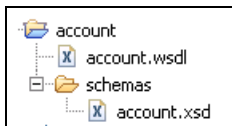
Both files are new and do not exist in the repository. When you upload, specify server folder `/accounting` and so Systinet creates `account.wsdl` in `/accounting/account.wsdl` (accessible from `http://hostname:port/systinet/platform/rest/location/accounting/account.wsdl`) and `account.xsd` in `/accounting/schemas/account.xsd` (accessible from `http://hostname:port/systinet/platform/rest/location/accounting/schemas/account.xsd`).

Uploading a WSDL Document and Reusing an Existing XML Schema

When you want to reuse an existing XML Schema document you must use the correct relative link in your WSDL document and specify an appropriate server folder when uploading.

Assuming that you want to reuse an existing `core.xsd` in a shared `/schemas` folder in a new `account.wsdl` that you upload to the `/accounting` server folder. You need to use an appropriate relative link in your WSDL document.

Examining the intended structure of the repository workspace after uploading `account.wsdl`, the link must be `./../schemas/core.xsd`.
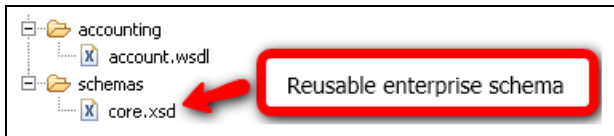
Service Provision Workflow

Systinet uses a set of authoring processes, enabling you to add services to the Catalog and make them available to consumers.

> **Note:** This workflow considers the steps required for Service Provision separately from Lifecycle Management. In practice, these steps are associated with particular lifecycle stages with relevant verification and approval processes. For more details, see "Lifecycle Overview" on page 238.

The diagram describes a typical service provision process and the steps required to achieve it. The workflow is split into the user roles that would typically perform the actions in an organization.
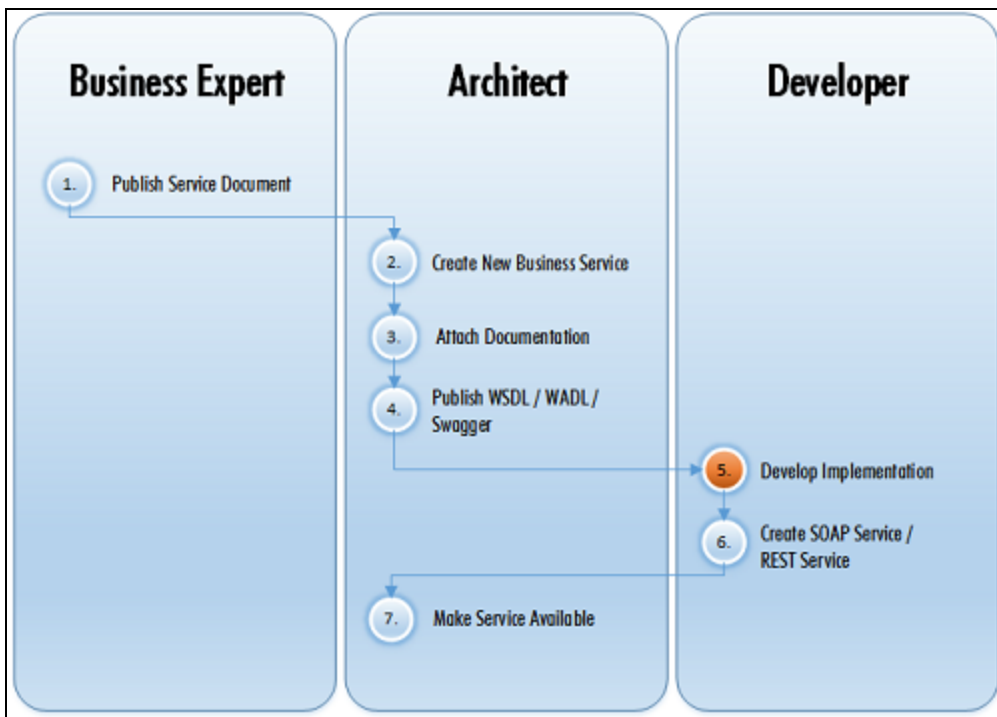
1. The Business Analyst publishes a service document in Systinet describing a proposed new service. For details, see "How to Publish Content" on page 127.

2. The Architect receives notification of the new service requirement, reviews the documentation,

and creates a new service artifact. For details, see "How to Create Artifacts" below.

3. The Architect creates a new technical document and attaches the new document and the existing service document to the service. For details, see "How to Attach Documentation" on page 123.

4. The Architect publishes a WSDL document, which contains a technical description of the service. The architect can use the publishing functionality of Systinet or develop the WSDL in an integrated IDE and publish it directly. For details, see "How to Publish Content" on page 127.

5. The Developer reviews the documentation and uses the WSDL to develop an implementation of the service. Typically, the developer would use an integrated IDE to develop the implementation. HP Software provides plugins for Eclipse and Visual Studio that integrate Systinet discovery and publishing functionality in these IDEs. For details, see *Plugin for Eclipse* and *Plugin for Visual Studio*.

6. The Developer publishes the WSDL. Systinet publishing functionality creates the artifact structure defined by the WSDL and modifies any relevant existing artifacts. The Developer can use the publishing functionality in Systinet or publish directly from an integrated IDE. For details, see "How to Publish Content" on page 127 or *Plugin for Eclipse* and *Plugin for Visual Studio*.

7. The Architect edits the service and makes it available for consumption. For details, see "How to Edit Artifact Properties" on page 163.

How to Create Artifacts

You can create artifacts in Systinet in the following ways:

- Automatically, using the Import publishing feature.

- Manually, using the UI

This topic describes the manual creation of artifacts. For details about automatic artifact creation using publishing, see "How to Publish Content" on page 127.

> **Note:** In addition to the artifacts listed in this topic, the administrator has additional permissions to create system artifacts. For details, see the relevant sections in the *Administration Guide*.

To Create Artifacts:

1. Access the Create Artifact page in one of the following ways:
   - In the Catalog tab Artifact menu, expand the **Create** menu and select the artifact type you want to create.

   > **Note:** The create menu only lists the most important artifact types. To see all the artifact

> types you can create, expand the **Create** menu and click **More...** to open the Catalog Browser in create mode.

- In an Artifact Detail page with a table of attached artifacts, click **Create**, and for tables that allow multiple artifact types, select the artifact type to create.

- In Artifact Browse pages, click **Create**.

The Create Artifact page opens for the specific artifact type you want to create.

2. Input the specific details for the artifact type referring to the relevant UI Reference topic for details:

Content Artifacts	Address Book Artifacts
"Create Application Page" on page 140	"Create Organizational Unit Page" on page 146
"Create Business Process Page" on page 141	"Create Contact Page" on page 141
"Create Endpoint Page" on page 143	
"Create File Endpoint Page"	
"Create SOAP Service Page" on page 151	
"Create XML Service Page" on page 152	
"Create SOAP Operation Page"	
"Create Project Page" on page 147	
"Create Service Page" on page 148	
"Create SLO Page" on page 149	
"Create REST Service Page"	
"Create HTTP Request Page"	
"Create HTTP Response Page"	
"Create File Transfer Page"	

> **Note:** The set of artifacts available depends on your role and permissions.

3. Click **Save** to create the artifact.

> **Note:** The Administrator can assign policies to validate artifact creation. These policies inform you if you have not completed all the required details for the artifact. For details, see "How to Manage Artifact Form Validation" in the *Administration Guide*.

How to Manage Contacts

Systinet enables you to define contacts for artifacts in various roles.

The primary purpose of contacts is to establish who is responsible for artifacts, to define their contact roles, and to simplify the process of notifying stakeholders. A contact can be a user, an external contact, or an organizational unit. For details about notifications, see "How to Use Notifications" on page 96.

To Access Contact Information for an Artifact:

- Do one of the following:

  - Open the Artifact Details page and in the Overview tab, in the Contacts component, click **Edit** to open the Contacts tab.

    Screenshot: Artifact Details Page Overview Tab Contacts Component
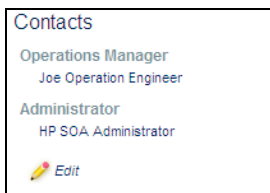
  - Open the Artifact Details page, and select the Contacts tab.

  The Artifact Details Contacts tab opens display a list of roles and the current contacts for each role.

  > **Note:** The contact roles are defined by the job roles system taxonomy and have no direct relationship to assigned user roles in Systinet.

Screenshot: Artifact Details Page Contacts Tab
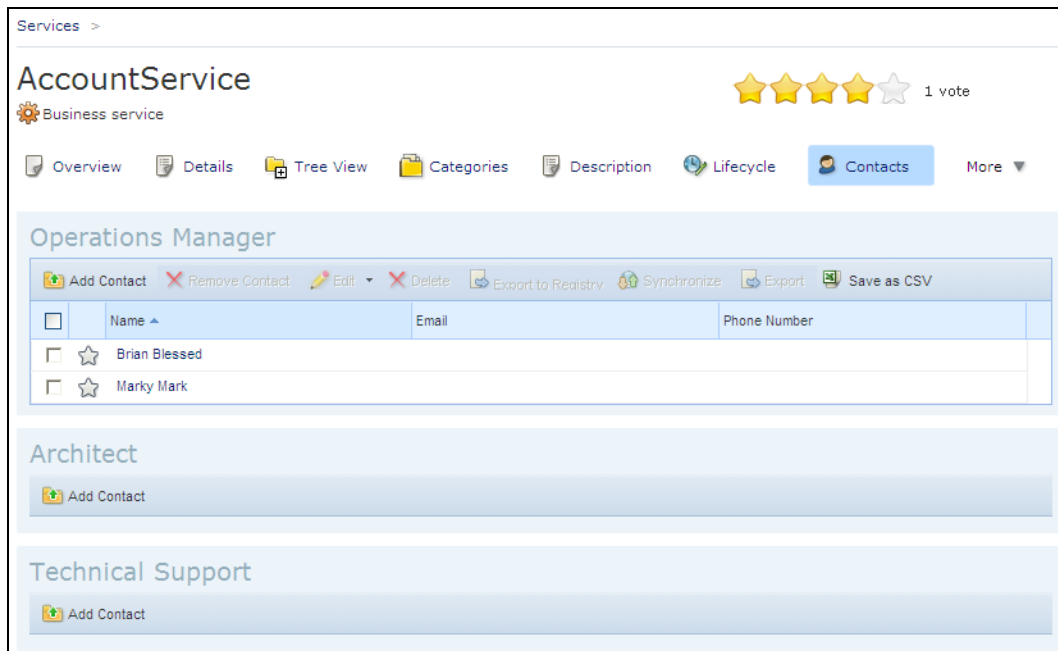
To Manage Contacts for an Artifact:

1. For the role you want to add, click **Add Contact** to open the Add Contact dialog box.

2. Do one of the following:
   - Type in the contact. Systinet returns a list of available contacts (user profiles and organizational units). Select a contact from the list of results.

   - Click **Browse Addressbook** to select People and Organization Units in the local or global address books.

   - Expand **Create Contact**, and select **Organizational Unit** or **Contact** to create a new contact to add for the artifact. For parameter details, see "Create Contact Page" on page 141 and "Create Organizational Unit Page" on page 146.

How to Manage Relationships

At the heart of Systinet is the ability to link artifacts together with relationships. For more details about the concept of relationships, see "Relationships" on page 110.

To Access the Relationships for an Artifact:

- Do one of the following:

   - Open the Artifact Details page and in the Overview tab, in the Related Artifact component, click **Details** to open the Details tab.

Screenshot: Artifact Details Page Overview Tab Related Artifact Component
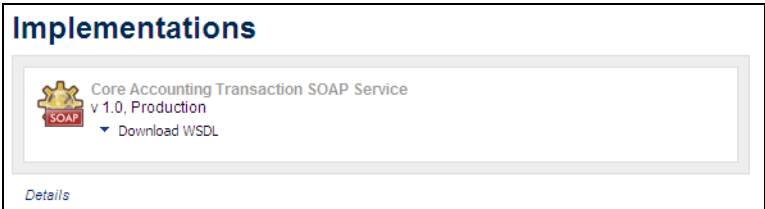
- Open the Artifact Details page, and select the **Details** tab.

  The Artifact Details page Details tab opens displaying a set of related artifact tables organized by artifact types.
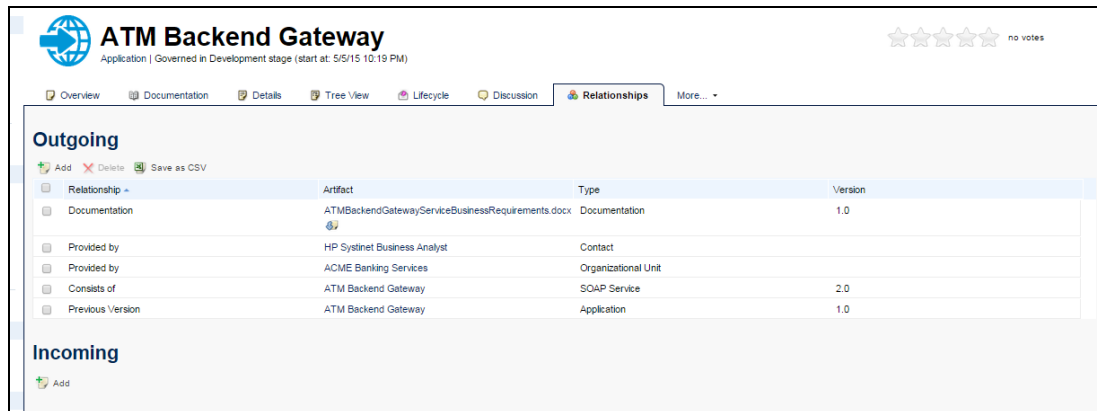
Screenshot: Artifact Details Page Details Tab
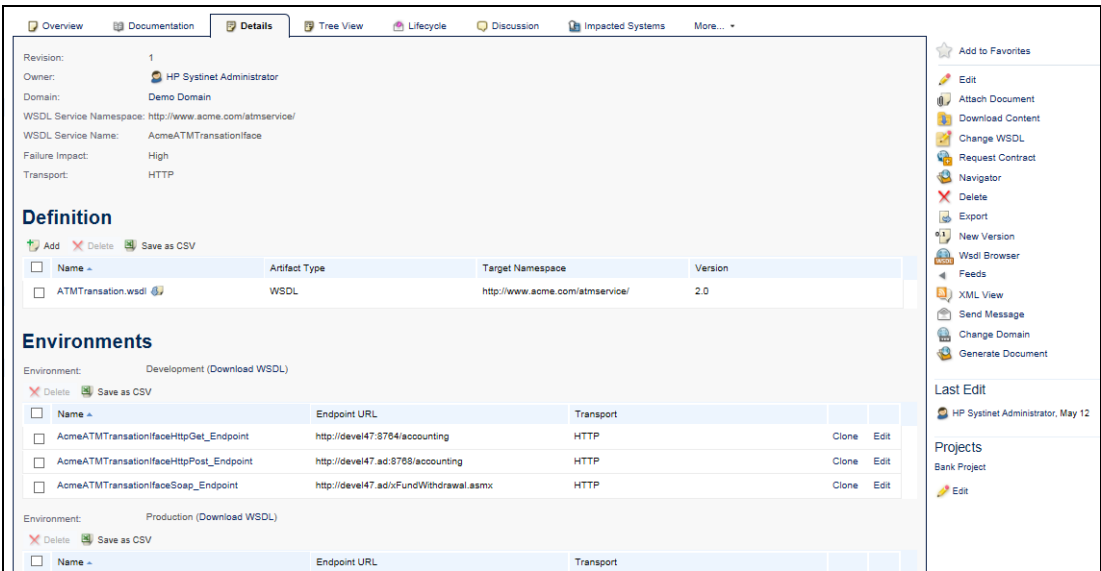
- Open the Artifact Details page, and select the **Relationships** tab.

  The Artifact Details page Relationships tab opens displaying the set of related artifacts organized by outgoing and incoming relationship types.

Screenshot: Artifact Details Page Relationships Tab

Each table shows the related artifacts of that type with options to add and remove the relationships in addition to the normal functionality of tables.

To Add Related Artifacts:

- Do one of the following:

  - To add an existing artifact from the Catalog, click **Add** in an artifact type or relationship table in the Details or Relationships tab to open the Select Artifacts or Add Relationship dialog box. Use the Search criteria to locate the artifact you want and click **Add** or **Select**.

    > **Note:** For the Relationship tab, you must also select the relationship type, and then click **Add**.

  - To add a new artifact, click **Create** in an artifact type table in the Details tab to open an artifact type selection menu or to directly open the artifact create page for the artifact type. For details, see "How to Create Artifacts" on page 117.

  - To add a new artifact with attached data content, click **Upload** in an artifact type table in the Details tab to open the Upload File dialog box. For details, see "How to Publish Content" on page 127.

    > **Note**: Systinet provides additional options for documentation. For details, see "How to Attach Documentation" on the next page.

To Remove or Delete Related Artifacts:

1. In the Details or Relationships tab, select the artifacts or relationships that you want to remove or delete and click **Delete** to open the Remove Relations dialog box.

2. Expand **Advanced Options** and do one of the following:

   ■ Select **Remove Relationships Only** to remove the relationship without deleting the related artifacts.

   ■ Select **Delete** and any of the following options:

      ○ Select **Include Subartifacts** to also delete related artifacts below the selected artifacts in the aggregate structure.

      ○ By default, deletion checks for relationships to other artifacts and does not delete the artifact if there are any. Select **Ignore Usage** to delete the related artifact even if it is related to other artifacts.

3. Click **OK** to confirm your removal or deletion. Systinet places deleted artifacts in the Recycle Bin. For details, see "Recycle Bin" on page 158.

How to Attach Documentation

Artifacts often have associated documentation.

Systinet provides the following ways to associate artifacts with documentation.

To Add an Existing Documentation Artifact:

- Use the **Link Existing** relationship functionality of the Documentation table in the Artifact Details page Documentation tab. For details, see "How to Manage Relationships" on page 120.

To Add a New Documentation Artifact:

- Use the **Create** relationship functionality of the Documentation table in the Artifact Details page Documentation tab. For details, see "How to Manage Relationships" on page 120.

To Upload a New Document as a Related Documentation Artifact:

1. Do one of the following to open the Attach Document dialog box:

   ■ In the Artifact Details page Overview or Details tab, click the **Attach Document** context action.

   ■ In the Artifact Details page Documentation tab, click the **Add** action in the Documentation table.

2. In the Attach Document dialog box, do one of the following:

   ■ **To Upload from a Local File System:**

      i. Select the **File** source.

      ii. Use **Browse** or input a file path to select the file to upload.

    iii. Select a **Type** to categorize the document.

    iv. Input the Server Folder to store the data content files in the repository.

> **Note:** Click **Select** to browse the repository workspace.

    v. Optionally, expand **Advanced Options** and add Keywords to tag the new documentation artifact. For more details, see "How to Edit Artifact Keywords" on page 177.

    vi. Click **Upload** to create a new Documentation artifact with the referenced document attached, and the relationship to the original artifact.

- **To Upload from a Remote Location:**

    i. Select the **URL** source.

    ii. Input the URL for the remote file location.

    iii. Select a **Type** to categorize the document.

    iv. Input the Server Folder to store the data content files in the repository.

> **Note:** Click **Select** to browse the repository workspace.

    v. Optionally, change the Location after Upload.

    vi. Optionally, expand **Advanced Options** and select from the following options:

Options	Description
Keywords	Add Keywords to tag the new documentation artifact. For more details, see "How to Edit Artifact Keywords" on page 177.
URL Requires Authentication	If the remote location requires authentication to access, select **URL Requires Authentication**, and input the username and password required to access it. Optionally, select to store the credentials in your credentials store for later access to the same location. For details, see "How to Manage Your Credentials" on page 52.
Synchronization	Select a Synchronization policy for change management of artifacts referenced to a remote location.

    vii. Click **Upload** to create a new Documentation artifact with the referenced document attached, and the relationship to the original artifact.

To Reference an External Document:

1. In the Artifact Details page Documentation tab, click the **Link a Web Resource** action in the Documentation table to open the Add Document Reference dialog box.

2. Input the document URL and the link text that you want to appear in the table of documentation.

3. Click **Add** to create a new Documentation artifact and the relationship to the original artifact.

> **Note:** The external document is not uploaded to the repository workspace by this functionality.

To View or Edit a Microsoft Office Document:

Once a Microsoft Office document has been imported, you can view or edit it directly in the associated Microsoft Office application.

> **Note:** Internet Explorer and Firefox browsers are supported.

To view a Microsoft Office document:

- Go to the Documentation page, choose a document from the table and click it. The Documentation Overview will be displayed. Click the **Open in Office** command on the right-side menu. The document will open in the associated Microsoft Office application.

To edit a Microsoft Office document:

- Go to the Documentation page, choose a document from the table and click it. The Documentation Overview will be displayed. Click the **Open in Office** command on the right-side menu. The document will open in the associated Microsoft Office application. After you finish editing, Save and Close the file. The imported file updates automatically within HP Systinet.

- You can also open the file directly in the associated Microsoft Office application to edit it. After saving, the file will update automatically within HP Systinet.

> **Note:** You will be required to confirm that the document you are opening is safe and that you have permissions to edit the file.

To Customize Microsoft Office Associations:

The Administrator can customize which applications and plug-ins Microsoft Office uses with files in Systinet by modifying the following System Settings configuration values:

Parameter	Value
platform.rest.dav.msoffice.activex	ActiveX used by IE for loading MS Office applications to edit documents remotely.

Parameter	Value
platform.rest.dav.msoffice.plugin.filename	Name of the plug-in used by Firefox for loading MS Office applications to edit documents remotely.
platform.rest.dav.msoffice.editor	Prefix of the editors used to load the correct MS Office applications (Word/Excel/PowerPoint) for the corresponding Office types.

For details, see "How to Manage System Settings" in the *Administration Guide*.

Supported Microsoft Office File Types

Extension	Description
.doc	Word 97–2003 Document
.dot	Word 97–2003 Template
.docx	Word 2007-2010 Document
.dotx	Word 2007-2010 Template
.docm	Word 2007-2010 Macro-Enabled Document
.dotm	Word 2007-2010 Macro-Enabled Template
.xls	Excel 97–2003 Workbook
.xlt	Excel 97-2003 Template
.xlsx	Excel 2007-2010 Workbook
.xltx	Excel 2007-2010 Template
.xlsm	Excel 2007-2010 Macro-Enabled Workbook
.xltm	Excel 2007-2010 Macro-Enabled Template
.xlsb	Excel 2007-2010 Binary Workbook
.ppt	PowerPoint 97–2003 Presentation
.pot	PowerPoint 97–2003 Template
.pps	PowerPoint 97–2003 Show
.pptx	PowerPoint 2007-2010 Presentation
.potx	PowerPoint 2007-2010 Template
.ppsx	PowerPoint 2007-2010 Show
.pptm	PowerPoint 2007-2010 Macro-Enabled Presentation

Extension	Description
.potm	PowerPoint 2007-2010 Macro-Enabled Template
.ppsm	PowerPoint 2007-2010 Macro-Enabled Show

How to Use Projects

You can organize your artifacts into projects, enabling you to govern a project as a whole and track its progress.

To Create Projects:

- Use the Catalog tab Artifacts menu to access the Create Project page. For details, see "How to Create Artifacts" on page 117 and "Create Project Page" on page 147.

To Manage Project Content:

1. Open the Project Details page Details tab.

2. Use the generic **Add**, **Create**, and **Delete** functionality of the Consists Of table to manage the content of the project. For details, see "How to Manage Relationships" on page 120.

   > **Note:** Add functionality does not enable you to add content that already belongs to another project.

Once an artifact belongs to a project the Artifact Details page Details tab displays project information.

The component shows the current project that the artifact is part of. Click the project name to view its details page and edit its content.

How to Publish Content

Systinet enables you to upload data content with specific support for definition documents that describe application infrastructures. Systinet processes the content of the document and enables you to create or modify artifacts to represent this infrastructure in Systinet.

Systinet can process the following definition document types:

- Web Service Definition Language documents (**WSDL 1.1 and 2.0**)

- Web Application Description Language documents (**WADL 2.0**)

- Swagger (**Swagger 1.2**)

- Service Component Architecture documents (**SCA 1.0**)

- Business Process Execution Language documents (**BPEL 1.1 and 2.0**)

- XML Process Definition Language documents (**XPDL 2.1**)

- XML Schema Definition documents (**XSD**)

- Extensible Stylesheet Language Transformation documents (**XSLT**)

- Document Type Definition documents (**DTD**)

- Any other file type is treated as documentation.

- Archives containing the above types of files.

SOAP Services use a transport protocol, typically HTTP (known as SOAP over HTTP). It is also possible to use JMS (Java Message Service) as a transport protocol for underlying SOAP communication. JMS provides a more robust and reliable method for transporting SOAP messages than HTTP. Web services that use the JMS transport protocol can be described in WSDL attributes. Systinet publishing detects these attributes and applies them to the appropriate artifacts during upload.

> **Note:** The WADL file containing the DDT code inline is not allowed to be parsed due to security risk. To enable parsing of the data we need to change the value of **platform.publishing.disallowDoctypeDecl** system property to **false**.

To Publish Data Content:

1. In the Catalog tab Import menu, click **File** to open the File Import page.

2. In the File Import page do one of the following:

   - **To upload from a local file system:**

     i. Select the **File** tab.

     ii. Use **Browse** or input a file path to select the file to upload.

     iii. Input the Server Folder to store the data content files in the repository. For details, see "Server Folders" on page 114.

        > **Note:** Click **Select** to browse the repository workspace.

   - **To upload from a remote location:**

i. Select the **URL** tab.

ii. Input the URL for the remote file location.

iii. Input the Server Folder to store the data content files in the repository. For details, see "Server Folders" on page 114.

> **Note:** Click **Select** to browse the repository workspace.

3. *Optional*: Expand **Advanced Options** and select from the following options:

Parameter	Definition
URL Requires Authentication (URL only)	If the remote location requires authentication to access, select **URL Requires Authentication**, and input the username and password required to access it. Optionally, select to store the credentials in your credentials store for later access to the same location. For details, see "How to Manage Your Credentials" on page 52.
Synchronization (URL only)	Select a Synchronization policy for change management.
Local Changes	Select whether to preserve local copies that are different or overwrite them.
Process Archive Content	Select to extract an archive and publish all the content (default option). If not selected, the archive is published as a Documentation artifact with attached data content. **Note:** By default, this functionality supports ZIP, JAR, EAR, WAR, and BPR. Other archive types are not processed by the publisher and are published as documentation artifacts with attached data content. The administrator can configure the types of archive that the publisher processes.
BPEL Decomposition	Select whether to create Business Process artifacts defined by a BPEL document.

Parameter	Definition
WSDL Decomposition	Select a WSDL processing level from the following options: ■ **Services** Process any WSDLs, create all defined content and create business services to associate with any newly created implementations. ■ **Implementations** Process any WSDLs and create all defined content. Business services are not created. ■ **None** Publish any WSDLs as artifacts with attached data content. The WSDL files are not processed.
Service Type	Select a service type from the available options: ■ **Business Service** A specific business function that offers an interface to users for interaction which is designed for reuse in multiple applications. ■ **Application Service** A business function associated with a specific application. Typically identified and defined by application developers and specific to the application scope. ■ **Infrastructure Service** A technical support function, such as authentication or logging, designed for reuse in multiple applications and services

4. Click **Import**. The import report page opens showing the publishing task in progress.

The import page displays the result of the publishing process, showing all files including any additional files referenced by the uploaded file.

This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description
100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet.This prevents the client from timing out and assuming the request was lost.
200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.

208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.
226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.
301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.
306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" below below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.

406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.
409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.
414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.

422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.
428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.
444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).

495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.
499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.
502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.

511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products
Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain
Content-Type	multipart/form-data	Administrator	Top-level Domain

Note: HP recommends updating Oracle Database schema statistics after importing large amounts of data. Old statistics may impact the performance of some data queries. Consult your database administrator.

To Update Oracle Schema Statistics:

- Execute the following command:

   EXEC DBMS_STATS.GATHER_SCHEMA_STATS (ownname => '&1',no_invalidate => FALSE,options => 'GATHER');

   This command does not require database admin privileges and can be run by the schema owner (ownname).

After publishing is complete, there is an option to **Undo Changes**. Click **Undo Changes** to undo changes to published artifacts. This also executes as a bulk operation.

Note: Undo only applies to the revision of the artifact created by publishing. Any subsequent modifications of an artifact change its revision and the Undo option does not rollback these artifacts.

Systinet also uses the definition document processing functionality of the publisher if you publish a document using the Upload functionality of related data artifact tables in the Artifact Detail page Details tab. For details, see "How to Manage Relationships" on page 120.

How to Try Out with DHC

Systinet enables you to use Try Out HTTP Request with DHC Restlet in Chrome browser.

DHC (Dev HTTP Client) Restlet is a full featured tool for manual HTTP/REST API testing and discovery. It is designed and developed by developers for developers to make direct HTTP/REST resource discovery, manipulation and testing more easily.

To use Try Out from HTTP Request page:

1. Go to https://chrome.google.com/webstore/category/apps from Google Chrome browser. Install and enable the **DHC Restlet** application.

2. Click **Try OUT** context option from HTTP Request details page.

3. You can also use the Try Out option from HTTP Requests table from REST service artifact.

How to Browse WSDLs and XML Schemas

After you upload a WSDL or an XML Schema, Systinet enables you to browse its content.

The WSDL / Schema Browser is available from the following pages in the UI:

- In the Artifact Details page for SOAP Services with an attached WSDL, the Overview and Details tabs contain a WSDL Browser context action. Click **WSDL Browser** to open a new Schema Browser window.

- In the Artifact Details page for WSDLs and XML Schema, the Overview tab displays the Schema Browser.

Screenshot: WSDL Browser

For WSDLs, the browser displays a list of the operations defined by the WSDL displaying the operation elements.

For XML Schema, the browser displays a list of constructs defined by the schema.

Click a context link for an element to open an XML Schema diagram for that element.

Screenshot: XML Schema Diagram
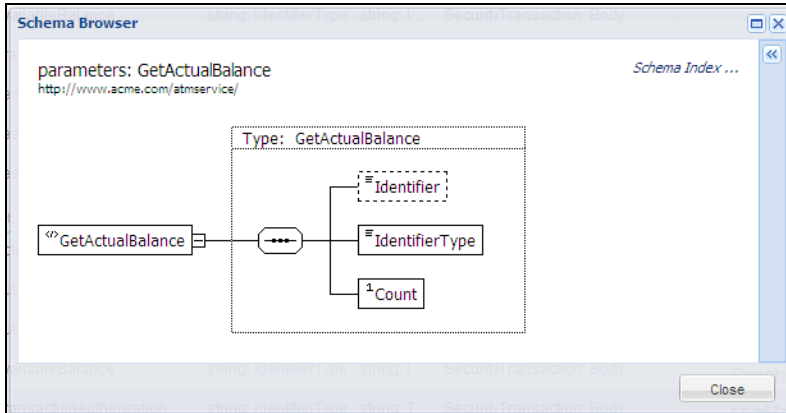
The XML Schema diagram offers the following options:

- Click ⟪ to show an XML representation of the main element.

- Click an element to display a dialog box with more details with an option to **Inspect** some elements in more detail. Click **Inspect** to open the XML Schema diagram for that element.

Authoring UI Reference

This chapter provides a reference to artifact creation and edit pages that are accessible from the Catalog tab.

Content Artifacts	Address Book Artifacts
"Create Application Page" on the next page	"Create Organizational Unit Page" on page 146
"Create Business Process Page" on page 141	"Create Contact Page" on page 141
"Create Endpoint Page" on page 143	
"Create File Endpoint Page"	
"Create SOAP Service Page" on page 151 "Create XML Service Page" on page 152	
"Create SOAP Operation Page"	
"Create Project Page" on page 147	

Content Artifacts	Address Book Artifacts
"Create Service Page" on page 148	
"Create SLO Page" on page 149	
"Create REST Service Page"	
"Create HTTP Request Page"	
"Create HTTP Response Page"	
"Create File Transfer Page"	

Note: The set of artifacts available depends on your role and permissions.

Create Application Page

The Create and Edit Application pages contain the following parameters:

Parameter	Definition
Name	The name of the new application.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details see, "Versioning Overview" on page 213
Alias	An alias for the application.
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Provider	Select the user, group, or role responsible for provision of the artifact.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create Business Process Page

The Create and Edit Business Process pages contain the following parameters:

Parameter	Definition
Name	The name of the new business process.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details, see "Versioning Overview" on page 213
Failure Impact	Select an impact from the drop-down list.
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Provider	Select the user, group, or role responsible for provision of the artifact.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create Contact Page

The Create and Edit Contact pages contain the following parameters:

Parameter	Definition
Name	The name of the new contact.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Language Code	Select a language from the drop-down list.
Email	Add e-mails for the contact.

Parameter	Definition
Phone	Add phone numbers for the contact.
Instant Messenger	Add IM contacts for the contact.
Address	Contact address for the contact.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Note: The administrator has different options for user profiles. For details, see "User Management" in the *Administration Guide*.

Create File Transfer Page

The Create and Edit File Transfer pages contain the following parameters:

Parameter	Definition
Name	The name of the new File Transfer.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details, see "Versioning Overview" on page 213.
Format	Format in which the file is transferred, for example: Binary, CSV, Text, XML.
Transport	Select the transport type from the drop-down list. Systinet supports CIFS, FTP, HTTP, HTTPS, SMTP,JMS, Local, MFT, MQ, SCP, SFTP and WebDAV transport methods.
Availability	Select the availability depending on the priority.
Integrity	Select the integrity for the file.
Confidentiality	Select the levels of confidentiality from the drop-down list.

Parameter	Definition
Version Remark	The version remark of the artifact.
Consumable	Select to make file transfer available to consumers.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .

Create File Endpoint Page

The Create and Edit File Endpoint pages contain the following parameters:

Parameter	Definition
Name	Name of the file endpoint.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Endpoint URL	URL for the actual implementation of the service.
Environment	Categorize the file endpoint by selecting an environment from the drop-down list.
Consumable	Select to make the file endpoint available to consumers.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .

Create Endpoint Page

The Create and Edit Endpoint pages contain the following parameters:

Parameter	Definition
Name	Name of the endpoint.

Parameter	Definition
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details, see "Versioning Overview" on page 213.
Endpoint URL	URL for the actual implementation of the service.
Transport	Select the transport type from the drop-down list. Systinet supports HTTP, HTTPS, SMTP, and JMS transport methods. Selecting JMS adds an additional set of JMS and JNDI parameters to the page.
Type	Select the endpoint type from the drop-down list.
WSDL Port Name	Port name defined by an associated WSDL.
Environment	Categorize the endpoint by selecting an environment from the drop-down list.
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Consumable	Select to make the endpoint available to consumers.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create HTTP Request Page

The Create and Edit HTTP Request pages contain the following parameters:

Parameter	Definition
Name	The name of the HTTP Request sent.

Parameter	Definition
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
HTTP Method	A set of common methods for HTTP Request. Example: CONNECT, GET, PUT, DELETE and so on.
URL Pattern	URL pattern used for depending on the HTTP Request. Example: For **Create user**, the URL pattern is **/user**.
Media Type	Indicates the type of data the artifact contains.
Resource	Use the http request resource to send the HTTP Request.
Stakeholder Emails	Email addresses of the stakeholders.
Consumable	Select to make HTTP Request available to consumers.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .

Create HTTP Response Page

The Create and Edit HTTP Response pages contain the following parameters:

Parameter	Definition
Name	The name of the HTTP Response received.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Media Type	Indicates the type of data the artifact contains.
Consumable	Select to make HTTP Response available to consumers.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .

Create SOAP Operation Page

The Create and Edit SOAP Operation pages contain the following parameters:

Parameter	Definition
Name	Name of the operation.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Operation Name	Operation name defined by the associated WSDL.
Port Type	Port type defined by the associated WSDL.
Target Namespace	Namespace defined by the associated WSDL.
Input Types	XSD input types defined for the operation.
Output Types	XSD output types defined for the operation.
Fault Types	XSD fault types defined for the operation.
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Consumable	Select to make SOAP Operation available to the consumers.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create Organizational Unit Page

The Create and Edit Organizational Unit pages contain the following parameters:

Parameter	Definition
Name	The name of the organizational unit.

Parameter	Definition
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Email	Add e-mails for the organizational unit.
Phone	Add phone numbers for the organizational unit.
Instant Messenger	Add IM contacts for the organizational unit.
Address	Contact address for the organizational unit.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create Project Page

The Create and Edit Project pages contain the following parameters:

Parameter	Definition
Name	The name of the new project.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details, see "Versioning Overview" on page 213.
Completion	The current progress of the project.
Start Date	Use the calendar icon to set a start date for the new project.

Parameter	Definition
Planned End Date	Use the calendar icon to set a projected completion date for the new project.
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Provider	Select the user, group, or role responsible for provision of the artifact.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create REST Service Page

The Create and Edit REST Service pages contain the following parameters:

Parameter	Definition
Name	The name of the new REST service.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details, see "Versioning Overview" on page 213.
Version Remark	The version remark of the artifact.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Consumable	Select to make the REST service available to consumers.
Documentation	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create Service Page

The Create and Edit Service pages contain the following parameters:

Parameter	Definition
Name	The name of the new service.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details, see "Versioning Overview" on page 213.
Service Type	Select a service type from the available options: • **Business Service** A specific business function that offers an interface to users for interaction which is designed for reuse in multiple applications. • **Application Service** A business function associated with a specific application. Typically identified and defined by application developers and specific to the application scope. • **Infrastructure Service** A technical support function, such as authentication or logging, designed for reuse in multiple applications and services
Failure Impact	Select an impact from the drop-down list.
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Provider	Select the user, group, or role responsible for provision of the artifact.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create SLO Page

The Create and Edit SLO pages contain the following parameters:

Parameter	Definition
Name	Name of the SLO.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details, see "Versioning Overview" on page 213.
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Throughput [Msg/Min]	Measured number of messages per minute.
Response Time [sec]	Measured time for the service to respond.
Availability [%]	Measured availability of the service during it operating hours.
Performance [%]	Measured performance of the service.
Hours of Provider Operation	Select an option and then enter start and end times if required.
Hours of Service Operation	Select an option and then enter start and end times if required.
Service Availability	Expected time of service delivery.
Service Termination	Expected time of service termination.
Expected Messages per Day	Estimated messages per day.
Maximum Messages per Day	Maximum messages per day.

Parameter	Definition
Daily Peak Period	Select an option and then enter start and end times if required.
Peak [Msg/Min]	Expected messaging activity.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create SOAP Service Page

The Create and Edit SOAP Service pages contain the following parameters:

Parameter	Definition
Name	The name of the new SOAP service.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details, see "Versioning Overview" on page 213.
Failure Impact	Select an impact from the drop-down list.
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Provider	Select the user, group, or role responsible for provision of the artifact.
Transport	Select the transport type from the drop-down list. Systinet supports HTTP, HTTPS, SMTP, and JMS transport methods.
JMS Provider	Set the JMS Provider if applicable.
Consumable	Select to make the SOAP Service available to consumers.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .

Parameter	Definition
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Create XML Service Page

The Create and Edit XML Service pages contain the following parameters:

Parameter	Definition
Name	The name of the new XML service.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. For details, see *User Guide* "Artifact Details Page Overview Tab" on page 28. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property. For details, see "How to Edit Artifact Specifications" on page 178 .
Version	The version number of the artifact. For details, see "Versioning Overview" on page 213.
Failure Impact	Select an impact from the drop-down list.
Keywords	A set of custom tags to help users identify the artifact. For details, see "How to Edit Artifact Keywords" on page 177.
Provider	Select the user, group, or role responsible for provision of the artifact.
Transport	Select the transport type from the drop-down list. Systinet supports HTTP, HTTPS, SMTP, and JMS transport methods.
Consumable	Select to make the XML Service available to consumers.
Categories	Apply a set of categorizations to help other users to identify the artifact. For details, see "How to Categorize Artifacts" on page 170 .
Documents	Attach documents to the artifact. For details, see "How to Attach Documentation" on page 123.

Chapter 6: Artifact Management

This part contains the following topics:

Concepts

Tasks

Reference

Artifact Management Overview

When you have data in the Catalog, there is a requirement to maintain and update it. Systinet provides extensive support for artifact maintenance covering the following functionality:

- "How to Use Bulk Operations" on page 159

  Maintaining artifacts individually is time-consuming, so Systinet offers bulk operations, enabling you to maintain multiple artifacts in a single operation.

- "How to Edit Artifacts" on page 161

  Artifacts contain a lot of information, so Systinet offers numerous edit options, enabling you to focus on particular aspects of the artifacts that you want to edit.

- "How to Delete Artifacts" on page 194

  Artifact deletion is a two-step process. Deleting artifacts places them in the Recycle Bin where they can then be restored or permanently removed from the Catalog. For detail, see "Recycle Bin" on page 158.

- "How to Synchronize Artifacts" on page 200

  Artifacts in the Catalog may have attached data content which originated from external content. Systinet provides change management with a synchronization tool enabling you to keep track of changes to the external content and to manually or automatically update the version of that content stored in the repository.

- "How to Export and Import Artifacts" on page 201

  Your deployment may be deployed across several instances of Systinet. Export and import functionality enables you to transfer content across instances or maintain backed-up versions of content for later reference. For more details about multi-instance deployment, see "Federation" on page 156.

- "Definition Data Best Practice" on page 211

  Systinet provides extensive support for definition files, such as WSDLs and XML Schemas. This best practice topic describes the recommended ways to exploit this functionality to best effect.

> **Note:** Systinet controls access to artifact management functionality based on user roles and permissions. For details, see "Security and Access Control" in the *Administration Guide*.

Aggregation

Abstractions such as services are not represented in the Catalog by a single artifact. Artifacts in Systinet form natural hierarchies corresponding to hierarchies of business objects. It is necessary to be able to maintain, govern, and provide these hierarchies or *aggregates* as a whole. For example, when moving service to a different domain it is necessary to be able to move not only to the service artifact itself but also to its aggregated artifacts such as SLOs, documentation, implementations, and endpoints.

For example, the following diagram shows a service aggregate with an implementation, its operations, endpoint, and WSDL. Performing a bulk operation on the service such as delete, offers the **Include Sub-artifacts** option, which would also delete the implementation, operations, endpoint, and WSDL.
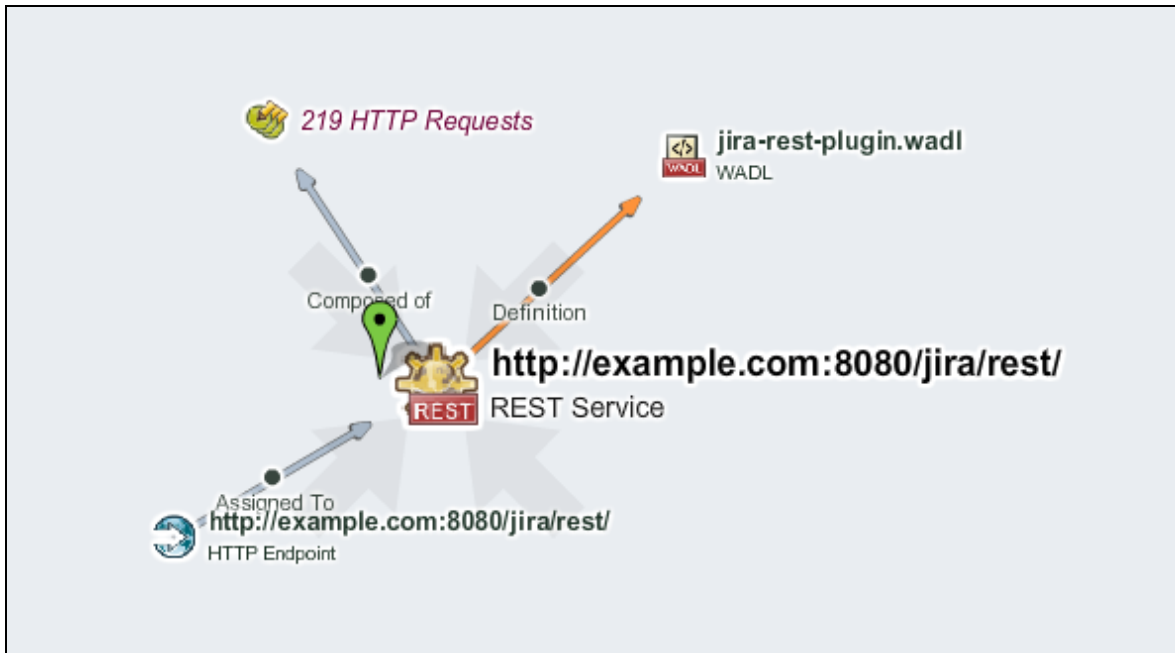
Systinet constructs these hierarchies or *aggregates* as follows:

- **Aggregations** are relationships whose relationship type is marked as an aggregation (relationship types in the model can be marked as aggregations, dependencies, or generic relationships). For example, the relationship *implementation of* between Service and Implementation artifact types is marked as an aggregation. Whenever there is an aggregation between artifacts A and B, we say that A aggregates B, for example, a service aggregates its implementations.

- **Sub-artifacts** of an artifact are all the artifacts that the artifact directly or indirectly aggregates (indirectly aggregated artifacts are artifacts aggregated by the artifact or any of its recursively aggregated artifacts).

- **Aggregates** are an artifact together with all its sub-artifacts. For example, an implementation with all its endpoints, operations, and WSDLs. Many operations, for example, change owner, can be applied to whole aggregates by selecting the **Include Sub-artifacts** option.

You can maintain, govern, and provide *aggregates* using the following functionality:

- **Artifact Export**

  When you export an artifact, Systinet also exports all related sub-artifacts in the same aggregate. For details, see "How to Export and Import Artifacts" on page 201.

- **Artifact Operations**

  Most operations that you can perform on an artifact or set of artifacts, such as delete and edit, offer an Advanced Option to **Include Subartifacts**. Selecting this option propagates the same operation to all sub-artifacts in the aggregate.

Another function that uses the ability to classify relationship types as aggregations or dependencies (or just regular relationships) is *Impact Analysis*. It enables you to assess the potential effect of a change. When assessing the impact of a change for an artifact, aggregations and dependencies are considered. So, artifacts that aggregate or depend on an artifact (directly or indirectly) may be impacted by a change.

You can visualize these aggregate structures and the impacts and dependencies they form using the following features:

- **Graphical Navigator**

  The Graphical Navigator provides a visual representation of the Catalog content. You can use hierarchical and custom layered layouts to examine specific aggregate structures and use highlighting to focus on the impact and dependencies between artifacts. For details, see "Navigator" on page 71.

- **Artifact Detail Tree View**

  The Tree View tab in Artifact Details pages enables you to view and navigate through the aggregates that the artifact is part of. You can switch between Impact view listing parent artifacts and Dependency view listing sub-artifacts. For details, see "Artifact Details Page Tree View Tab" on page 35.

Federation

Whether because of regulatory (for example, ISO 17799-based Separation of Environments and Duties) or organizational reasons, it may be necessary to deploy multiple physically separated repositories instead of a single centrally managed repository. In such environments, the synchronization and maintenance of data can prove problematic if the repository does not address these problems directly.

Systinet introduces new features that help to federate physically separated repositories in a controlled fashion. Users can control which data to federate by selecting a domain, project, or even particular applications that are subject to migration between repositories. They can also control whether lifecycle metadata, security, settings, etc. migrate together with the data or are adopted from the target repository. Systinet supports various federation topologies – for example, 'department' repositories federated in a central enterprise repository or an 'air-gap' scenario where there is no direct connectivity between the repositories.

Systinet provides the following scenarios, functionality, and concepts to support federation:

Multiple instances of Systinet

Many organizations require a physical separation between pre-production and production environments.

A typical use case for this scenario is a Staging instance for pre-production to manage the lifecycle of a development project, and to establish all the supporting data for services which can then be transferred to a Production instance which is primarily used by consumers and captures contract information.
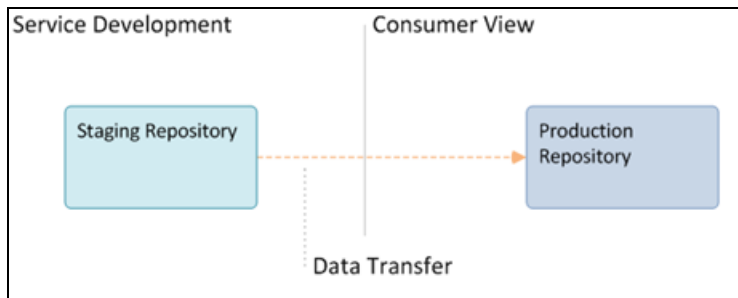
An alternative is multiple Staging instances for separated development paths.
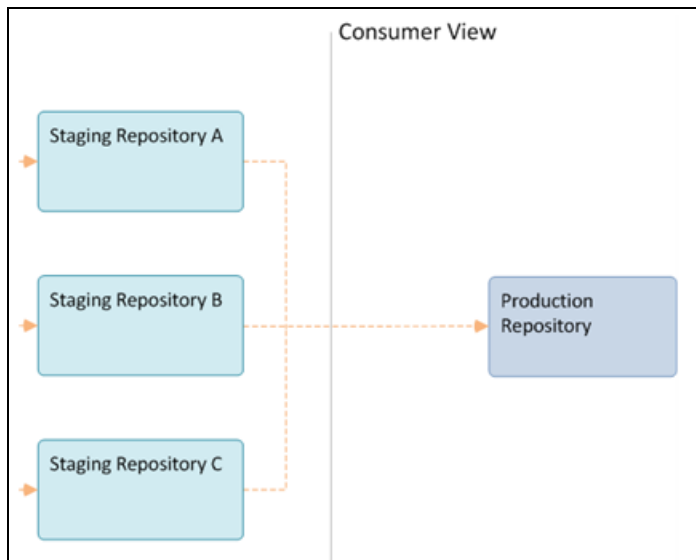
Domains

Systinet provides domain support enabling you to separate the Catalog into distinct areas. These domains are available for use in the same way as separate instances of Systinet. For more details, see "Domains Overview" in the *Administration Guide*.

Environments

Systinet enables you to apply specific environment categorizations to different endpoints or integrated product servers. This categorization enables the use of a single service definition document to define the same service across different instances or environments. For more details, see "Environments" on page 111.

Artifact Export / Import

Supporting the transfer of data between separate instances or domains is a selective export / import process. Export utilizes the concept of aggregation enabling you to perform an export on a single artifact, which collects all the related sub-artifacts in a single archive which can then be imported elsewhere. A typical scenario is to export a project from a Staging instance to a Production instance when its development lifecycle is complete and it is ready for use by consumers. For details about aggregation, projects, and export / import, see "Aggregation" on page 154, "How to Use Projects" on page 127, and "How to Export and Import Artifacts" on page 201.

Lifecycle and Domain Export / Import

In addition to transferring artifacts, Systinet uses the same export / import functionality to support the transfer of lifecycle processes and domains. Exporting a lifecycle process enables you to take advantage of generic lifecycle templates featuring the use of roles instead of specified users and groups and apply the same process across multiple instances or domains. Domain export enables to transfer of the entire content of a domain, including its configuration to another instance or domain. For details, see "How to Export Lifecycle Processes" and "How to Export Domain Content".

Repository Export / Import

In addition to the export / import functionality provided in the user interface for artifact, lifecycle, and domain transfer. The administrator has command-line export / import tools to create backups or copies of the entire repository including all its configuration and security data. For details, see "Export Tool" and "Import Tool" in the *Administration Guide*.

Recycle Bin

Systinet places all deleted items in the Recycle Bin.

> **Note:** By default, Systinet runs a recycle bin cleaner task every hour which automatically purges artifacts deleted more than 30 days ago. To change this default, change the **platform.recycleBin.timeout** property (listed in seconds) using the functionality described in "How to Manage System Settings" in the *Administration Guide*. To change the task settings, see "Administration Task Management" in the *Administration Guide*.

You can access the Recycle Bin from the Catalog Tab View menu.

The Recycle Bin displays a table of deleted artifacts with the following options:

- **Delete Permanently**

  Select artifacts and then click **Delete Permanently** to purge them from the repository.

- **Restore**

  Select artifacts and then click **Restore** to make them available again in the repository.

- **Empty Recycle Bin**

  Click **Empty Recycle** Bin to purge all the artifacts in the recycle bin from the repository.

Click an artifact name to open the Artifact Details page for the deleted artifact.

The normal icon for the artifact type is replaced with a wastebasket icon.

The context actions component contains options to **Restore** or **Delete Permanently**.

How to Use Bulk Operations

Many artifact management tasks may be performed in bulk as well as individually.

Wherever there is a table of artifacts (for example in list views, browse pages, and tables of search results) there are a set of context-sensitive options enabling bulk operations.

The set of operations varies depending on your role, the type of artifacts in the table, and the particular page.

The overall process for bulk operations is the same for each different action.

To Perform Bulk Operations:

1. In the table of artifacts, select the artifacts to perform the bulk operation on.

2. Select the bulk operation you want to perform:

   ■ **Edit**

   The owner and maintainers of an artifact or the administrator can edit various properties of the selected artifacts from the following options:

   ○ **Properties**

   Edit properties, categories, or keywords of the selected artifacts. For details, see "How to Edit Artifact Properties" on page 163, "How to Categorize Artifacts" on page 170, and "How to Edit Artifact Keywords" on page 177.

- **Change Owner**

    Change the person or group responsible for the artifact. For details, see "How to Edit Artifact Ownership" on page 179.

- **Change Access Rights**

    The owner of an artifact or administrator can edit the permissions of selected artifacts. For details, see "How to Edit Access Rights" on page 185.

- **Change Domain**

    Move the selected artifacts to another domain. For details, see "How to Edit Artifact Domains" on page 192.

- **Share / Unshare**

    Set read permission to the artifacts so that all users can view it. For details, see "How to Share Artifacts" on page 102.

- **Delete**

    Mark the selected artifacts as deleted and put them in the Recycle Bin. For details, see "How to Delete Artifacts" on page 194.

- **Synchronize**

    For artifacts that contain attached data, such as documentation, the synchronize action enables you to perform change management on the artifacts. For details, see "How to Synchronize Artifacts" on page 200.

- **Export**

    Export the selected artifacts to a file structure. For details, see "How to Export and Import Artifacts" on page 201.

- **Governance**

    Perform the following lifecycle governance actions on the set of selected artifacts:

    - **Start Governance**

        Enter the selected artifacts into a lifecycle process if applicable. For details, see "How to Start Governance" on page 247.

    - **End Governance**

        End the governance of the selected artifacts if applicable. For details, see "How to End Governance" on page 251.

○ **Set Lifecycle Process**

Change the lifecycle process, lifecycle stage, and force promotion without approval for the selected artifacts if applicable (administrator only).

3. These processes executes asynchronously. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

> **Note:** Stopping a task may take some time as all unprocessed artifacts are added to the final report. Stopping a task is also asynchronous so you can continue with other work. If you requested a notification it is sent on completion of the report.

How to Edit Artifacts

You can manually edit various aspects of artifacts individually or in bulk and the administrator can set up automatic actions to edit artifacts as a result of lifecycle changes.

> **Note:** Systinet limits access to edit functionality based on access rights. For details, see "Security and Access Control" in the *Administration Guide*.

> **Note:** Edit is not available for governed artifacts awaiting promotion or rollback approval.

For an individual artifact you can edit the following:

- Edit artifact properties from the Artifact Detail page Overview and Details tabs. For details, see "How to Edit Artifact Properties" on page 163.

- Edit artifact keywords using the Keywords component in the Artifact Detail page overview tab. For details, see "How to Edit Artifact Keywords" on page 177.

- Edit artifact specifications with HTML and rich-text support. For details, see "How to Edit Artifact Specifications" on page 178.

- Edit artifact categorization from the Artifact Detail page Categories tab. For details, see "How to Categorize Artifacts" on page 170.

- Edit the sub-artifacts from the Artifact Detail page Details tab. For details, see "How to Manage Relationships" on page 120.

- Edit the project the artifact belongs to from the Artifact Detail page Project component. For details, see "How to Use Projects" on page 127.

- Edit the contacts for the artifact from the Artifact Detail page Contacts component or Contacts tab. For details, see "How to Manage Contacts" on page 119.

To edit multiple artifacts, Systinet provides a set of bulk operations that you can access from the Edit context menu in tables of artifacts.

The **Edit** context menu provides the following options:

- **Properties**
  Opens an Edit Properties dialog box enabling you to edit the following.

  - **General** - Edit the general properties of the selected artifacts. For details, see "How to Edit Artifact Properties" on the next page.

  - **Categories** - Edit the categorization of the selected artifacts. For details, see "How to Categorize Artifacts" on page 170.

  - **Keywords** - Edit the keywords used to tag the selected artifacts. For details, see "How to Edit Artifact Keywords" on page 177.

- **Change Owner**

  Change the user, group, or role responsible for the artifact. For details, see "How to Edit Artifact Ownership" on page 179.

- **Change Access Rights**

  The owner of an artifact or administrator can edit the permissions of selected artifacts. For details, see "How to Edit Access Rights" on page 185.

- **Change Domain**

  Move the selected artifacts to another domain. For details, see "How to Edit Artifact Domains" on page 192.

- **Share / Unshare**

  The owner of artifacts or the administrator can share or unshare the selected artifacts. For details, see "How to Share Artifacts" on page 102.

Every edit of an artifact creates a new revision of the artifact. You can use the Artifact Details page History tab to compare revisions. For details, see "How to Compare Revisions" on page 193.

As part of Lifecycle Processes, the administrator can set up Automatic Actions to edit artifacts when specific lifecycle conditions occur. For details, see "Automatic Actions" on page 242.

How to Edit Artifact Properties

You can edit artifact properties individually or in bulk.

> **Note:** Edit is not available for governed artifacts awaiting promotion or rollback approval.

To Edit the Properties an Artifact:

1. From the Artifact Detail page Overview or Details tab, click the **Edit** context action.

   The Edit Artifact page opens containing the same details as the relevant Create Artifact page. For details, see the relevant UI Reference topic:

Content Artifacts	Address Book Artifacts
"Create Application Page" on page 140	"Create Organizational Unit Page" on page 146
"Create Business Process Page" on page 141	"Create Contact Page" on page 141
"Create Endpoint Page" on page 143	
"Create File Endpoint Page"	
"Create SOAP Service Page" on page 151 "Create XML Service Page" on page 152	
"Create SOAP Operation Page"	
"Create Project Page" on page 147	
"Create Service Page" on page 148	
"Create SLO Page" on page 149	
"Create REST Service Page"	
"Create HTTP Request Page"	
"Create HTTP Response Page"	
"Create File Transfer Page"	

> **Note:** The set of artifacts available depends on your role and permissions.

2. Click **Save** to apply your changes.

The Administrator can assign policies to validate artifact editing. The policy informs you if the edit is invalid or incomplete. For details, see "How to Manage Artifact Form Validation" in the *Administration Guide*.

To Edit Properties for Multiple Artifacts:

1. In a table of artifacts, select the artifacts to Edit.

2. Open the **Edit** context menu, and select **Properties** to open the Edit Properties dialog.

3. Select the **General** tab.

4. Select the properties to edit using the Add Property drop-down list.

5. Set a value for each selected property.

6. Click **Apply** to set the properties for the selected artifacts.

This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description
100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet.This prevents the client from timing out and assuming the request was lost.

200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.
226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.
301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.

306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" below below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.
406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.
409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.

414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.
422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.
428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.

444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.
499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.

502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products
Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain

Content-Type	multipart/form-data	Administrator	Top-level Domain

How to Categorize Artifacts

Systinetenables you to apply classification categories to artifacts making it easier for users to find relevant artifacts..

The Artifact Details page Categories tab displays the list of categories applicable to the artifact.

Screenshot: Artifact Details Page Categories Tab
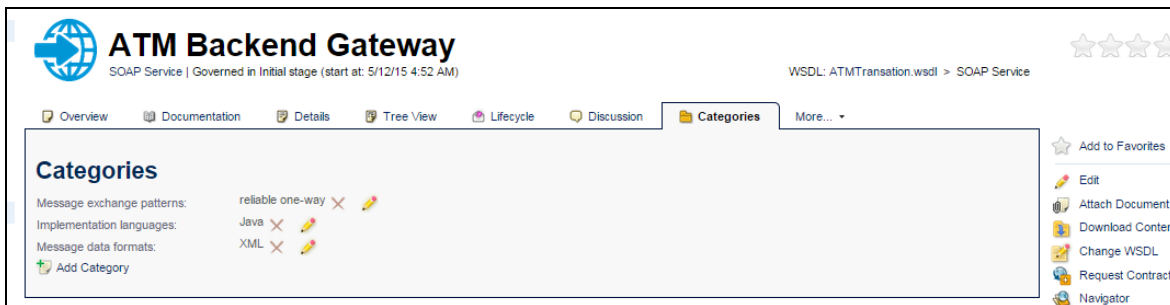
To Categorize an Artifact:

1. Open the Artifact Details page and select the Categories tab to display the list of categories applicable to the artifact.

2. In the Categories component, do any of the following:

   - To add a category from a new taxonomy. click **Add Category** to open a list of categories. Select a category to open the list of values. Select the values and click **OK** to add those category values to the artifact.

   - To add a value from an existing category, click ⊞ for the applicable category to open the list of values. Select the values to add and click **OK** to add those category values to the artifact.

   - To remove a category value, click the red **X** next to the value.

   > **Note:** The Categories component functionality is also available when you create or edit an artifact. For details, see "How to Create Artifacts" on page 117 and "How to Edit Artifact Properties" on page 163.

Alternatively, you can apply a categorizations to a set of artifacts.

To Categorize a Set of Artifacts:

1. In a table of artifacts, select the artifacts to Categorize.

2. Open the Edit context menu, and select **Properties** to open the Edit Properties dialog box.

3. Select the categories to apply to the selected artifacts:

   a. In the Categories tab, click **Select Category** to open the list of taxonomies.

   b. Select a taxonomy to open the list of categories.

   c. Select the categories to apply and click **OK** to add the categories to the Edit Properties Categories tab.

4. For each selected category, select whether to **Add** or **Remove** it from the selected artifacts.

5. Expand Advanced Options and do any of the following:

   ▪ Select **Include Subartifacts** to apply the changes to related sub-artifacts.

   ▪ Select **Remove Existing Categories** to replace any existing categories with the new selection.

   ▪ Select **Remove Existing Keywords** to delete any keywords applicable to the selected artifacts.

6. Click **Apply** to make your changes.

This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description

100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet.This prevents the client from timing out and assuming the request was lost.
200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.
226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.
301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.

303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.
306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" below below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.
406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.
409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.

411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.
414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.
422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.
428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.

431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.
444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.
499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).

500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.
502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products

Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain
Content-Type	multipart/form-data	Administrator	Top-level Domain

How to Change Artifact Icon
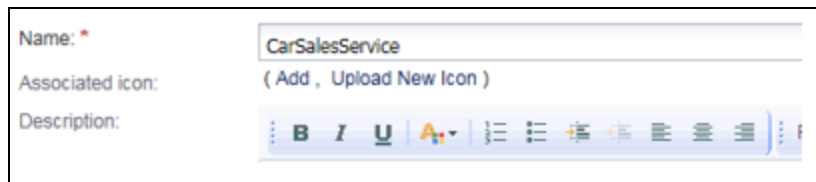
You can change the icon of an artifact of your preference after it is created.

To Change Artifact Icon:

1. Click **Edit** to open artifacts editing page.

2. In **Associated Icon** tab:

   ▪ Click **Add** to add an existing icon, or

   ▪ Click **Upload New Icon** to add a new icon from local machine or from URL.

   The name of icon artifact is displayed at **Associated Icon**.

3. Click **OK** to save changes, or click **Cancel** to ignore changes.

How to Edit Artifact Keywords

In addition to using categorizations to classify artifacts, you can add keywords as custom tags for single or multiple artifacts.

To Set Keywords for an Artifact:

1. In the Artifact Details page for the artifact you want to edit, open the Overview tab.

2. In the Keywords component, click **Edit** to open the Edit Keywords dialog box.

3. Type in your keywords. As you type your keywords, Systinet returns a dynamic set of existing keywords. Either select an item from this list, or complete your keyword. Separate multiple

keywords with commas.

4. Click **Save** to set the keywords for the artifact.

To Set Keywords for Multiple Artifacts:

1. In a table of artifacts, select the artifacts to Edit.

2. Open the **Edit** context menu, and select **Properties** to open the Edit Properties dialog.

3. Select the **Keywords** tab.

4. Click **Add** to open the Select Keywords dialog.

5. Type in your keywords. As you type your keywords, Systinet returns a dynamic set of existing keywords. Either select an item from this list, or complete your keyword. Separate multiple keywords with commas.

6. Click **Select** to add the Keywords to the Edit Properties Keywords tab.

7. For each Keyword, select whether to **Add** it or **Remove** it from the selected artifacts.

8. Expand **Advanced Options** and do any of the following:

   ▪ Select **Include Subartifacts** to apply the changes to related sub-artifacts.

   ▪ Select **Remove Existing Categories** to replace any existing categories with the new selection.

   ▪ Select **Remove Existing Keywords** to delete any keywords applicable to the selected artifacts.

9. Click **Apply** to set the keywords for the selected artifacts.

How to Edit Artifact Specifications

The artifact description in artifact create and edit pages is limited by the display length available in the Overview tab. In certain situations, you may want to have an unlimited description for some artifacts. To enable this, Systinet provides an Artifact Specification property with support for rich-text and HTML.

To Edit Artifact Specifications:

1. Open the Artifact Details page for the artifact you want to edit and select the **Specification** tab.

2. Click **Edit Specification** to open the rich-text editor for the Artifact Specification.

3. Use the input area and rich-text formatting buttons to enter your artifact specification or click

Source to view and edit the HTML for the specification.

4. Click **Save**.

How to Edit Artifact Ownership

The owner of an artifact or the administrator can change the ownership of artifacts to other users, groups, or roles.

To Change Ownership of Artifacts:

1. From a table of artifacts, select the artifacts to change, and open the **Edit** context menu, and select **Change Owner** to open the Change Owner dialog box.

2. Click **Change** to open the Select User dialog.

3. In the Select User dialog, use the Roles, Users, or Groups tabs to locate and select the new owner.

> **Tip:** Use the search dialog to find a particular entry in the currently selected tab.

4. Click **Select** to return to the Change Owner dialog box.

5. In the Change Owner dialog box, expand **Advanced Options** to view the following options:

   - Select **Include Subartifacts** to change the ownership of sub-artifacts in the same aggregate. For details about aggregates, see "Aggregation" on page 154.

6. Click **OK** to change ownership of the selected artifacts.

This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description
100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet.This prevents the client from timing out and assuming the request was lost.
200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.
226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.

301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.
306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" below below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.
406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.

409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.
414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.
422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.

428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.
444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.

499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.
502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products
Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain
Content-Type	multipart/form-data	Administrator	Top-level Domain

As part of lifecycle processes, the administrator can define automatic actions to change the ownerships of artifacts as a result of a lifecycle transition. For details, see "Automatic Actions" on page 242.

How to Edit Access Rights

Access rights are read and write permissions. The owner and maintainers of an artifact or the administrator can change the access rights for artifacts based on users, groups, and roles.

To Edit Access Rights for a Single Artifact:

1. Open the Artifact Details page for the artifact you want to edit and select the **Access Rights** tab.

2. Do one of the following:

   ■ To add read only access, click **Add Read Only Access** to open the Add Read Only Access dialog box.

   ■ To add read/write access, click **Add Read/Write Access** to open the Add Read/Write Access dialog box.

3. *Optional:* Expand **Advanced Options** and select **Include Subartifacts** to also apply the additional permissions to sub-artifacts.

4. Click **Select Entities** to open the Select Entities dialog box.

5. In the Select Entities dialog box, click **Browse Address Book** and use the Roles, Users, or Groups tabs to locate and select the new principal to give access rights to.

   > **Tip:** Use the search dialog to find a particular entry in the currently selected tab.

6. Select the user, groups, and roles to add permissions to and click **Select** to return to the relevant Add Permission dialog box.

7. Click **Add** to add the selected permission to the selected users.

To Edit Access Rights for Multiple Artifacts:

1. From a table of artifacts, select the artifacts to change, open the **Edit** context menu, and select **Change Access Rights** to open the Edit Access Rights dialog box.

2. Add the Principals to edit. The principals are the specific users, groups, and roles whose access rights you want to change for the selected artifacts.

   a. Click **Add Principal** to open the Select Principal dialog box.

   b. In the Select Principal dialog box, use the Roles, Users, or Groups tabs to locate and select the new principal to give access rights to.

      Tip: Use the search dialog to find a particular user, group, or role.

   c. Click **Select** to return to the Edit Access Rights dialog box.

      Systinet adds the selected principal to the Edit Access Rights list.

   Repeat this step until all the principals to edit are in the Edit Access Rights list.

3. For each principal select whether to **Add** or **Remove** their Read and Write permissions.

4. *Optional:* Expand **Advanced Options** to view the following options:

   ■ To change the access rights for sub-artifacts that are part of the same aggregates, select **Include Subartifacts**. For details about aggregation, see "Aggregation" on page 154.

   ■ Select **Remove Existing Rights** to replace any existing artifact access rights with the new settings.

5. Click **OK** to confirm your access right changes.

This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description
100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet. This prevents the client from timing out and assuming the request was lost.
200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.
226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.

301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.
306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" below below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.
406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.

409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.
414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.
422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.

428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.
444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.

499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.
502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products
Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain
Content-Type	multipart/form-data	Administrator	Top-level Domain

As part of lifecycle processes, the administrator can define automatic actions to edit the access rights of artifacts as a result of a lifecycle transition. For details, see "Automatic Actions" on page 242.

Note: Permissions are cumulative. For example, if an individual user without write permission is a member of a group with write permission, the user inherits permission to edit the artifact. Similarly, a user with write permission who is a member of a group without permission retains permission to edit the artifact. For more details, see "Security and Access Control" in the *Administration Guide*.

How to Edit Artifact Domains

Artifacts in the Catalog belong to a single domain with restricted visibility in other domains. You can move artifacts to another domain. For more details, see "Domains Overview" in the *Administration Guide*.

To Move an Artifact to Another Domain:

1. In the Artifact Details page Overview or Details tab, click the **Change Domain** context action to open the Change Domain dialog box.

2. Select the Target Domain from the drop-down list.

3. *Optional* Expand Advanced Options and select from the following options:

   ■ Select **Include Subartifacts** to change the domain of sub-artifacts in the same aggregate. For details about aggregates, see "Aggregation" on page 154.

   ■ Select **Include Dependencies** to change the domain of parent artifacts in the same aggregate. For details about aggregates, see "Aggregation" on page 154.

   ■ Select the ownership of the artifact in the new domain using **Retain** to keep the same owner or **Set to** and **Select** to open the Choose New Owner dialog box.

4. Click **Change** to move the artifact and any selected related artifacts to the selected domain.

To Move a Set of Artifacts to Another Domain:

1. From a table of artifacts, select the artifacts to change, open the **Edit** context menu, and select **Change Domain** to open the Change Domain dialog box.

2. Select the Target Domain from the drop-down list.

3. *Optional* Expand Advanced Options and select from the following options:

   - Select **Include Subartifacts** to change the domain of sub-artifacts in the same aggregate. For details about aggregates, see "Aggregation" on page 154.

   - Select Include Dependencies to change the domain of parent artifacts in the same aggregate. For details about aggregates, see "Aggregation" on page 154.

   - Select the ownership of the artifact in the new domain using **Retain** to keep the same owner or **Set to** and **Select** to open the Choose New Owner dialog box.

4. Click **Change** to move the artifacts and any selected related artifacts to the selected domain.
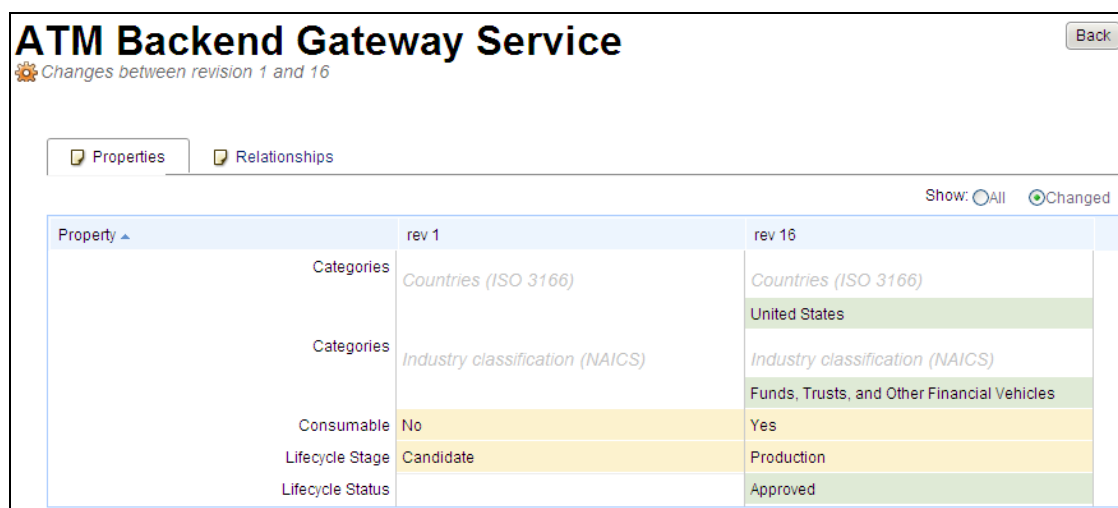
How to Compare Revisions

Systinet enables you to compare different revisions of artifacts.

To Compare Revisions:

1. In the Artifact Details page, open the History tab to view a list of revisions for the artifact.

2. Select the revisions you want to compare and click **Compare Revisions** to open the artifact changes page showing the differences between the selected artifact revisions.

   Screenshot: Artifact Changes Page

3. Use the artifact changes page to view the differences in the properties and relationships between

the revisions. Select to view only the **Changed** properties and relationships or **All** properties and relationships.

How to Delete Artifacts

You can delete artifacts individually or in bulk. Systinet places deleted artifacts in the "Recycle Bin" on page 158.

> **Note:** Deletion is not available for governed artifacts awaiting promotion or rollback approval.

To Delete Artifacts:

1. Do one of the following:

   - To delete multiple artifacts, from a table of artifacts, select the artifacts to delete, and click **Delete**.

   - To delete a single artifact, from the artifact detail view, click **Delete**.

   The Delete dialog box opens.

2. Depending on the artifact and your role, you can expand **Advanced Options** and select from the following options:

   - Select **Include SubArtifacts** if you want to delete related artifacts from the same aggregate (selected by default). For details about aggregates, see "Aggregation" on page 154.

   - Select **Ignore Usage** to delete the artifact even if there are incoming relationships from other artifacts in the repository. For example a SOAP service referenced from a Service.

3. Click **OK** to delete the selected artifacts.

This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing

list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description
100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet.This prevents the client from timing out and assuming the request was lost.
200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.

226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.
301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.
306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" below below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.
406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.

407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.
409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.
414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.
422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.

423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.
428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.
444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.

497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.
499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.
502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.

| 598 Network read timeout error | This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy. |
| 599 Network connect timeout error | This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy. |

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products
Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain
Content-Type	multipart/form-data	Administrator	Top-level Domain

How to Synchronize Artifacts

You can perform manual synchronization on artifacts that contain data content such as documentation or WSDLs. Systinet synchronizes data in the Catalog against external resources uploaded from remote URLs. For more details about publishing content, see "How to Publish Content" on page 127.

Data artifacts with attached content referenced by a remote URL contain a Synchronization component in the Artifact Details page Overview tab.

Screenshot: Artifact Details Synchronization Component
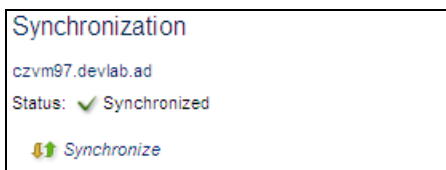
You can update changes to the remote document individually from this component or in bulk from tables.

To Synchronize an Artifact:

- In the Artifact Details page Overview tab, in the Synchronization component, click **Synchronize** and confirm your decision to update the copy of the data content in the server folder with the latest version of the external document.

To Synchronize Multiple Artifacts:

1. In a table of artifacts, select the artifacts that you want to synchronize.

2. Click **Synchronize** to open the Synchronize dialog box.

3. *Optional:* Expand **Advanced Options** and select whether to **Keep** local changes or **Overwrite** them with the external resource.

4. Click **Synchronize** to start the bulk synchronization operation.

This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

The same Synchronization component is available for artifacts imported from a UDDI Registry or from a UCMDB server with additional synchronization options. For details, see "How to Synchronize with Registries" on page 303 and "How to Synchronize Discovered Content" on page 330.

How to Export and Import Artifacts

Systinet enables you to export selected artifacts to a ZIP file for later import to the same or a different instance of Systinet.

You can export artifacts in bulk or individually. By default, export includes all sub-artifacts that are part of the same aggregations as the artifacts selected for export. For more details about aggregation, see "Aggregation" on page 154.

> **Caution:** The export / import functionality of the UI is not compatible with the command-line export / import tools available to the administrator. For more details about command-line export / import, see "Export Tool" and "Import Tool" in the *Administration Guide*.

To Export Single Artifacts:

1. In the Artifact Details page Overview or Details tab, click the **Export** context action to open the Export dialog box.

2. *Optional:* Change the name of the archive.

3. *Optional:* By default, Systinet exports all the sub-artifacts in the aggregation for the artifact. To export only the artifact, expand **Advanced Options**, and deselect **Include Dependencies**.

4. Click **Export** to start the export process.

5. Select a file location to store the exported archive.

To Export Sets of Artifacts:

1. In tables of artifacts, select the artifacts to export.

2. Click **Export** to open the Export dialog box.

3. *Optional:* change the name of the archive.

4. *Optional:* By default, Systinet exports all the sub-artifacts in the aggregation for the artifact. To export only the artifact, expand **Advanced Options**, and deselect **Include Dependencies**.

5. Click **Export** to start the export process.

6. Select a file location to store the exported archive.

> **Note:** The administrator can export the content of an entire domain and lifecycle processes. For details, see "How to Export Domain Content" and "How to Export Lifecycle Process" in the *Administration Guide*.

To Import Artifact Archives:

1. In the Catalog tab Import menu, click **Repository Archive** to open the Repository Archive Import page.

2. Select whether to import from a local file or a remote URL and input or browse for the file location or remote URL.

3. Select whether to **Preserve** or **Overwrite Data in the Repository**. If selected, the import overwrites the properties of existing artifacts with the properties defined in the archive.

4. *Optional:* Expand **Advanced Options** and select from the following options:

Advanced Option	Description
Conflicting Relationships	Select whether to **Merge** or **Overwrite** relationship conflicts. Merge adds any new relationships for defined in the archive for existing artifacts to any existing relationships. Overwrite replaces relationships for existing artifacts with the new relationships defined in the archive.
Import Access Rights	Select whether to use the access rights defined in the archive. If selected, the new access rights overwrite any access rights for existing artifacts.

Advanced Option	Description
Import Lifecycle Status	Select whether to use the lifecycle statuses defined in the archive. If selected the import replaces the lifecycle process, lifecycle stage, and status of any existing artifacts.
Destination Domain	Select whether to import to the domains defined in the archive or to the current domain. If you select to use domains defined in the archive and the domain does not exist, the artifacts are not imported.
Artifact Ownership	Select whether to use ownership defined in the archive or to assign ownership of imported content to the current user.
Applies To	Select whether the ownership option applies to all artifacts or only new artifacts.
System Settings	Select which system settings to import if the archive is a domain archive.

Note: These options only take effect if you have the appropriate permissions.

5. Click **Import** to start the import process.

   This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

   The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

   Note: Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description

100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet.This prevents the client from timing out and assuming the request was lost.
200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.
226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.

301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.
306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" below below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.
406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.

408 Request Timeout	The server timed out waiting for the request.
409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.
414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.

422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.
428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.
444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.

494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.
499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.
502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.

508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products
Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain
Content-Type	multipart/form-data	Administrator	Top-level Domain

Note: HP recommends updating Oracle Database schema statistics after importing large amounts of data. Old statistics may impact the performance of some data queries. Consult your database administrator.

To Update Oracle Schema Statistics:

- Execute the following command:

    EXEC DBMS_STATS.GATHER_SCHEMA_STATS (ownname => '&1',no_invalidate => FALSE,options => 'GATHER');

This command does not require database admin privileges and can be run by the schema owner (ownname).

How to Generate Document report

Systinet enables you to generate document as a custom report from Artifact Details page. By OOTB, you can generate an API document report from details page of Application Interface artifact.

To Generate Document report for a single artifact:

1. Open the Artifact Details page for the Application Interface artifact you want to edit in overview/details tab.

2. Click **Generate Document** from context action at right menu.
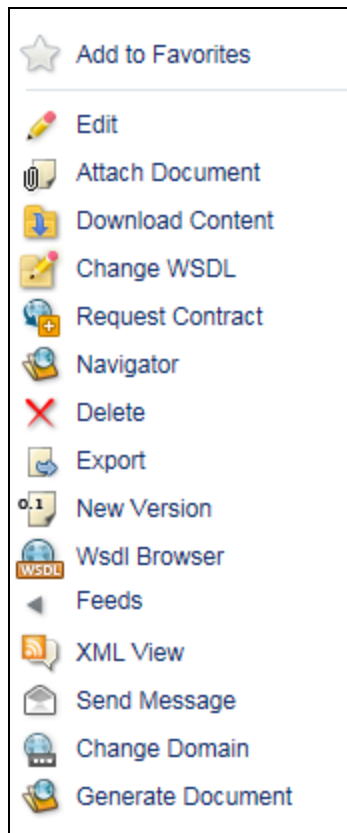
3. The API document report will be generated.

> **Tip:** You customize to generate your own custom report document type and also generate document report from details page of the artifact type that you want to have. For more details, refer to "How to Generate Document report action" topic in Administration Guide.

Definition Data Best Practice

HP recommends the following best practice for managing data definition files, such as WSDLs:

- **Relative References**

  HP recommends using relative references in definition data files whenever possible. With the appropriate repository workspace structure all links and imports are valid. When this is the case, you can use Systinet as master storage for your definition data files. For instance, you can use a WSDL compiler and other tools in order to develop services against WSDLs stored in Systinet.

  You should only use absolute references for third-party documents, such as W3C schemas. When you upload a document with an absolute reference then Systinet also stores the referenced document and you can use it in impact reports. You can find out which documents are dependent on absolutely referenced documents. For instance, when you upload `a.wsdl` that imports `http://hostname:port/schemas/a.xsd` the uploaded `a.wsdl` still refers to the remote XML Schema document, but an XML schema artifact is created for the remote document and you can run impact or dependency reports for this artifact and find out which artifacts are dependent on `http://hostname:port/schemas/a.xsd`.

- **Designing the Workspace**

  HP recommends appropriate design of the repository workspace. Even if you copy your directory structure, HP recommends the following principles:

  - **Shared / Enterprise Documents**

    Shared enterprise documents ready for reuse should exist in common folders. For instance, store all company core XML schemas in folder `/schemas` and referencing documents should reference these core schemas using relative links, such as `././../schemas/core.xsd`.

  - **Domains**

    The workspace should reflect the domain structure. The administrator can set a domain specific server folder when creating a domain or edit it in the Domain details page. The specified folder serves as the default server folder for publishing files in the domain.

  - **Organization Units**

    You can structure the workspace according to organization units or company departments. For instance, store all data definition files related to accounting in folder `/accounting` and store all documents related to marketing a `/marketing` folder.

  - **Versioning**

    The workspace structure should reflect the versioning of documents. Document version is very often related to the document namespace, but HP recommends using versions in file names and also in folder names. For example when you create a new version of `account.wsdl` then you

should change its namespace and its name (for example, `account_v2.0.wsdl`) and change its server folder (for example, `/accounting_v2.0`). For more details, see "Versioning Overview" on page 213.

For more details about the workspace, see "Server Folders" on page 114.

Chapter 7: Version Management

This part contains the following topics:

Concepts

- "Versioning Overview" below

- "Revisions" on the next page

- "Versions" on the next page

- "Lifecycle and Versioning" on page 215

Tasks

- "How to Create Versions" on page 216

- "How to Upload Versioned Data Files" on page 217

- "How to Compare Versions" on page 218

References

- "Versioning Best Practice" on page 219

Versioning Overview

Services and their associated artifacts sometimes need to be changed or modified to reflect changes in business processes, service dependencies, or other business criteria. Services are composed of metadata, documents, and implementation artifacts. Maintaining proper relationships between all these changing components can be challenging, so Systinet includes extended support for the versioning of services and artifacts. This support includes features such as automatic version resolution during file uploading, and easy version creation based on selected versioning strategies.

Systinet uses both revisions and versions and these different concepts are described in the following topics:

- "Revisions" on the next page

- "Versions" on the next page

There is a close relationship between lifecycle governance and versioning. For details, see "Lifecycle and Versioning" on page 215.

Systinet provides the following functionailty for creating and supporting versions:

"Versioning Best Practice" on page 219 describes HP recommendations for your versioning policy.

Revisions

Revisions are used to track changes to individual artifacts and content in the catalog. Systinet stores a complete history of each artifact instance as a revision. Whenever an artifact changes Systinet automatically creates a new revision. There is no versioning schema associated with the tracking of revisions.

You normally work with the latest revisions of artifacts in the UI, but Systinet provides the Artifact Details page History tab, enabling you to view and compare different artifact revisions and see the history of all changes. For details, see "Artifact Details Page History Tab" on page 47.

Versions

While revisions are used to track the history of changes to services and their underlying artifacts, Versioning is a different concept. All services and artifacts have version numbers assigned to them by Systinet when they are created, or imported to the Catalog. As users create new versions, the versioning process within Systinet implements automatic changes to version numbers for individual artifacts, groups of artifacts, or the entire service, based on the scope of the changes being made.

When a version is created in Systinet then a new branch (newly created artifact instance) is created. The properties of the new branch are automatically set according to the previous version. In other words, as a starting point, the new branch is essentially a clone of the previous version. HP recommends creating a new branch only when it is necessary to have more than one active version (artifact instance) at the same time. The typical example of this would be the need to have multiple implementations of the same service in production.
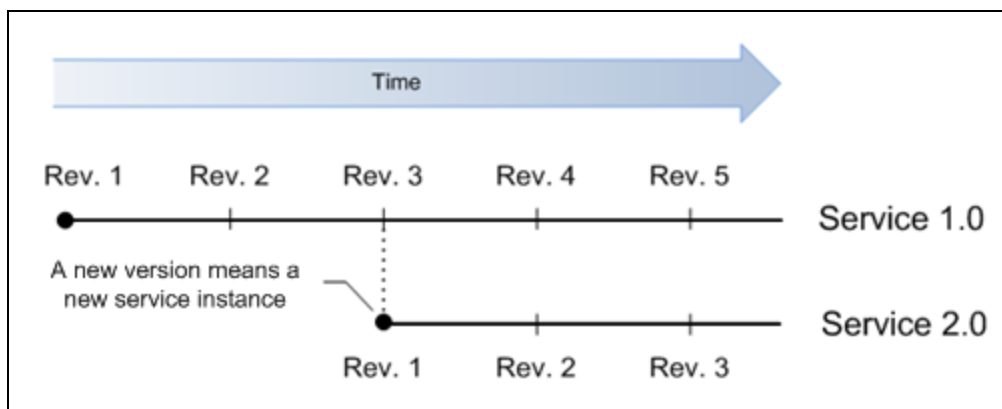
You can use the Version context actions in artifact detail pages for quick navigation between different versions (branches). For details, see .

Lifecycle and Versioning

Versioning is a natural consequence of using Lifecycle Governance.

Lifecycle Trees and Versioning

Artifacts that progress through a governance lifecycle together are grouped together in Systinet in a structure called a Lifecycle Tree. Lifecycle trees are defined by the user when setting up a lifecycle and can vary in how they are defined across different lifecycles.

Lifecycle trees are built around the concept of a Root artifact and Sub-artifacts, where there is a single root artifact for any lifecycle tree, with one or more sub-artifacts associated with that root artifact. For example, in the following diagram, the root artifact in the lifecycle tree structure is an Implementation, with sub-artifacts Endpoint, Operation, WSDL, Documentation, and SLO. The WSDL has an additional sub-artifact, XSD.
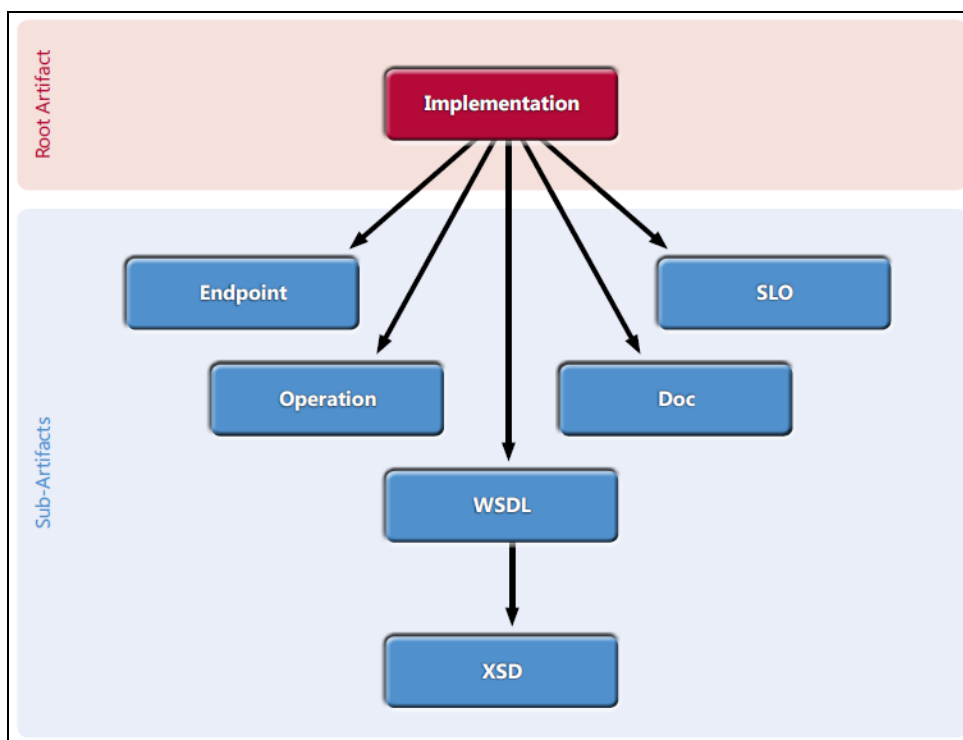
Because versioning involves potential changes to multiple artifacts in a lifecycle tree, there is a complex relationship between versioning and lifecycle. When a lifecyclee tree (root and sub-artifacts) is assigned to a lifecycle process, versioning is only allowed at the root artifact level; you cannot version sub-artifacts independently of the root artifact, and they always follow root artifact versions. This relationship between root and sub-artifacts in the lifecycle tree is tracked and understood by Systinet, so that the versioning relationship between the root and sub-artifacts is automatically maintained. Users who make changes to the version of a root artifact do not have to manually update the version numbers of the sub-artifacts.

However, not all services and artifacts are placed under governance. In this case, there is no lifecycle tree defined, hence no formal relationship between the different service artifacts. You can version any ungoverned artifact at anytime, but the versioning behavior is different. Unlike the example above, making a change to the implementation version of an ungoverned artifact does not automatically result in an update to the versions of the Endpoints, Operations, WSDLs, etc. The user must apply these changes manually if they want versioning to propagate across other artifacts.

Lifecycle Processes and Versioning

Lifecycle processes drive the versioning process. It is advisable to put artifacts under governance before versioning them, so that the versioning relationships across different artifact types can be maintained automatically by Systinet. This significantly reduces the complexity of managing versioning relationships.

In addition, HP recommends using appropriate lifecycle processes. For example, for the appropriate versioning of service implementations it is crucial to govern services and their implementations independently. For more details, see "Lifecycle Overview" on page 238.

How to Create Versions

Artifacts that progress through a lifecycle together are grouped into a structure called a *Lifecycle Tree*. For example, the root artifact in this structure (the one which controls the lifecycle stage of the others) could be *Implementation*, with sub-artifacts *Endpoint*, *Operation*, *WSDL*, *XSD*, *Documentation*, and *SLO*. For more details about Lifecycle Trees, see "Lifecycle Tree" on page 239.

HP recommends applying versioning to the root artifact and allowing Systinet to propagate the version information to the sub-artifacts as a top-down approach to versioning.

New Version functionality is accessible as a context action in Artifact Detail page Overview and Details tabs.

To Create New Versions:

1. In the Artifact Detail page Overview or Details tab, click the **New Version** context action to open the New Version page.

2. Specify the new version number according to the requirements of the version format.

3. *Optional:* For governed artifacts containing sub-artifacts governed by the same lifecycle process,

expand **Advanced Options** and select from the following options:

Versioning Advanced Options

Parameter	Definition
Create Blank Artifact	By default, Systinet copies the properties of the previous version to the new version, creates new versions of sub-artifacts in the lifecycle tree, and reuses sub-artifacts that are outside the lifecycle tree (in other words, copies relationships from previous versions). Select **Create Blank Artifact** to create an empty instance of the root artifact. For example, to create a new version of an implementation without copying properties or creating new versions of Endpoints, Operations, and WSDLs, check this option.
Lifecycle Process	Select a process from the list of applicable lifecycle processes for the artifact type, to manually select a lifecycle process to apply to the new version.

4. Click **Save** to create the specified new version of the artifact.

The versioning process also verifies the consistency of the existing versions of artifacts in the lifecycle tree. If the existing artifacts in the lifecycle tree do not all have the same version, the process offers the following options:

- Change the version of all artifacts to match the root artifact.

- Remove the non-matching sub-artifacts from the lifecycle tree.

> **Note:** The administrator, owner, or maintainer of an artifact can modify the version property without creating a new artifact. Change the version attribute in the same way as other attributes (for example, description) when you edit an artifact. For details, see "How to Edit Artifact Properties" on page 163. HP does not recommend using changing this property unless absolutely necessary. Instead lifecycle governance enables you to track changes to artifacts in governance. For details, see "Lifecycle Overview" on page 238.

The administrator can define the impact of the propagation of versioning to sub-artifacts. This enables the propagation of versioning to filenames, folders, or namespaces (where appropriate). By default, Systinet propagates versioning to the filename of attached data content (for example, `readme1.0.wsdl`) and for WSDL and XML Schema documents to the target namespace (for example, `http://atm/v1_0`). For more details, see "Configuring Versioning" in the *Administration Guide*.

How to Upload Versioned Data Files

HP recommends the top-down approach to versioning described in "How to Create Versions" on the previous page as it requires fewer manual changes to data content and is less error-prone. However,

Systinet does offer a bottom-up approach to versioning.

Systinet enables users to upload their data files into the Catalog. When uploading, users specify the files to be uploaded and the server folder (a location in repository workspace, where the document is available). Systinet parses the file, processes imports (referenced documents) and creates appropriate artifacts accordingly. For details, see "How to Publish Content" on page 127.

Managing versioning by uploading data files is useful when Systinet is not the master data storage. In such cases versioning is primarily handled outside Systinet and when data files are uploaded to the repository, Systinet automatically handles versions, and the setting of version and server folder attributes.

Systinet distinguishes versions and revisions of definition data files according to their file names. Revisions have the same location in the repository workspace whereas versions have different locations. When you publish a new revision of a data file to the same server folder, Systinet creates a new revision and updates the corresponding artifact. On the other hand, when you upload a new version then Systinet creates a new branch (artifact instance).

In the following example, WSDL document `account1.0.wsdl` with namespace `http://acme.com/account/v1.0` already exists in the repository. You change the document (for example, add a documentation element) but do not change the file name. When you upload the WSDL document to the same server folder then the existing WSDL artifact is updated and a new revision is created. On the other hand, when you upload the WSDL document with a changed file name `account2.0.wsdl` (or to a different folder) and change the namespace to `http://acme.com/account/v2.0` then Systinet creates a new WSDL artifact for a new branch.

Systinet automatically sets the version attribute of artifacts set according to the version in the data file. Versions of definition data files are usually specified in the document target namespace. Systinet enables the administrator to define which part of the namespace to treat as the version.

When you upload a WSDL document you can decide whether to create service and implementation artifacts together with data artifacts (for example, WSDL and XSD artifacts). If so, then Systinet checks whether the implementation and service exist in the repository and, if they do not exist, creates them during upload. However, if they do exist, they are reused.

In other words, when you upload a new WSDL document with a new file name and a changed namespace but with the same service name, then Systinet automatically creates a new *Implementation* artifact and associates it with the existing *Service* artifact.

How to Compare Versions

Systinet enables you to compare different versions of artifacts.

To Compare Versions:

1. In the Artifact Details page Overview tab for a version of the artifact you want to compare, click the version number to open the version history dialog.

Screenshot: Artifact Details Page Overview Tab Version History
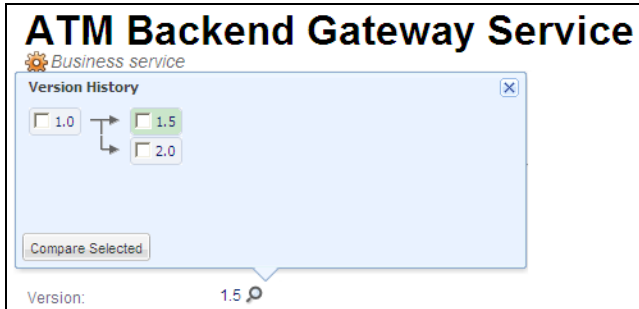
2. Select the versions you want to compare and click **Compare Selected** to open the version changes page showing the differences between the selected artifact versions.

Screenshot: Version Changes Page

3. Use the version changes page to view the differences in the properties and relationships between the versions. Select to view only the **Changed** properties and relationships or **All** properties and relationships.

Versioning Best Practice

Although Systinet provides with many versioning features, this section describes the recommended scenarios for creating new versions for the following most commonly used artifact types:

- "Service Versioning" on the next page

- "Implementation Versioning" on the next page

- "Data Artifact Versioning" on page 222

Note: This section describes best practice in terms of a single user, in practice you must take into account many users, such as developers and architects, acting on the same artifacts and documents. The examples in this section match the default Systinet versioning strategy. The administrator can change the strategy or even create their own. For details, see "Configuring Versioning" in the *Administration Guide*.

Service Versioning

Service artifacts represent business functionality (or concepts) from the real world. Service is an abstract artifact and so HP does not recommend creating new versions of services. Instead, HP recommends managing versions at the implementation level, because *Implementation* artifacts represent specific service versions.
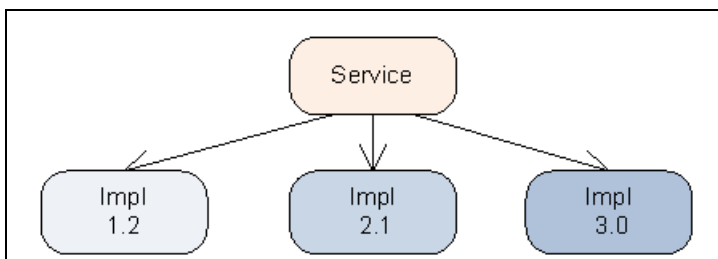
The diagram represents a *Service* artifact with three different *Implementations*. For details about creating new versions of service implementations, see "Implementation Versioning" below.

Implementation Versioning

Implementation artifacts represent a package of service version executables which can be deployed to multiple environments. Versioning of implementations that represent specific service versions is crucial for application governance.

Systinet provides the following options for creating a new implementation of a service implementation:

- "Using the Versioning Wizard" below

- "Uploading a Changed WSDL" on the next page

Using the Versioning Wizard

You can create a new service implementation version and then change the WSDL according to your requirements. In contrast to uploading data files, Systinet automatically changes the namespace, file names, and server folders.

To create a new service implementation version:

1. Create a new branch of a service implementation.

   In the service implementation detail page, run the versioning wizard. Specify the New Version field and leave all other options as default.

   Systinet creates a full service implementation as a new version associated with the same service as the current version. Systinet copies all basic properties of the current version to the new version and creates new versions of all sub-artifacts (Operations, Endpoints, and WSDLs) and

attaches them to the new branch. Systinet changes the version part of the target namespace in the WSDL document associated with the new WSDL artifact according to the specified new version and changes the file name (for example, `accounting/account1.0.wsdl` changes to `accounting/account2.0.wsdl`).

Note: The creation of new versions of sub-artifacts only applies to artifacts governed by a lifecycle process.

2. Change the WSDL.

Download the new WSDL and modify it according to your requirements, and then upload it back to the same server folder. The namespace, filename, and server folder do not require any changes as a result of the versioning performed in the previous step.

Uploading a Changed WSDL

You can create a new service implementation version by uploading a changed WSDL document.

To create a new version by uploading a WSDL:

1. Download the old WSDL containing the current service description in one of the following ways:

   ■ Use the **Download WSDL** functionality in the Implementation Detail page to download the WSDL and all referenced documents as a ZIP archive.

   ■ Use a WebDav client to obtain the WSDL from the server folder.

   ■ Use Systinet Plug-In (Eclipse or Visual Studio) to download the WSDL from the repository.

2. Change the WSDL.

   Modify the WSDL according to your requirements, for example, by adding a new operation. As part of the change, modify the service version namespace and the WSDL file name.

3. Upload the changed WSDL in one of the following ways:.

   ■ Use the Systinet publishing functionality to upload the WSDL into the appropriate server folder. Put all the changed documents into a ZIP and upload the archive, or upload them one-by-one.

   ■ Use a WebDav client to put the WSDL into an appropriate server folder.

   ■ Use Systinet Plug-In (Eclipse or Visual Studio) to upload the WSDL.

   HP recommends using the original server folder.

   Systinet creates a new WSDL artifact, artifacts for referenced documents (other WSDL and XSD artifacts), and an Implementation artifact. The version is set according to the document namespace in all created artifacts. Moreover, the Implementation artifact is associated with the existing Service artifact and so the result is a Service artifact with several Implementations (versions).

Data Artifact Versioning

Data artifacts refer to all artifact types that can have attached data content. These artifacts are normally a result of uploading data files to the repository. For example, WSDL, XSD, DTD, XML, XSLT, and Documentation are all data artifact types.

The versioning of data artifacts is different for different artifact types. For instance, versioning of XML-based artifact types (for example, WSDL and XSD artifacts) must handle document target namespaces, file names, and server folders, whereas versioning of Documentation artifacts must handle only file names. When non XML-based artifacts contain references to other documents (for example, DTD artifacts), then versioning must control the server folder together with the file name.

The versioning of data artifacts is similar to the versioning of implementations described in "Implementation Versioning" on page 220. You can create new versions with the versioning wizard or by uploading changed data files. HP recommends using the versioning wizard to remove the need for manual versioning. The wizard helps you to set versions, namespaces, file names, and folders according to HP Software best practice.

Chapter 8: Contract Management

This part contains the following topics:

Concepts

-

-

Tasks

-

-

-

References

-

Contracts Overview

A key service relationship is the provider-consumer relationship that you can create between service consumers and providers.

There are several reasons to capture provider-consumer relationships in Systinet:

- **Change Management**

  By requiring consumers of services to register, the provider ensures that all users of that service are notified whenever some aspect of it changes.

- **Contract Access Control**

  A controlled way of providing consumers access to a service, helping prevent unanticipated or inappropriate usage.

- **Contract Monitoring**

  Monitors how successful a particular service is, based on its level of reuse and its use across organizational units.

- **Contract Auditing**

Providers need to know who is using specific services and who granted them access. This could be for many reasons, such as security, accounting, capacity planning, or change management.

Contract management controls the contract of services by defining the following:

- **Consumer**

  The consumer is the person or organizational unit that uses a service.

- **Provider**

  The provider is the user or organizational unit responsible for contracts associated with the artifact that the consumer wants to use.

- **Contract Request**

  When a consumer finds an artifact that they want to use, they create a contract request which is submitted to the provider.

- **Contract**

  When a request is approved, a contract is formed between the consumer and provider artifacts.

- **Service Level Objectives**

  Each provider artifact can be associated with a set of offerings that the provider sets out as performance levels for the service. When a consumer makes a request they can select an SLO as part of the terms of the contract.

  There are two SLO artifact types:

  - Service Level Offering artifact

  - Service Level Objectives artifact

  Service Level Offering artifact:

  - It will be created by provider

  - It will be added into provider artifact

  Service Level Objectives artifact:

  - It is a copy of **Service Level Offering** artifact.

  - It will be created when consumer create a contract and add a SLO artifact for contract.

Contract Management consists of the following processes:

Contract Management fits onto the larger service provision use case as described in "Contract Workflow" on the next page.

Systinet provides a default set of artifact types that can be consumer or provider artifacts. The administrator can customize these settings. For details, see "Default Provider / Consumer Artifact Types" on page 235 and "Configuration Management" in the *Administration Guide*.

Contract Workflow

For a provided service to be consumed, there are typically, several steps required by the provider and a discovery process for the consumer. Having found a service that meets their requirements the consumer and provider can establish a contract. The "Contract Workflow" diagram displays the typical tasks that should be performed, and an overview of each step is given below.

> **Note:** The workflow uses a service as the example. Systinet enables other artifact types to be providers as well. For details, see "Default Provider / Consumer Artifact Types" on page 235.
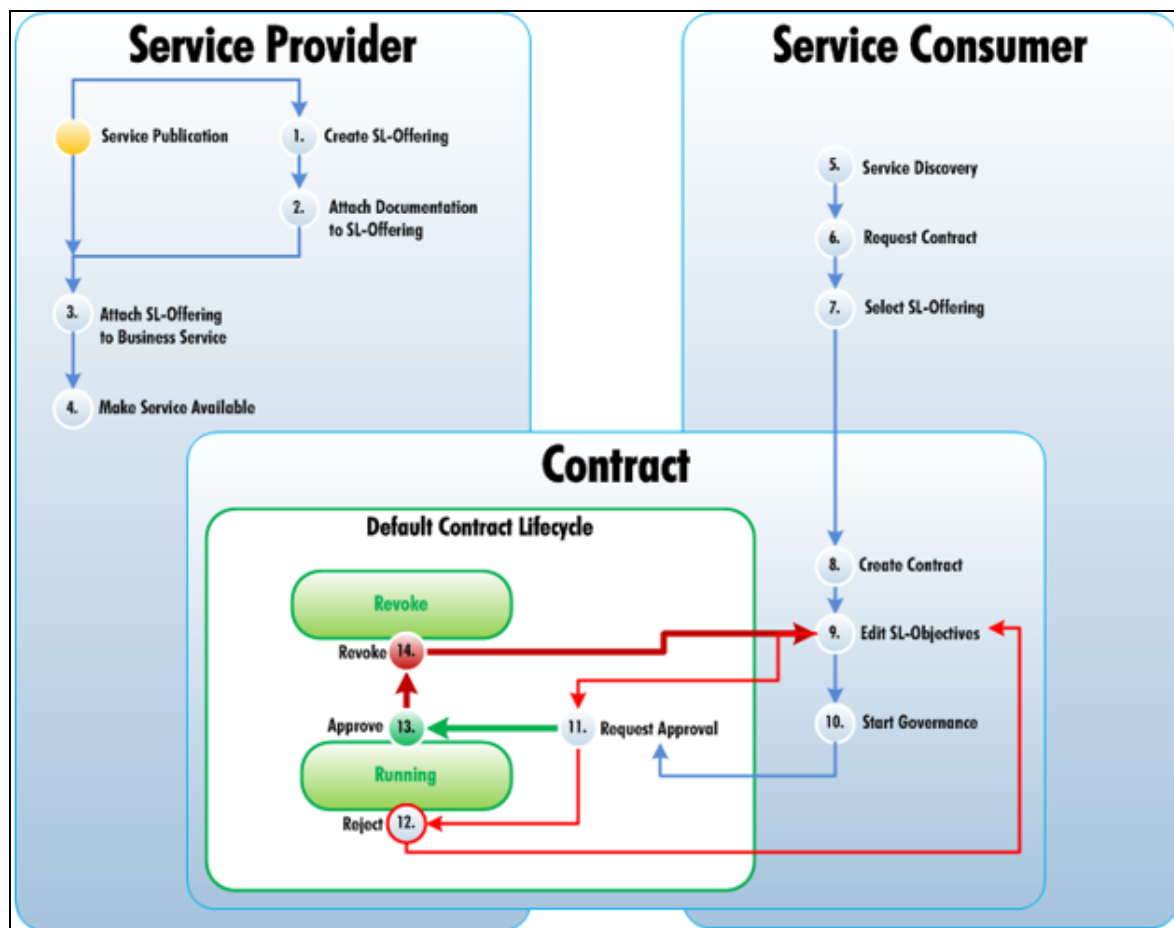
1. After you create a service, it is normal practice to offer a number of service-level objectives describing the terms of use for the service. You can create an SLO independently, or as part of a service. For details, see "How to Create Artifacts" on page 117 and "Create SLO Page" on

page 149.

2. You can also attach documentation to the SLO. For details, see "How to Attach Documentation" on page 123.

3. Attach the SLO to the Service. For details, see "How to Manage Relationships" on page 120.

4. Make the Service available by marking it as consumable. For details, see "How to Edit Artifact Properties" on page 163.

    Note: This can be automated as part of a lifecycle process. For details, see "Automatic Actions" on page 242.

5. The consumer browses Systinet to see which services are available and whether they meet their requirements. For details, see "Search and Browse Overview" on page 55.

6. After the consumer finds the service, they can request it. For details, see "How to Create Contract" on the next page.

7. When a consumer requests consumption of a service, you must either approve or reject the request. For details, see "How to Process Contract Requests" on page 229.

Contract Management as Lifecycle Process

Previous versions of Systinet used special artifacts SDM model to model contract management. Systinet 10 obsoletes part of contract management SDM and implements contract management functionality on top of more powerful and flexible lifecycle process to govern directly contract artifacts. Administrators can create new lifecycle process to govern contracts and customize stages, approvals, actions according to their corporate standards.

How to Set the Provider

Each consumable artifact should have a user or organizational unit, known as the *provider*, that is responsible for managing the contracts associated with the artifact.

Note: If no specific provider is set for an artifact, the artifact owner is the default provider.

To Set Artifact Providers:

1. Open the Artifact Details page for the artifact in the Overview tab.

2. In the artifact properties section, locate the Provider property and click **(Change)** to open the Select Provider dialog box.

3. Do one of the following:

- Type in the provider. Systinet returns a list of available providers (user profiles and organizational units). Select a provider from the list of results.

- Click **Browse Addressbook** to select a Person or Organization Unit from the local or global address books.

- Expand **Create Contact**, and select **Organizational Unit** or **Contact** to create a new contact to add as the provider for the artifact. For parameter details, see "Create Contact Page" on page 141 and "Create Organizational Unit Page" on page 146.

How to Create Contract

An artifact must make a contract request to the provider.

1. Do one of the following:

   Open the Artifact Detail page Overview or Details tab for the artifact you want to consume, and click the **Request Contract** context action to open the Create Request page with the Contracted Artifact set.

   In the Catalog page Main menu, click Request Contract to open the Create Request page without the Contracted Artifact set.

Screenshot: Create Request Page

The Create Request page displays a visual representation of the potential contract with the consumer or consuming artifact on the left and the provided artifact on the right.

2. *Optional:* By default, you are the consumer. To change the consumer, click **Change Consuming Artifact** and use the search component to select the consumer artifact.

   > **Note:** Only Contact artifact and Application Component artifact types can be set to Consuming Artifact by default. To change the list of Consuming Artifact types, go to Administrator > Configuration > System Settings > filter and change the value of **platform.cm.consumers** property.

3. Do one of the following:

   - If the Contracted Artifact is not set, click **Select Contracted Artifact** and use the search component to select the contracted artifact.

   - *Optional:* If the contracted artifact is set but you want to form a contract with a related sub-artifact, click **Select Sub-Artifact** and select from the hierarchy of sub-artifacts.

     Tip: HP recommends consuming a particular implementation of a service as they represent a particular instance of the service.

   - *Optional:* If the contracted artifact is set and you want to change it, click **Change Contracted Artifact** and use the search component to select the contracted artifact.

   > **Note:** Only Application Interface, Endpoint, and Operation artifact types can be set to Contracted Artifact by default. To change the list of Contracted Artifact types, go to Administrator > Configuration > System Settings > filter and change the value of **platform.cm.providers** property.

4. *Optional:* If required, expand **Select SLO** and select from the available SLOs to use to determine the agreed set of performance criteria.

5. *Optional:* Enter a message for the service provider.

6. *Optional*: Enter Customer Identification as IP addresses or a SubjectDN and customize Layer7 template in order to use Client IP Address/Client Certificate identity retrieval methods. For more information, see "Client IP Address/Certificate Retrieval Method".

7. *Optional:* Attach documentation to your contract request. For details, see "How to Attach Documentation" on page 123.

8. Review the request details and click **Save** to place your contract request.

Systinet notifies the provider of your request and adds it to your Pending Requests list.

The provider of the service either approves or rejects your request. The result of their decision is sent as a notification email and updates in the Events component. An approved request appears in the Subscriptions component.

HP Software also provides a set of IDE plugins with integrated request contract functionality.For details, see *Plugin for Eclipse* and *Plugin for Visual Studio*.

How to Process Contract Requests

This action is only executed by users have admin or domain admin role.

To Process Contract Requests:

1. In the Catalog tab Main menu, click **Subscriptions** to open the list view of contracts and requests that you are involved in.

2. In the Providing section click the request name with a **Pending Request** status to open a view of that request.

> **Tip:** Pending requests also appear in your Events and Task lists if you are a contract approver. Click the event or task name to open the request details.

3. Use **Bulk Operation** or **Set Lifeycle Process** action to change contract's state.

When consumers create a contract then the contract will be automatically started governance with "**Default Contract Lifecycle**" and contract state will be controlled by lifecycle process

To process contract state:

1. Consumer creates a contract.

2. Consumer request for approval. Systinet sends a notification e-mail.

3. Maintainers of artifact provider will view the "**request approval**" from "**My Tasks**" page or "**Pending Tasks**" section.

4. The provider approves the lifecycle request then contract state will be changed to "**Accepted**" state. Systinet sends a notification e-mail. Stage will be moved to "**Active**" stage.

5. If the provider rejects the lifecycle request then contract state will be changed to "**Rejected**" state. Systinet sends a notification e-mail. If contract is rejected then consumer should update contract and re-request approval.

6. The provider moves to "**Revoked**" lifecycle stage to revoke contract.

Default Contract Lifecycle

This stage is created during installation of Systinet.
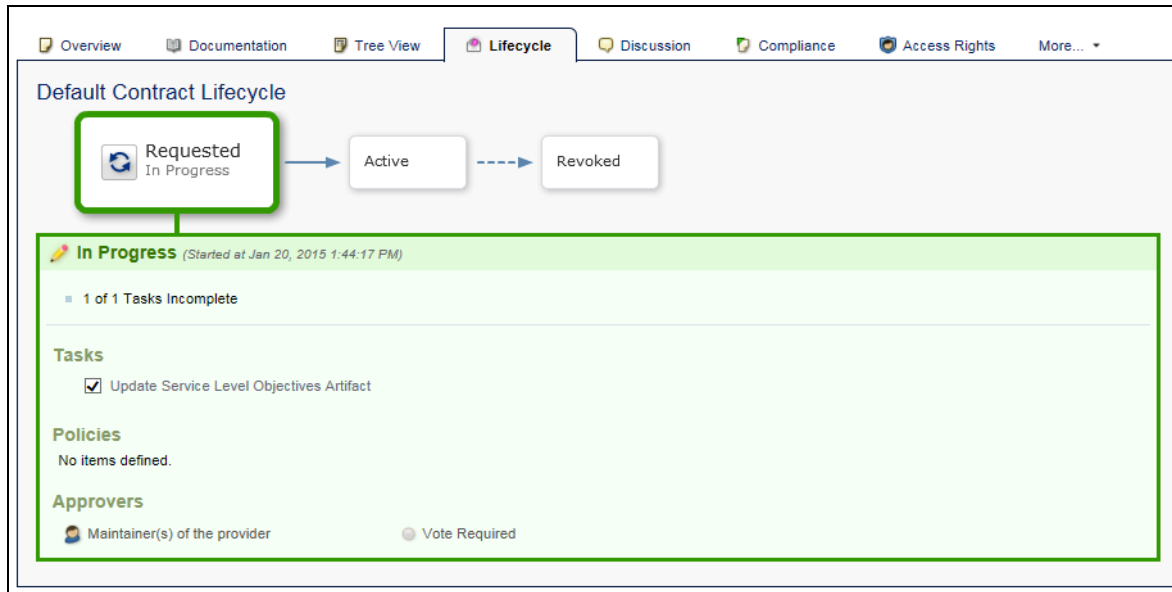
Default Contract Lifecycle lifecycle has three stages:

1. **Requested**

2. **Active**

3. **Revoked**

Requested stage includes:

- **In Progress**: List of tasks which is incomplete.

- **Approvers list:** Maintainer(s) of the provider list includes owner and users, roles or groups who have write permission on artifact provider.

- **Automatic actions are:**

  - **Change Contract Status** action: Applicable when Entering stage and Set Status/Pending

  - **Change Contract Status** action: Applicable when Approval succeeded and Set Status/Accepted

  - **Change Contract Status** action: Applicable when Approval failed and Set Status/Rejected

  - **Auto Request/Approval Stage** action: Applicable when Entering stage

  - **Notify** action: Applicable when Approval succeeded, Recipients/ Owner , Maintainers , Contacts (Provider, Technical Support, Architect, Developer, Administrator, Business Expert, Operations Manager) , Consumers , and Providers.

  - **Notify** action: Applicable when Approval failed, Recipients/ Owner , Maintainers , Contacts

(Provider, Technical Support, Architect, Developer, Administrator, Business Expert, Operations Manager) , Consumers , and Providers.

Active stage includes the **Tasks** and **Approvers list**.

Revoked stage includes the **Tasks** and **Approvers list**.

Client IP Address/Certificate Retrieval Method

The following instructions introduces how to implement **Client IP Address** and **Client Certificate** Identity Retrieval methods.

This helps to customize the consumed artifact in order to add one more property. The consumer will be asked to provide the specific information/identification when requesting a contract.

1. The consumer enters the IP address or the field name in certificate (e.g. SubjectDN) into Consumer Identification when creating a contract.

2. Create **Contract ID Management** artifacts for Client IP Address and Client Certificate.

   - **Client IP Address method**

- **Client Certificate method**

Contract ID Management

Name: *   Client certificate

Description: **B** *I* <u>U</u> **A** ... | ☰ ☰ ☰ ☰ | ☰ ☰ ☰ | For

Identity retrieval method: Client certificate ▾

ID message location:

3. Change L7 template.

   In the L7 proxy template file, we use FreeMarker to get consumed or provided artifact information. For example:

   - ${context['contract']['contractName']} returns the name of the contract artifact.

   - ${context[consumer]['consumerName']} returns the name of the consumer artifact.

     For more details see, "How to provide additional artifact properties to Layer7 server" on page 266

   Consumer specific information will be then used to validate with Client IP Address and Certificate clients. Client IP Address and Certificate clients can be got by using context variables or expressions of Layer 7 Policy Manager. For more detail, read the documentation of Layer 7 Policy Manager.

   Client IP Address method:

- List of IP Addresses:

```
<#if .globals.identity=='Client IP'>
  <L7p:ComparisonAssertion>
      <L7p:CaseSensitive booleanValue="true"/>
      <L7p:Expression1 stringValue="${.globals.rootArtifact.consumerIdentification}"/>
      <L7p:FailIfVariableNotFound booleanValue="false"/>
      <L7p:MultivaluedComparison multivaluedComparison="ANY"/>
      <L7p:Operator operatorNull="null"/>
      <L7p:Predicates predicates="included">
          <L7p:item dataType="included">
              <L7p:Type variableDataType="string"/>
          </L7p:item>
          <L7p:item binary="included">
              <L7p:Operator operator="CONTAINS"/>
              <L7p:RightValue stringValue="<#noparse>${</#noparse>request.tcp.remoteAddress}"/>
          </L7p:item>
      </L7p:Predicates>
  </L7p:ComparisonAssertion>
</#if>
```

- Range of IP Addresses:

Modify the script artifact **iPepAPI.js** to provision necessary properties to this template.

```
    consumerIdentification = contract.getConsumerIdentification();
    slashIdx = consumerIdentification.indexOf('/');
    networkMark = consumerIdentification.substring(slashIdx + 1);
    startIp = consumerIdentification.substring(0, slashIdx);
    ...
    var result = {device:{uuid:deviceUUID,name:deviceName},
                contract:{uuid:contractUUID,name:contractName,
                    networkMark:networkMark,startIp:startIp,
                    consumerIdentification:consumerIdentification},
                slo:{uuid:sloUUID,name:sloName},
                consumer:{uuid:consumerUUID,name:consumerName},
                provider:{uuid:providerUUID,name:providerName},
                identity:{uuid:contractIDManagementUUID,name:idManagementName},
                script:{uuid:scriptUUID,name:scriptLocation}
                };
    ...
```

For more details see,

```
<#if proxy.identityMethod=='Client IP'>
  <L7p:RemoteIpAddressRange>
      <L7p:NetworkMask intValue=
      "${context['contract']['networkMask']}"/>
      <L7p:StartIp stringValue=
      "${context['contract']['startIp']}"/>
  </L7p:RemoteIpAddressRange>
</#if>
```

When requesting a contract, set the Client IP range as follows:

Create Contract

Name: *	HP Systinet Administrator uses HelloService
Message:	**B** *I* U A· ≔ ≔ ⇥ ⇤ ≡ ≡ ≣ ┊ Font

Consumer Identification:	16.154.0.0/16
Using: *	

Client Certificate:

You can refer to Policy Assertion Reference > Access Control Assertions > Extract Attributes from Certificate Assertion in the help content of Layer 7 Policy Manager to know how to extract attributes of client certificate, and the use them to compare with **Consumer Identification** property.

When requesting a contract, set the Distinguished Name (DN) of client certificate as follows:

Default Provider / Consumer Artifact Types

By default the following artifact types are defined as providers and consumers:

Provider Artifact Types	Consumer Artifact Types
Application Interfaces • REST Services • XML Services • SOAP Services • File Transfers	Application Components • Applications • Components
Endpoints • HTTP Endpoints • SOAP Endpoints • File Endpoints	Contacts • Contacts • Organizational Units

Provider Artifact Types	Consumer Artifact Types
Operations • HTTP Requests • HTTP Responses • SOAP Operations	-

Note: Systinet controls the set of consumer and provider artifact types with a set of configurable properties. For details of these properties and how to change them see the Contract Management table in "System Configuration Properties" and "How to Manage System Settings" in the *Administration Guide*.

Chapter 9: Lifecycle Governance

This part contains the following topics:

Concepts

Tasks

Lifecycle Overview

Artifacts have several lifecycle stages ranging from being merely a candidate through development, implementation, and eventual deprecation and reuse. Each stage has own specific features and each organization has different detail requirements for these different stages. The stages can be divided into development and runtime stages and before a service can be allowed to move from one stage to another, all necessary policy requirements and approvals must be in place. See "Lifecycle Tree" on the next page for details.

In Systinet lifecycle processes are defined and given policy, task and approval requirements by an administrator. See "Lifecycle Process Management" in the *Administration Guide* for details.

These processes are then either automatically or manually applied to artifacts. See "Automatic Assignment" on page 246 and "Lifecycle Best Practice" in the *Administration Guide* for details.

Manual lifecycle tasks can be assigned to different users and can have policies that must be validated before task is completed. Policies as well as permissions can be associated with the lifecycle process stage. See "Lifecycle Tasks" on page 241, "Permissions" on page 243, and "Policy Management" in the Administration Guide for more details.

When all the requirements and tasks are complete, the artifact owner makes a request to move the process to the next stage. If the administrator has assigned approvers, they are notified and are required to vote on approval. Depending on transition type, the governed artifact moves to the next stage and the lifecycle automatic actions defined for these stages are triggered. See "Approvers" on page 243, "Stages and Transitions" on the next page, and "Automatic Actions" on page 242 for more details.

For procedural topics please see:

- "How to Start Governance" on page 247

- "How to Review/Close Lifecycle Tasks" on page 248

- "How to Review/Validate Policies" on page 249

- "How to Review Compliance Status" on page 249

- "How to Request Approval" on page 250

- "How to Approve/Reject Approval Requests" on page 251

- "How to View Lifecycle History" on page 251

- "How to End Governance" on page 251

Lifecycle Tree

Artifacts that progress through the governance lifecycle together are grouped in Systinet by a structure called a Lifecycle Tree. Lifecycle trees are defined by the user when setting up a lifecycle process and can vary in how they are defined across different lifecycle processes.

Lifecycle trees are built around the concept of a Root artifact and Sub-artifacts, where there is a single root artifact for any lifecycle tree, with one or more sub-artifacts associated with that root artifact. For example, the root artifact in the lifecycle tree structure below would be Implementation, whereas sub-artifacts are Endpoints, Operations, WSDLs, XSDs, Documentation, and SLOs.
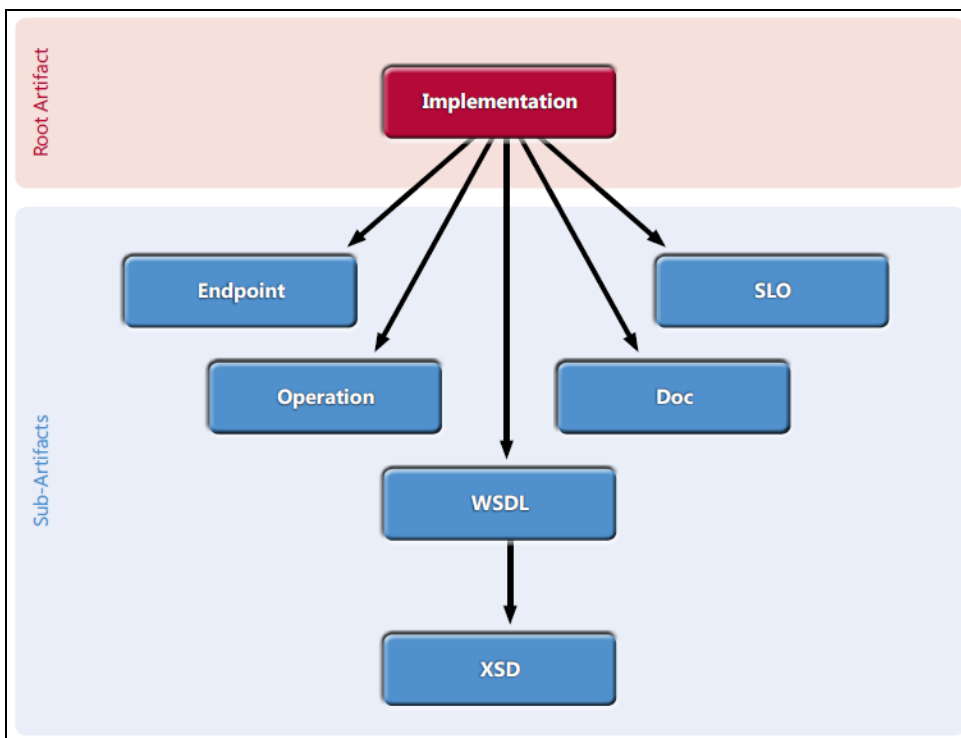
Stages and Transitions

A lifecycle consists of a set of stages and transitions between them.

The following image shows part of the lifecycle process:
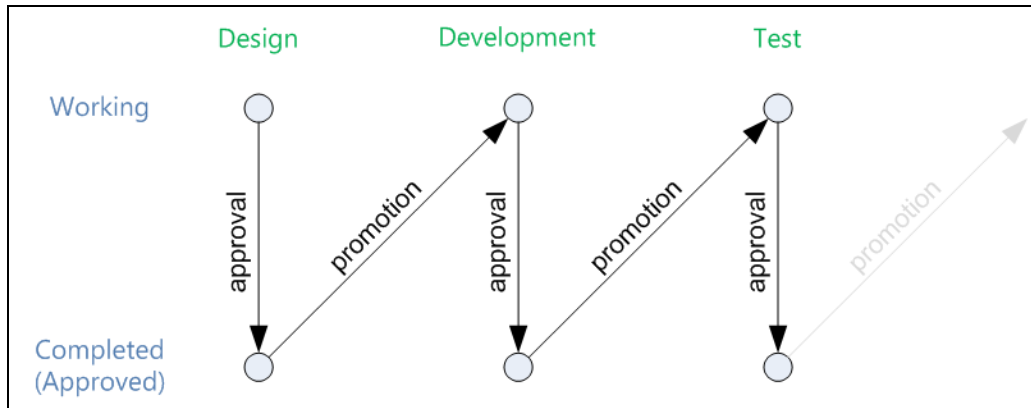
The current stage of a service in a lifecycle can be described and queried in the following ways:

- **Working Stage**

  The working stage represents the stage currently in progress. These are stages in which there may be tasks to be completed and/or policies to be validated before stage can approved for advancement.

  A working stage can have different status:

  - In Progress - Work is being performed on the stage.

  - Voting - Stage is waiting for assigned approvers to vote.

  - Rejected - Stage has been voted against being approved by an assigned approver.

  - Failed - An assigned and required policy constraint has not been met.

  See "Approvers" on page 243 and "Policy Management" in the *Concepts Guide* for more information.

- **Approved Stage**

  The approved stage represents the last completed stage in which all tasks and policies have been completed and has been approved.

Moving between stages involves the following lifecycle transitions:

- **Approval**

  The approval process is the confirmation that all the requirements for the current Working Stage are complete and results in the Approved Stage.

- **Promotion**

  Promotion initiates the next Working Stage. This requires the current Working Stage to be Approved and can be automated as part of the process definition.

There are different types of transitions that are described in "How to Define Transitions" in the *Administration Guide*. There may also be a set of automatic actions associated with a lifecycle events as described in .

Lifecycle Tasks

A task specifies the execution of an action against a lifecycle stage. These tasks are tracked and they must be marked as completed before starting the approval process.

Tasks may be assigned to users, groups or roles.

Lifecycle process is applicable to contract artifact, tasks can be assigned to contact. Task contact assignee includes:

- **Maintainer(s) of the provider** task contact assignee, this assignee is a list of users, roles, groups which have **WRTIE** permission on provider artifact. This list will be automatically filled when requesting approval and list's number required is one.

- **Contact role** task contact assignee includes Provider of the provider(s), Developer of the provider (s)", and so on and this assignee is an user who is a contact of provider of consumer artifact.

Examples of common tasks:

- Requesting that documentation is attached to an artifact such as business requirements

- Requests for deployment

- Requests for builds

- Requests for testing

Technical policies applicable to the root artifact may be associated with tasks Systinet automatically validates these policies when of a task is marked as complete. If the policy fails validation, the user is notified in the application and the task is not considered complete.

See "Lifecycle Policies" in the *Administration Guide* for more information about policies.

Lifecycle Policies

Lifecycle stages in a lifecycle process can have a set of policies associated with them that serve as validation checks for artifacts at that lifecycle stage. Policies can be optional or required and can also validate that tasks associated with the stage are complete. These policies are automatically validated when a user requests stage approval or can be manually validated by users. The collective status of these policies gives a Compliance Status for an artifact which measures the current percentage of required policies which pass validation. The administrator assigns policies to lifecycle stages during the creation of lifecycle processes. For details about setting up and performing lifecycle validation, see "How to Define Policies" in the *Administration Guide* and "How to Review/Validate Policies" in the *User Guide*.

Systinet includes a default Lifecycle Validation Task, scheduled to run once a day, which automatically validates all artifacts in governance against the policies that apply to their current lifecycle stage. For details about managing this task, see Administration Task Management.

Automatic Actions

At each lifecycle stage, you can define a set of Automatic Actions to modify an artifact when certain lifecycle events occur.

The available automatic actions are:

- **Execute Script**

  Invoke a custom javascript code previously defined using administration, customization, and manage scripts. In the Advanced tab, you can define a javascript fragment, which is executed prior to running the referenced script artifact.

- **Notify**

  Send automated mail notifications to sets of users as a result of a lifecycle change. For example, notify consumers that a new version of a service is approved in the Production stage and is available for consumption.

- **Deploy/Undeploy Contract**

  Deploy or undeploy a contract. Deploy action also updates an existing proxy if available. For example, you can deploy a contract to a Layer 7 server when a contract in the Production stage is approved.

- **Update Proxies**

  Rebuilds PEP proxies for contracts of the approved service or contract provider. For example, the proxies of contracts associated with the governed artifact will be recreated when a contract in the Production stage is approved.

- **Enable Consumption or Disable Consumption**

  Allow or disallow the creation of contracts for an artifact as a result of a lifecycle change. For example, set the Consumable flag when a Service in the Production stage is approved. This allows the artifact to be used in a contract and make it visible for the partner role.

- **Export to Registries or Delete from Registries**

  Either exports or deletes an artifact from registries. This would be used if registries contained artifacts for specific purposes i.e.: development, production, etc.. Once artifacts move from one stage to another, it may be desired that they be removed or added to different registries.

- **Parent Request Approval**

  Creates an approval request for a parent artifact. Allows the tying together of processes so that a parent implementation can have an automatic approval requested when the dependencies have been approved.

- **Remove Comments**

  Removes all comments from the artifact when moved to next stage. This may be used when there is some debate about functionality within the commentary, but once resolved, does not need to be retained when moving to the next stage.

Approvers

At each lifecycle stage, you can define a set of approvers whose votes are required to approve the current stage. These votes can be assigned to users, groups or roles. If the specified number of votes is not met by a group or a role, or there is a negative vote, there will be no approval and the stage will not advance.

Lifecycle process is applicable to contract artifact, votes can be assigned to contact. Contact voter includes:

- Maintainer(s) of the provider contact voter, this voter is a list of users, roles, groups which have **WRTIE** permission on provider artifact. This list will be automatically filled when requesting approval and list's number required is one.

- **Contact role** contact voter includes Provider of the provider(s), Developer of the provider(s) and this voter is an user who is a contact of provider of consumer artifact.

It is also possible to allow passive approval which is an automatic approval after a specified time period. This may be used if approval is not mandatory but still needs to be visible in the application to show responsibility.

See "How to Define Approvers" in the *Administration Guide* for more information on the procedure for setting of passive approval.

Permissions

Each stage in a lifecycle process has a user(s) assigned different Permissions for that stage allowing them to manipulate the stage based on predefined security settings. These settings are initially empty and are defined by the administrator to include read/write access, ownership, assignment as an approver for advancing the stage and the option to share the artifact globally during a specified stage. These settings are only for the assigned stage and are not carried forward to the next stage.

Note: In only the first stage, domain users are granted read permission by default.

See "How to Define Permissions" in the *Administration Guide* for details about setting stage permissions.

Lifecycle Approval

The following image depicts the lifecycle of each lifecycle stage. When an artifact enters a stage then the stage has the status *In Progress*. After all tasks have been completed , a user requests approval and the stage status is changed from *In Progress* to *Voting*. At the beginning of *Voting*, the system checks whether all required policies are complaint. If not, the status is changed from *Voting* to *Failed*. The requestor is informed about the failed policy validation and he must repair the artifact and then request approval again. Once all required policies are compliant, approvers are notified that their vote is required. When all approvers approve the request, the stage status is changed from *Voting* to *Approved*. If any approver rejects the request, the request is rejected (regardless of the fact that some approvers may approve it) and the stage status is change from *Voting* to *Rejected*. The requestor is notified about the rejection and he must repair the artifact(s) and request approval again.
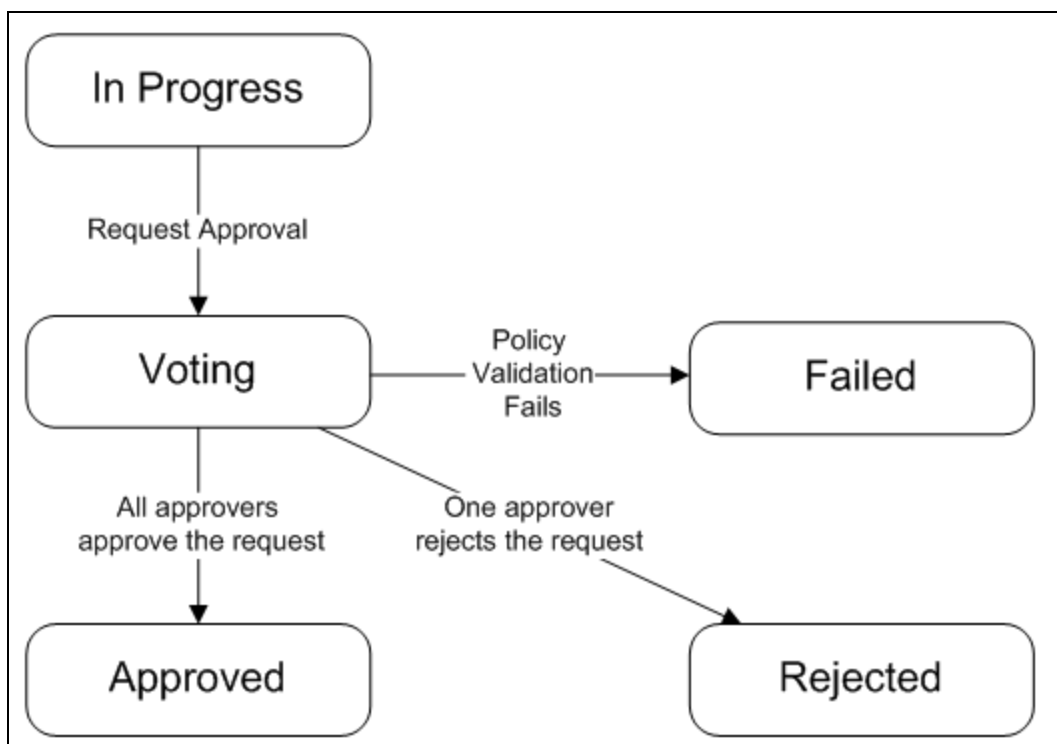
The following list describes the main activities for each stage status.

- **In Progress**

    User primarily wants to have the artifact approved in the current stage and so he wants to request approval. Before he can request approval, all tasks must be completed and required policies should be compliant. If any required policy is non-compliant then the user can ask for approval, but the

system checks the compliance of policies at the beginning of the voting and status is automatically changed to *Failed*.

Tasks can be assigned to specific users (e.g. via roles), but task assignment does not block task completion. This means that even users who do not have the task assigned can complete it. The system tracks who completes the task and when. If a policy is assigned to the task, the task cannot be marked as finished if policy validation fails.

In addition to requesting approval, a user can move artifact to next stage, but only stages connected via the transition *Manual Anytime* type can be used.

- **Voting**

  The artifact is locked for editing during voting.

  Approvers are notified (via mail and events) about a new request waiting for their vote. They can either approve or reject the request. Before an approver casts his vote he can view any changes that happened in the stage; i.e. he can compare the last approved revision with the current one via differences. Approvers approve or reject not only the root artifact but also all sub-artifacts from the lifecycle tree.

  Users and approvers can view the voting status. They can find out which approvers approved the request, read approvers' comments, and when the approval was given. They can also view which approvers did not vote.

  At the beginning of the voting process, policies are checked and if any required policy is non-compliant then approvers are not notified about the request. The request is automatically canceled, its status changed to *Failed* and the requestor is notified via mail and events.

- **Approved**

  In a perfect scenario, all approvers approve the request and so the request is approved. The stage status is changed from *Voting* to *Approved* and the requestor is notified about the approval via mail and events.

  When an artifact(s) is approved and if there is a transition of *Automatically, When Candidate is Approved* type (e.g. in initial lifecycle stages) then the artifact(s) is automatically moved to the next stage and so the *Approved* status of one lifecycle stage is triggering the automatic change to the *In Progress* status of the next lifecycle stage.

If approved artifact from the lifecycle tree is changed then the status is changed from *Approved* to *In Progress*.

- **Rejected**

  When any approver rejects the request, the request is rejected and the stage status is set to *Rejected*. The requestor is notified about the rejection via mail and events, and must repair the artifact and request approval again.

- **Failed**

  As mentioned above, when any required policy is non-compliant, the status is set to *Failed*. The user is notified about this by events and must repair the artifact and ask for approval again.

Automatic Assignment

Automatic Assignment applies Lifecycle processes to ungoverned new or recycled artifacts and functions in the following way:

1. Systinet checks that there are lifecycle processes for the artifact type which are marked as **Automatically Assigned.**

2. Systinet checks that the artifact matches the following conditions:

   - The artifact is a root artifact type for the lifecycle process.

   - **In Categories**: The artifact must contain these categories to be assigned to the lifecycle process.

   - **Not In Categories**: The artifact must not contain these categories to be assigned to the lifecycle process.

   - **In Domains**: The artifact in these domains to be assigned to the lifecycle process.

   - **By Script**: The artifact to be assigned to the lifecycle process by specified script. For more details on scripts for lifecycle process refer to "How to Create Lifecycle Processes" in *Administration Guide*.

3. The artifact is assigned to a lifecycle process as follows:

   - If there is no matching lifecycle process, the artifact is not assigned to any lifecycle process.

   - If there is one matching lifecycle process, the artifact is assigned to it.

   - If there are two or more matching lifecycle processes, the artifact is assigned to the most appropriate one based firstly on the number of matching **In Categories**, and then the number of matching **Not in Categories**.
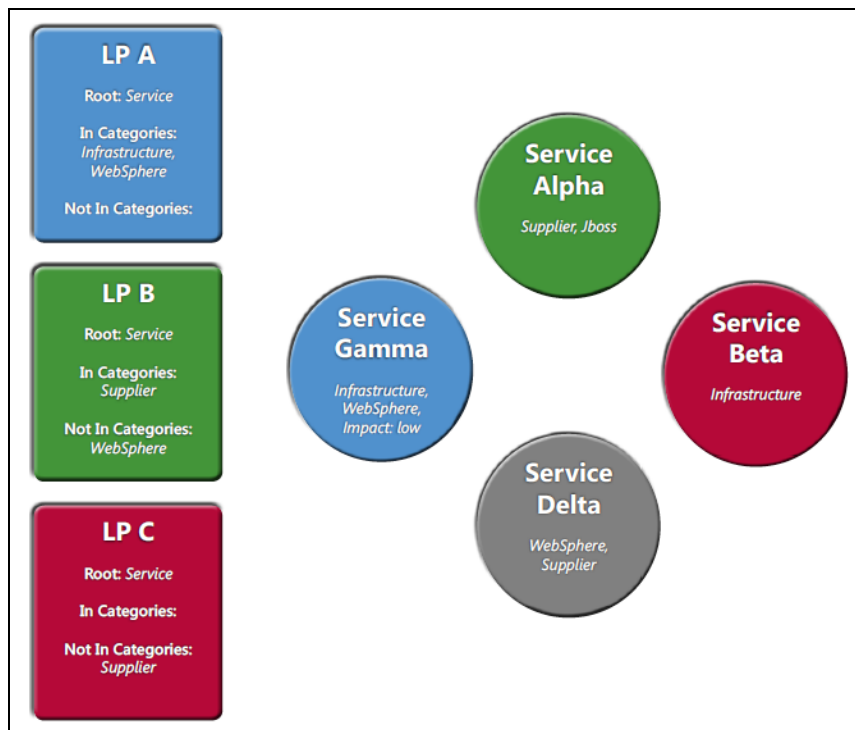
- Service Alpha is assigned to Lifecycle Process B because of the *Supplier* category match.

- Service Beta is assigned to Lifecycle Process C. It cannot be assigned to Lifecycle Process A because it is not categorized as *Websphere*.

- Service Gamma is assigned to Lifecycle Process A because of the *Infrastructure* and *Websphere* category matches.

- Service Delta cannot be assigned to a lifecycle process so it remains unassigned. Lifecycle Process A specifies that an artifact must be **In Category** *Infrastructure* and Lifecycle Processes B and C both specify **Not In Categories** used by the service.

> **Tip:** Ideally, all artifacts are automatically assigned. See "Lifecycle Best Practice" in the *Administration Guide* for more information.

How to Start Governance

This function enables the governance of an artifact that is not currently governed, and where the artifact's type matches the root artifact type in an existing lifecycle process.

> **Tip:** Systinet automatically assigns new artifacts to lifecycle processes according to association rules set when creating a lifecycle process. This also applies when an existing artifact is modified,

is not yet governed and an automatic process exists. Assigned Automatically must be selected for this process to come into effect.

To start governing an artifact:

1. From the Lifecycle tab of any ungoverned artifact, select **Start Governance**.

   Systinet does one of the following:

   - If appropriate lifecycle processes exist that are marked for automatic governance, Systinet automatically assigns the artifact to the most appropriate process.

   - If there are no appropriate lifecycle processes, Systinet provides a warning message.

   - If there are appropriate processes that are not marked for automatic governance, the Start Governance popup opens enabling you to manually select from the appropriate lifecycle processes for the artifact.

     Note: Starting governance is also available from a table of artifacts and can be used to start governance on multiple artifacts at once.

2. If Systinet offers manual lifecycle process selection, in the Process field, select the process you want to use and click **OK**.

How to Review/Close Lifecycle Tasks

The full list will be presented in the Tasks page along with tabs separating tasks into **Lifecycle Tasks**, **Lifecycle Requests** and **Contract Requests**.

Clicking on a lifecycle task will present the task in the Lifecycle tab and display what is required of the user and clicking the check box will indicate that the task has been completed. Tasks will also appear on the right-hand menu of the artifacts' Overview tab under the heading **My Tasks** and can also be marked as completed from this location.

Administrator can remove the policies and complete the task by **Remove Policy** button next to each policies.

Assignee can only complete/uncomplete the assigned tasks.

Screenshot: Lifecycle Task Request
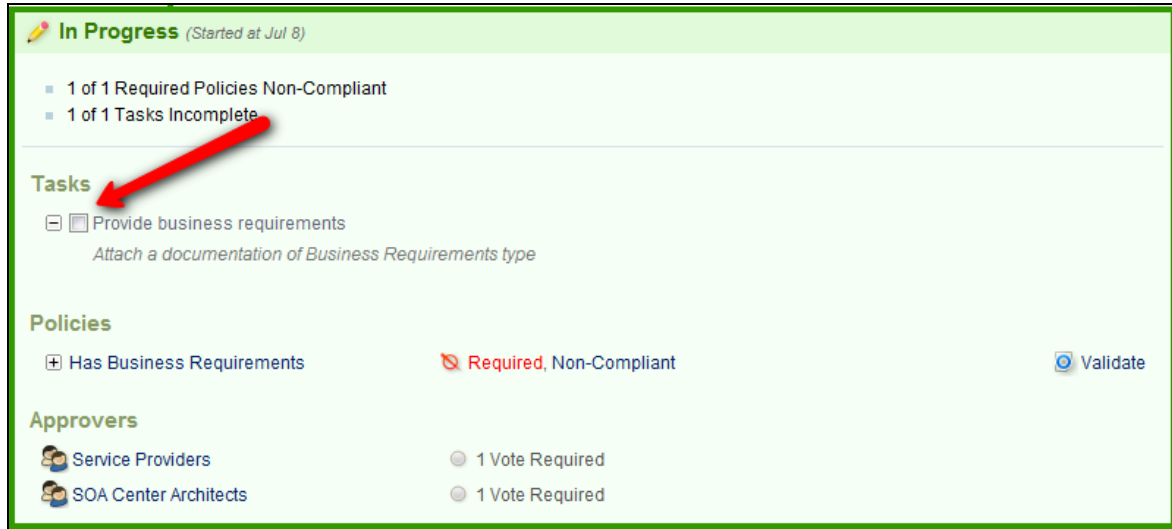
How to Review/Validate Policies

Policies can be reviewed and validated in the Lifecycle tab of any governed artifact detail page. Clicking on the + icon associated with the policy will give more information about the policy. The UI shows whether a policy is required or optional. A User can run validation of a policy manually and the policy validation result is presented after it is complete. The artifact's compliance status based on validation of policies is presented on page right-hand side.

See "How to Review Compliance Status" below for more information about compliance.

See "Lifecycle Policies" in the *Administration Guide* for more information about policies.

How to Review Compliance Status

As part of a lifecycle process, there may be technical policies associated with a lifecycle process lifecycle stage. When the artifact is at a particular lifecycle stage, these technical policies apply to it.

There are two policy types:

- A policy associated with a task: there can be only one policy per task and such policies are required for compliance to be met.

- Policies associated with stage: such policies may be required or optional and can be applied on selected types of artifacts.

Note: Only the policies associated with tasks and the required policies are taken into account in compliance status. When compliance is recalculated by the user, validation of all these policies is run.

Systinet executes a Policy Compliance Task which performs a validation of all artifacts with applicable policies. The summation of these policy validations gives a compliance status for an artifact, expressed in terms of a percentage of policies that the artifact complies with.

This task sets a new compliance status based on the results with an accompanying compliance status report. The compliance status is displayed in the artifact view sidebar and in all list views or tables of artifacts and can also be viewed in the Compliance tab.

When an artifact changes, the compliance status becomes obsolete, indicated by a dashed line. You can manually execute a compliance status validation.

To execute a manual compliance status validation:

Do one of the following:

- In the Overview or Lifecycle tab of an artifact Compliance is shown in a widget on the right-hand side.

- In the Compliance Status tab, click **Recalculate**.

The artifact is revalidated and a new compliance status calculated. The percentage of compliance is displayed in a bar graph and if further details are required, clicking **Details** opens the compliance tab for artifact. An artifact must be governed in order for a compliance status to exist.

See "Artifact Details Page Compliance Tab" on page 42 for a description of the Compliance tab.

How to Request Approval

To submit an approval request:

1. In the Lifecycle tab of a governed unapproved artifact that has all required tasks completed, click **Request Approval**

2. Add an optional comment, and then click **Submit Request** to create the approval request for the users, roles and groups listed in the Approvers section for the current lifecycle stage.

As part of the approval process, Systinet automatically validates any policies associated with the current working stage. A progress bar shows the validation progress in the message bar. If the policy validation fails, Systinet cancels the approval request. The report can be accessed from the artifact view page. If policy validation succeeds, the artifact owner, stakeholders, contacts, and approvers are notified of the request by e-mail.

Approvers can be sent a notification with an optional text by clicking **Notify Approvers.**

Requests can be canceled by clicking **Cancel Approval**.

> **Note:** While waiting for approval, the artifacts in the lifecycle tree is locked and cannot be edited or deleted. Artifacts in the lifecycle tree are unlocked when approval is granted or denied.

To check the status of an approval request, look at the voting details in the artifact in the Lifecycle tab.

HP Software also provides a set of IDE plugins with integrated request approval functionality. For details, see *Plugin for Eclipse* and *Plugin for Visual Studio*.

How to Approve/Reject Approval Requests

A request requiring approval is sent as an automated notification mail, and an event in the UI and appears in the pending tasks list. The request can be approved by clicking on **Approve** or the request can be accessed by clicking on the task or from the **My Tasks** list on the right-hand side of an artifact Overview tab and is presented in the Lifecycle tab.

Voting buttons are presented as well as other votes required for approval.

Clicking on **Approve** adds the vote to the list and provides an optional text box that can be used to give a description of the change.

Clicking on **Reject** stops the advancement of the process and opens a text box requiring input as the reason for the rejection.

A notification e-mail is sent to the artifact owner, contacts, approvers, and stakeholders with the result of the approval request (approved or rejected). If Automatic transition is set to next stage, the artifact is moved to the next stage when the current stage is approved. Otherwise user moves the artifact to the next stage manually from the right context menu in Lifecycle tab.

How to End Governance

This function enables you to end the governance of an artifact.

To end artifact governance:

1. From the Lifecycle tab of any governed artifact, click **End Governance**

2. A confirmation dialog will open and clicking **End Governance**.will end governance. Click **Cancel** to maintain governance.

   The artifact is no longer governed.

   > **Note:** Ending governance is also available from a table of artifacts and can be used to end governance on multiple artifacts at once.

   > **Note:** Artifact governance can be ended at any time including when it is locked for changes.

How to View Lifecycle History

The history of approvals for an artifact is available from the Lifecycle tab of any governed artifact.

To open the governance history of an artifact:

- Click on the **History** button to open the **Lifecycle History** popup window.

  The window presents Lifecycle changes to the artifact and if the **Approved** , **Cancelled**, **Rejected** or **Approval Failed** links are clicked the user is taken to the stage description of the event.

Click **Close** to close the **Lifecycle History** window.

Chapter 10: Runtime Gateway Interoperability Framework (RGIF)

This part contains the following topics:

Concepts

- "RGIF Overview"

-

- "Manual Proxy Creation and Contract Deployment"

-

-

-

Runtime Gateway Interoperability Framework:

- Remote implementation (adapter)

- RGIF discovery

- REST service deployment

Remote implementation (adapter)

RGIF implementation is decoupled from Systinet 10 codes. Integration between Systinet and PEP (Policy Enforcement Point) is now communicated through a RGIF interface.

Systinet 10 contains RGIF implementations for Layer7 (7.1, 8.1) and Datapower (v6, v7) OOTB.
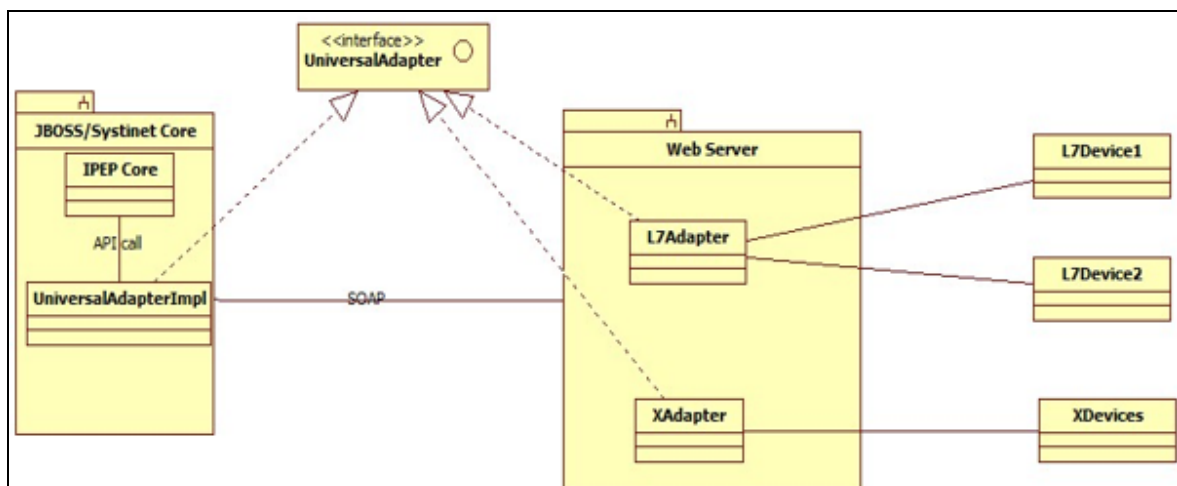
RGIF discovery

Systinet 10 RGIF provides ability to discover REST and Web services from PEPs.

From runtime devices, Web services are discovered and synchronized to Systinet as SOA artifacts, while REST services are synchronized as API artifacts. Discovered artifacts could then be governed within Systinet or deployed to different runtime devices or environments.

Policies from runtime devices are also discovered, decomposed and synchronized into Systinet as RGIF artifacts (proxy, universal policy and policy parameter artifacts).

Discovery UI

Discovery UI is introduced for manual discovery by administrator (or domain administrator).

Discovery Task

Discovery task is introduced for automatic discovery.

Administrator first specifies whether to decompose WSDL of runtime services and synchronization mode by using **platform.rgifdiscovery.wsdls.decomposition** and **platform.rgifdiscovery.synchronization.mode** system properties or by the configuration.

Then set schedule for RGIF Discovery Task to run at a regular interval.

This Week

Report Cleanup Task
Next Run: Jul 4, 2014 7:00:00 AM
Recurrence: Daily
▶ Run | 🖊 Edit Schedule

Lifecycle Validation Task
Next Run: Jul 4, 2014 7:30:00 AM
Recurrence: Daily
▶ Run | 🖊 Edit Schedule

Policy Report Validation Task
Next Run: Jul 4, 2014 8:30:00 AM
Recurrence: Daily
▶ Run | 🖊 Edit Schedule

More...

No Schedule

RGIF Discovery Task
Next Run: None
▶ Run | 🖊 Schedule

Repository Data Diagnostic Task
Next Run: None
▶ Run | 🖊 Schedule

RGIF Devices Validation Task - Layer7 v8
Next Run: None
▶ Run | 🖊 Schedule

RGIF deployment

REST service deployment - Repository users could deploy REST services from Systinet 10 to PEP as they used to doing with Web services in Systinet 4.10.

RGIF Overview

Every corporation has information about IT assets dispersed in two different worlds:

1. Operations running the applications in run-time.

2. Applications in design-time governance.

RGIF provides enforcement of both run-time and design-time policies and is based on Systinet contract management with contracts being identified during run-time as well as detecting rogue services. Service proxies are auto-deployed to run-time integration as managed endpoints and service capacities are monitored on a per contract basis.

RGIF features multi-device support (such as L7 and IBM Datapower).

The following image shows how a typical communication scenario appears with runtime policy enforcement and without the HP Systinet Repository.
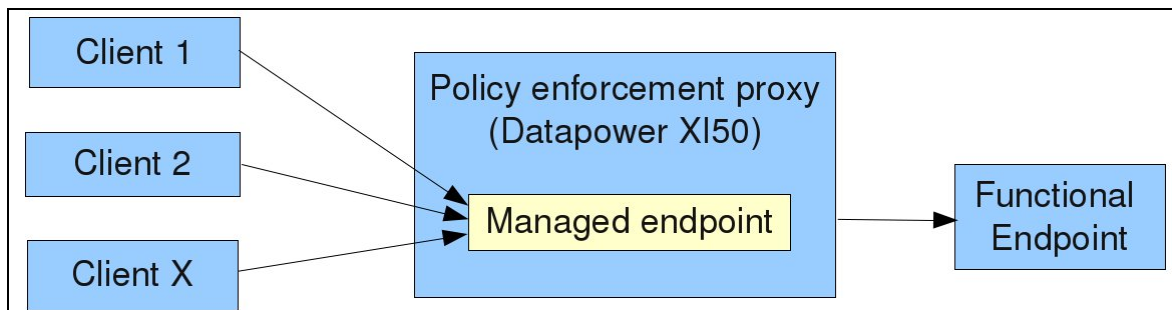
Clients send messages to the policy enforcement RGIF device (also known as policy enforcement point, PEP) and onto a managed endpoint that has been created by the PEP administrator in order for the forward messages to be sent to the functional (application) endpoint. During the forwarding process runtime policies are applied on the messages being forwarded. These are configured by the PEP administrator as well.

The image below represents the same process with the Systinet repository.
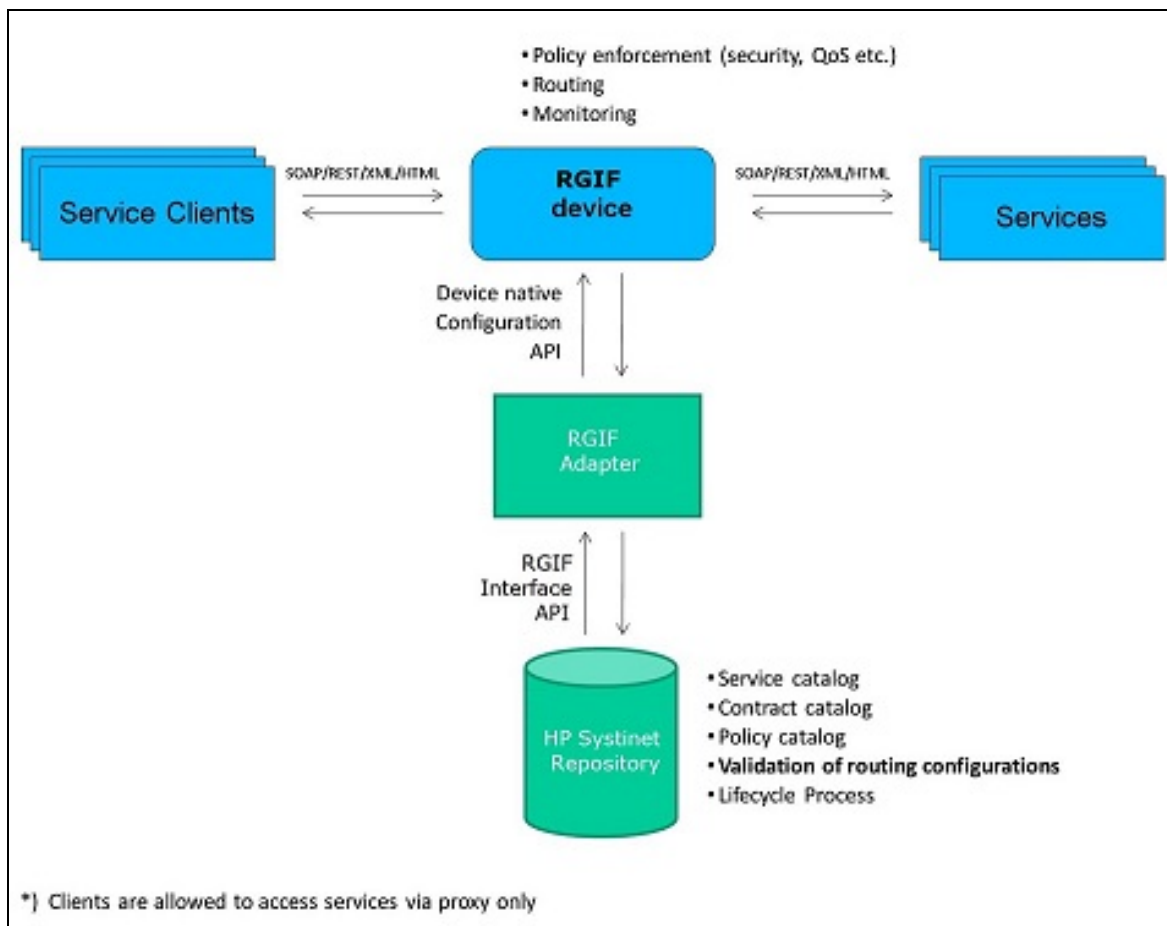
When the Systinet repository comes into play the messages between the service client the service flow in the same way as did in the scenario without Systinet repository, however, the configuration of the PEP device is now handled by Systinet Repository based on the service / policy metadata configured within. Note that Systinet is able to configure multiple devices from different vendors at the same time. When a service consumer / client is using a service, it is captured within the Systinet repository as a contract. Support for different PEP vendors is handled by independent PEP devices and these can be created by implementing an API.

Each device configuration is partitioned into proxy objects with each representing a single managed endpoint and is intended to be created for each functional service endpoint. Each proxy object has a persistent counterpart in Systinet repository represented by a single proxy artifact.

Proxy objects configured on PEPs that do not have the proxy artifact counterpart are referred to as rogue proxies/services.

Proxy objects are created either manually using a dedicated wizard or by an automatic action of a lifecycle process as a side effect of contract deployment. In other words, contract deployment is an action when a proxy object is created or updated to allow message exchange between a service consumer and a service provider.

Feature summary

Contract development process

- User defined workflow with approval process

- Custom policies and control over manual tasks

- Per contract service monitoring and QoS enforcement

- Tracking of the full history from design time to runtime (included)

- Discovering service utilization trends

- Identification of capacity issues

- Capacity planning

Environment management

- Separate environments for e.g. testing/staging/production

- User/role based access control to service information (endpoints, WSDLs etc.)

- Policy management

- Assured information reliability and consistency using policies and lifecycle identification of rogue contracts and service endpoints

- Automatic distribution of policies to integration devices

- Policy versioning

Devices for automatic configuration of runtime governance integration devices

- Vendor neutral (Layer 7, IBM Datapower)

- Open architecture

Runtime Contracts

Because proxy objects are created on behalf of a contract, you can apply contract enforcement. This means that only messages sent by negotiated service consumers that are configured in the repository are allowed to send messages to the application endpoints. This is the default behavior of the proxy objects created by Systinet.

RGIF devices need to perform contract identification to ensure contract enforcement. In proxy objects, it is encoded which contracts are allowed to send messages to the proxy object. The identification itself is performed based on message properties such as an HTTP header holding the contract ID sent within each message. There are some other means to perform contract identification; HTTP basic credentials, XPATH expression evaluated over the message and others.

Proxy objects created by Systinet also allow per contract service monitoring, which means that statistics such as the number of messages sent per day by an individual contract may be monitored.

See "Contracts Overview" on page 223 for more information about contracts.

Manual Proxy Creation and Contract Deployment

Proxy objects can be created on demand directly from the repository user interface using a dedicated UI wizard. This means of proxy creation/contract deployment is meant to be used primarily by administrators to fix exceptional states when an automated proxy creation within a lifecycle fails. In some situations where full lifecycle process is not required, it might be used by regular users to create proxies. This feature must be explicitly enabled in the general contract deployment settings.

Lifecycle Based Contract Development

Proxy objects can be also created during an contract artifact lifecycle by assigning Create/Remove PEP Proxy automatic action assigned to a lifecycle stage. The most simple lifecycle process looks like the following:
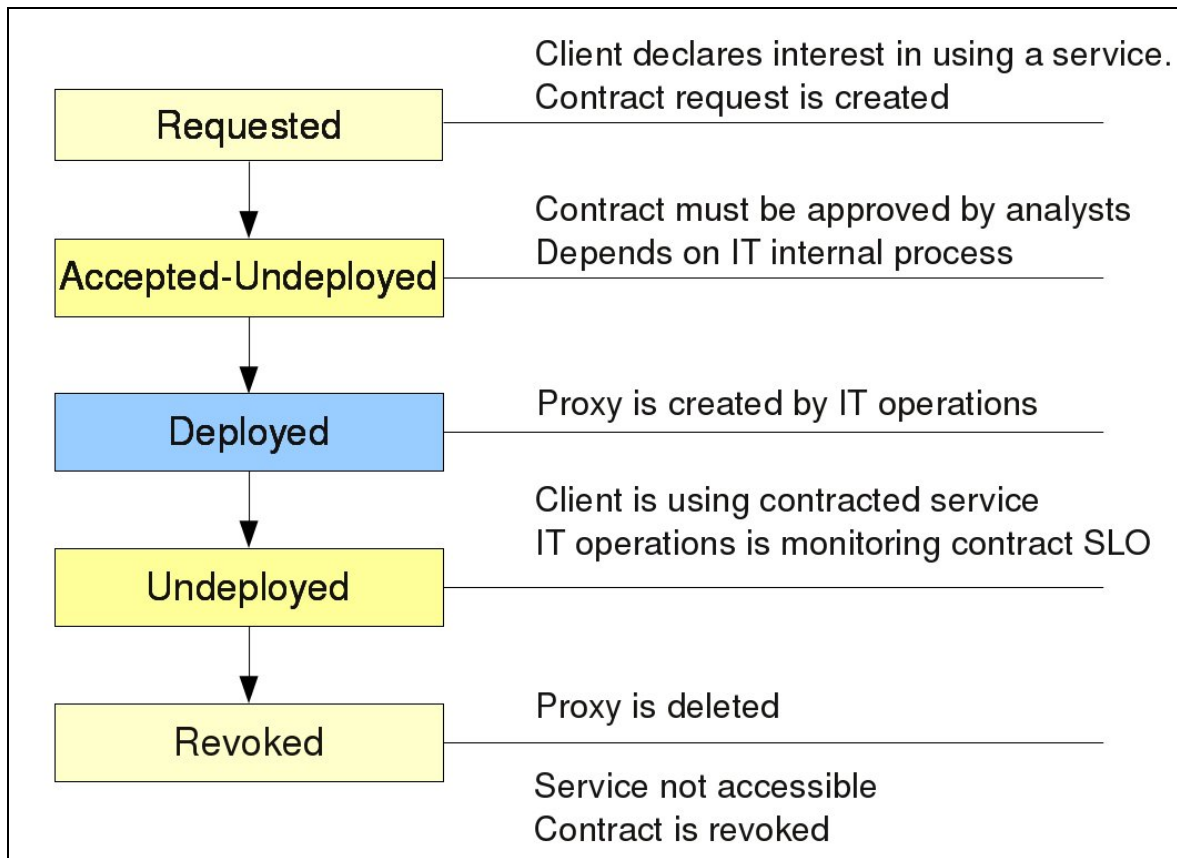
Note that the "Requested" state is represented by a contract request artifact (see "Contracts Overview" on page 223 for more information) and the contract lifecycle starts in the "Accepted-Undeployed" state. It is useful to configure the lifecycle process to be automatically assigned to contracts.

The above lifecycle might be too simple to be used in a typical IT environment where there are different (separated) environments/landscapes for service development/testing/production. Systinet supports this scenario as well and all that is required is to associate RGIF devices with environments and define the contract lifecycle in a similar fashion as likely done for the service lifecycle process – there will be stages matching the environments defined in the lifecycle.
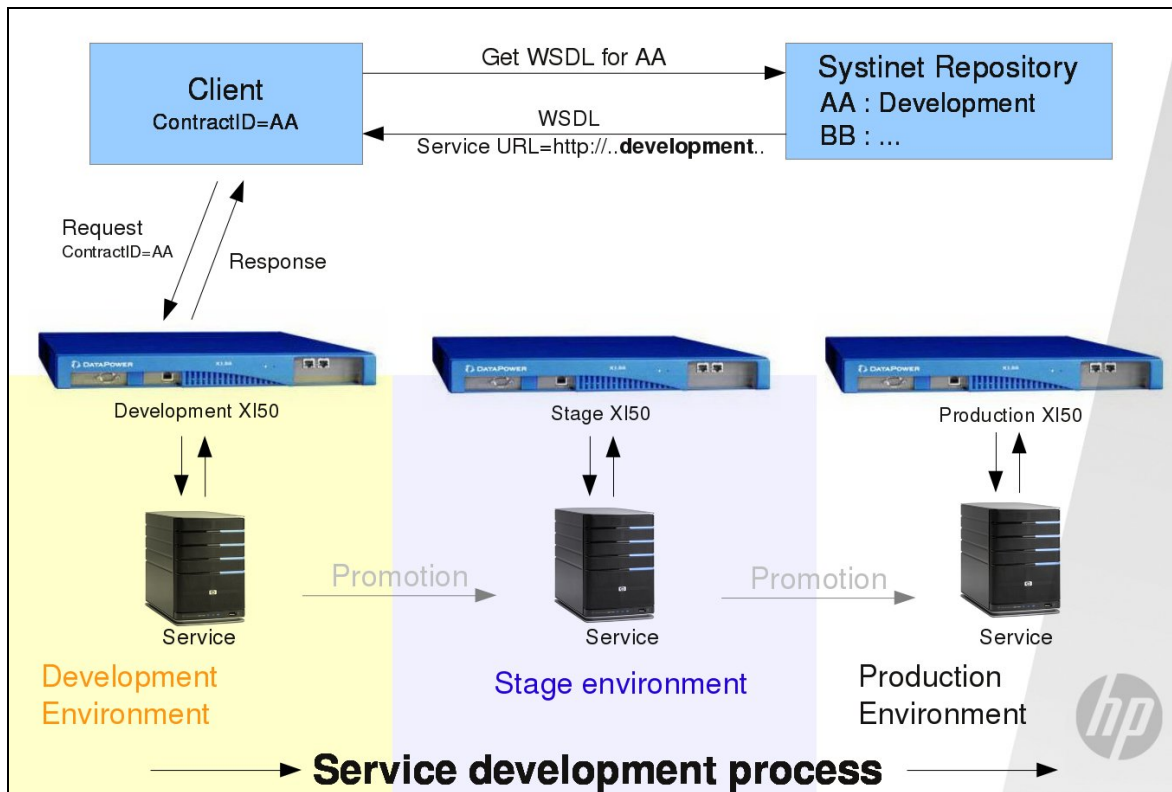
The contract lifecycle has stages matching the service lifecycle and it means the following – when a contract is in the development environment lifecycle stage it is accessing managed endpoints created for the service hosted in the development environment. In the image above the contract AA is in the Development environment lifecycle stage .

The process is as follows:

1. The client creates a WSDL request on the Systinet Repository passing his contract ID.

2. Repository returns WSDL with the service url pointing to the RGIF device assigned to the Development environment

3. Based on the information in the WSDL the client sends a message to the proper proxy object hosted on the RGIF device and the response is returned

When the contract lifecycle is promoted to the next lifecycle state the client acts in the very same way but will receive a different URL within the WSDL.
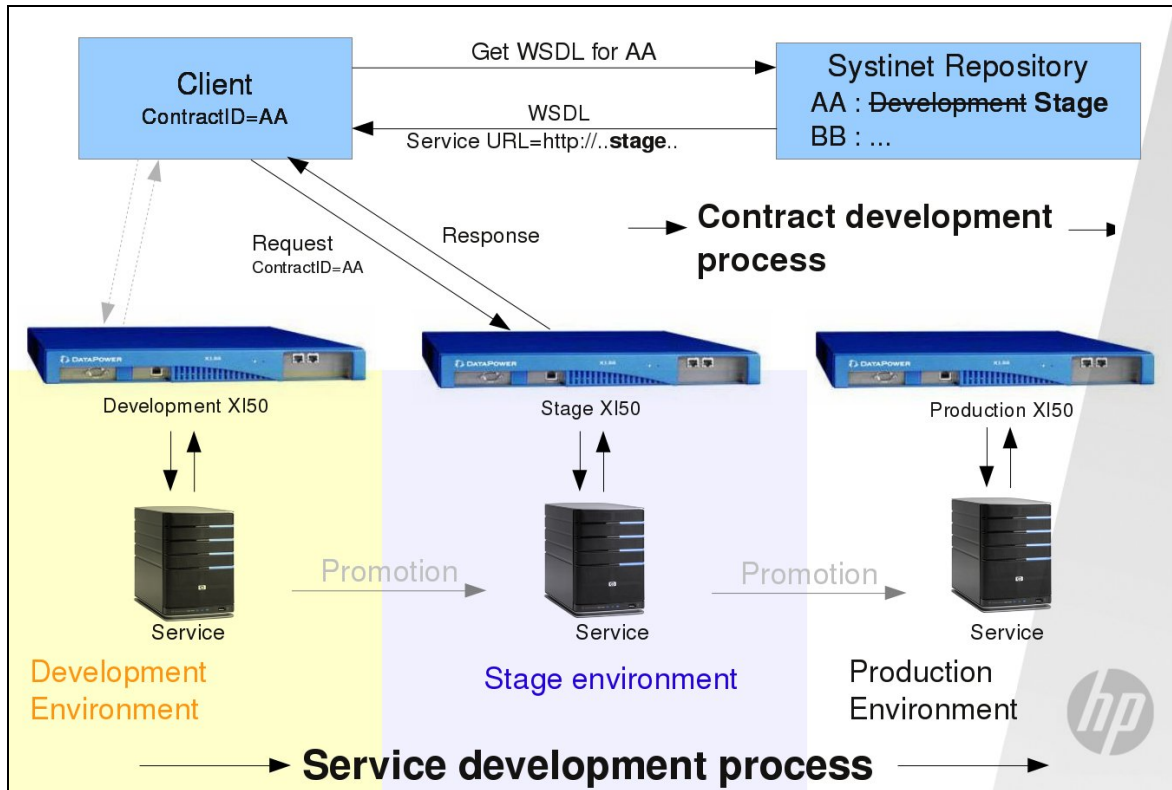
The important and valuable aspect about this is that contracts are moved between environments just by changing the information in the repository; client reconfiguration is not required. This becomes useful when the client is deployed multiple times on different computers as in the example of a desktop application being used by multiple users.

Also note that you can have a single contract deployed across multiple environments; in this case the client must pass except the contract ID also the environment which is intended to be accessed.

See "Lifecycle Overview" on page 238 and "Automatic Actions" on page 242 for more information about lifecycle processes.

Composite Application Development Lifecycle

Composite applications are those that are formed from multiple services, are captured within the Systinet repository and have contracts created between them. Composite applications may have their own dedicated lifecycle process and if this is correctly defined it can be used to move composite application contracts and proxies created in these lifecycle processes from one environment to another – Systinet automatically creates these proxies in the new environment.

See "Lifecycle Overview" on page 238 for more information about lifecycle processes.

Propagating Service Changes to Proxies

When a service is changed (such as the change of a functional endpoint), propagating these changes into the RGIF devices is required. This is not done automatically, and you will need to trigger this operation from a lifecycle associated with the modified service (this can be done for implementations and endpoints as well).

> **Note:** This procedure only updates proxies that were previously created for contracts where the service is the under the role of provider. If you are creating proxies for artifacts other than contracts you need to use the **Create/Remove proxy** procedure and check both the deploy and undeploy check boxes for the affected environment(s) – when executed the proxy will be recreated.

When the procedure **Update proxies** is executed, it drops all the related proxy objects and recreates them based on the latest information in the SLO and the related endpoint/wsdl information.

In the following image you can see the dialog for the configuration of the **Update proxies** procedure.
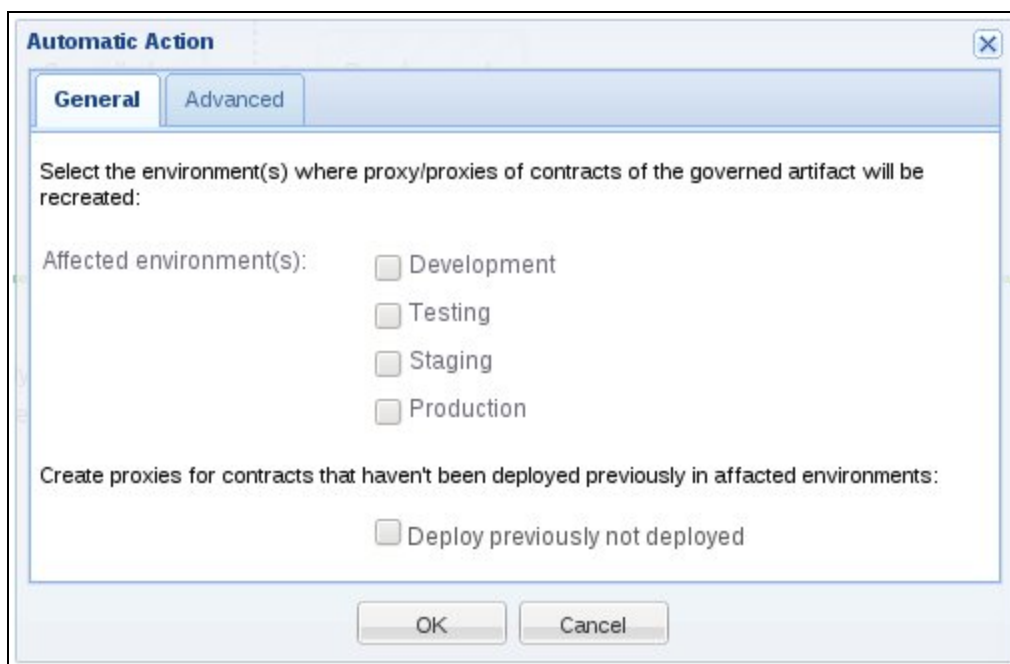
The update proxies lifecycle procedure is able to create proxies if they didn't previously exist. However, you need to enable this feature by checking the **Deploy previously not deployed** check box.

Quick Start

Follow this section to quickly set up Systinet and RGIF devices for use with policy enforcement automation.

- "Adding RGIF servers"

- "Creating Contract Management Artifacts"

- "Defining Default RGIF System Settings"

- "Setting Contract Development Lifecycle Process"

- "Publishing a Service"

- "Defining Service SLO"

- "Creating Contracts"

- "Accessing Service Through Proxy"

- "How to edit the adapter configuration"

Deploy a Remote Layer 7 Adapter

Systinet 10 introduces the concept of RGIF adapters. Systinet RGIF Adapter is designed to place between Systinet and the proxy management devices. For Systinet server, the RGIF adapter provides management of runtime proxies on the device.

RGIF adapter is a standalone software component deployable to the application server. The adapter exposes WSDL for Systinet. Systinet connects to the adapter as a web service client.

1. Layer 7 adapter package information is provided in: **%SYSTINET_HOME%/deploy/rgif-adapter-layer7.war**

2. Deploy Layer 7 adapter to `jboss` by copying **rgif-adapter-layer7.war** to **%JBOSS_HOME%/[node_name]/deployments**

3. Add Layer 7 certificate to **%SYSTINET_HOME%/conf/client.truststore**

4. Add Layer 7 certificate to **%SYSTINET_HOME%/bin/serverstart.bat(.sh)** using:

   ```
   + SET JAVA_OPTS=%JAVA_OPTS% -Djavax.net.ssl.trustStore=%SYSTINET_
   HOME%\conf\client.truststore
   ```

5. Restart **Systinet**.

6. Start the Layer 7 adapter wsdl: **http://[host]:[port]/rgif-adapter-layer7/layer7adapter?wsdl**

Deploy a Remote Datapower Adapter

Administrators can configure a remote datapower adapter that allows creation of IBM DataPower XI52.6.0.1.0 devices in Systinet.

1. Datapower adapter package information is provided in: **%SYSTINET_HOME%/deploy/rgif-adapter-dp.war**

2. Deploy datapower adapter to `jboss` by copying **rgif-adapter-dp.war** to **%JBOSS_HOME%/[node_name]/deployments**

3. Add datapower certificate to **%SYSTINET_HOME%/conf/client.truststore**

4. Add datapower certificate to **%SYSTINET_HOME%/bin/serverstart.bat(.sh)** using:

   ```
   + SET JAVA_OPTS=%JAVA_OPTS% -Djavax.net.ssl.trustStore=%SYSTINET_
   HOME%\conf\client.truststore
   ```

5. Restart **Systinet**.

6. Start the datapower adapter wsdl: **http://[host]:[port]/rgif-adapter-dp/dpadapter?wsdl**

Adding RGIF servers

1. Sign on to **Systinet GUI** as `Administrator`.

2. Click on **Integrations** and the **Default Domain** interface opens.
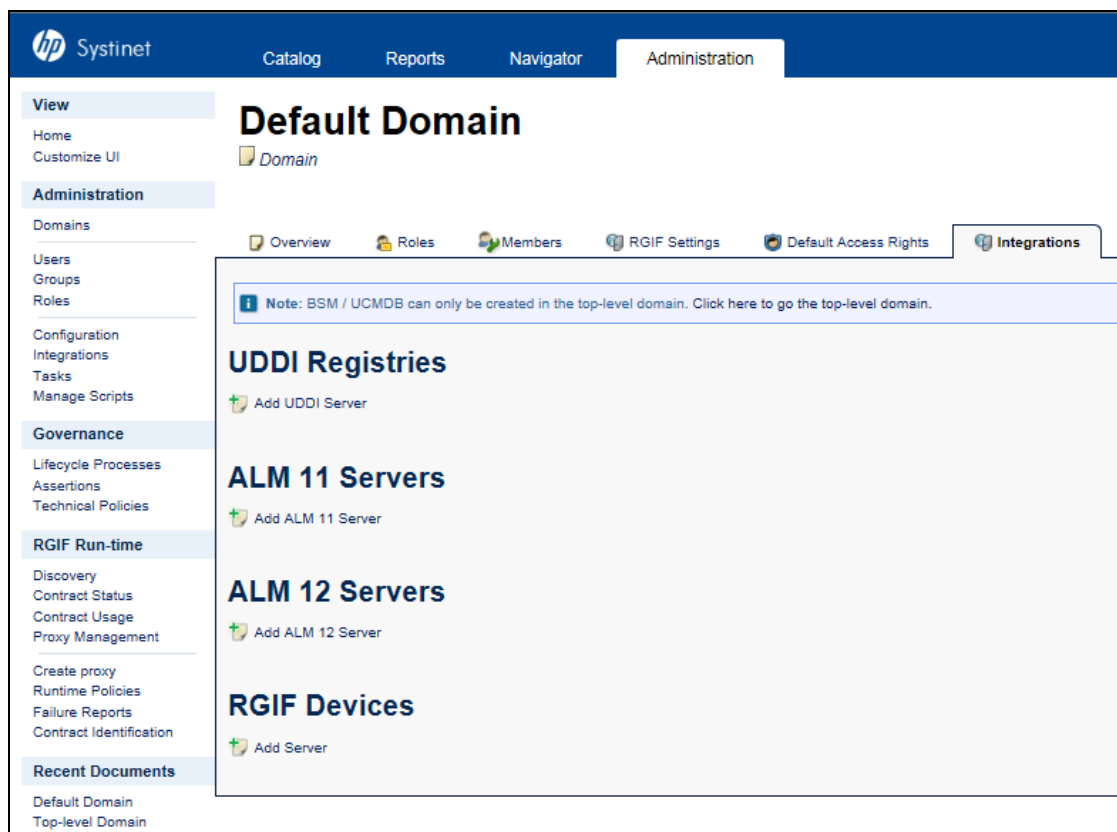
3. Add the adapter URL in the **Adapter URL** field.

4. Fill in other information and click **Test Connection**.

There are two links to review/change the remote adapter configuration:

Edit Configuration

a. Click **Edit Configuration** link, the Edit Configuration dialog box is displayed.

b. Modify the adapter configuration content, for more information see "How to edit the adapter configuration" section.

c. Click **Validate** to validate the configuration.

d. Once the content is valid, the **Save** button will be enabled. Click **Save** to save the configuration.

e. If the validation is invalid, a error message will be displayed to show the reason of invalid validating. Both **Validate** and **Save** buttons will be disabled till the configuration is fixed.

Reset Configuration

- Click on **Reset Configuration** link, a warning prompt will appear: `'Do you really want to reset the adapter configuration to its initial status? This will erase current settings without any possibility to restore them'`. Click OK if you want to reset the configuration.

5. Click on **Save** when the Test Connection is successful.
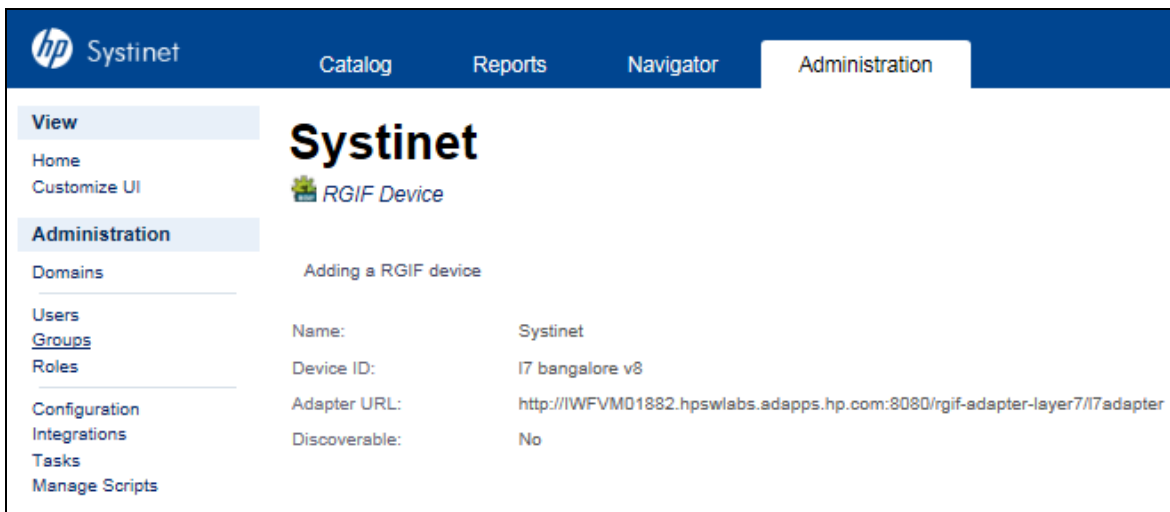
How to provide additional artifact properties to Layer7 server

In order to provide additional artifact properties, following are the steps:

1. Modify the script artifact **iPepAPI.js**
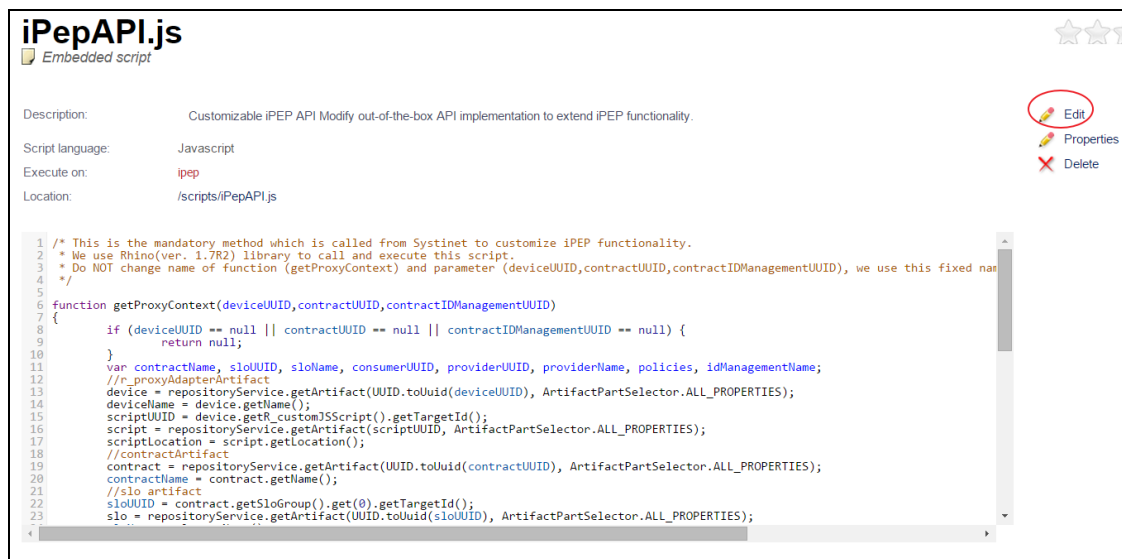
The script artifact needs to be modified in order to add additional artifact properties into result variable.

- Use the line of code below to get information of an artifact `var art = repositoryService.getArtifact(UUID.toUuid(uuid), ArtifactPartSelector.ALL_ PROPERTIES)`

- Artifact property is defined by:

  `art.getProperty(property Name).toString()` to get string property

- Additional artifact properties into result variable is defined by:

For example, property A1 of consumer and new artifact is added:

```
var A1= art.getProperty(A1).toString();

var newUuid=…..

var newArtifactName=…

var result = {device:{uuid:deviceUUID,name:deviceName},

contract:{uuid:contractUUID,name:contractName},

slo:{uuid:sloUUID,name:sloName},

consumer:{uuid:consumerUUID,name:consumerName, propertyA1:A1},

provider:{uuid:providerUUID,name:providerName},

identity:{uuid:contractIDManagementUUID,name:idManagementName},
```

```
script:{uuid:scriptUUID,name:scriptLocation},

newArtifact {uuid:newUuid,name:newArtifactName }

};

result['policies']=policies;
```

- To add the new properties of runtime policy artifact, the following line of code needs to be modified:

```
temp += '{"uuid":"' + policy.get_uuid() + '","name":"' + policy.getName() +
'"}';
```

2. Modify the template file Layer 7 template file

- Extract the OOTB default template file from `ipep-adapter-layer7.war\WEB-INF\classes\com\hp\systinet\proxy\mgmt\impl\l7\policy\templates\default.ft`

- Add your custom template file in Jboss7 AS in standalone mode.

- Open **standalone.bat** then add following file:

```
SET JAVA_OPTS=%JAVA_OPTS% -Drgif.adapter.l7.policy.template.file=custom-
config-path/custom-template.ftl
```

- In **custom-template.ftl**, use the syntax of FreeMarker to access to java objects.

> **Note:** Systinet is using FreeMarker version 2.3.14.

Examples

${context['consumer']['name']}	to get the artifact name of consumer artifact
${context['contract']['name']}	to get the contract name
${context['policies'][0].name}	to get the name of the first policy which is added into SLO artifact of the contract

> **Note:** Policies are stored in an array.

How to edit the adapter configuration

1. In the Administration tab, click on **New RGIF Device**.

2. Add the adapter URL in the Add RGIF Device homepage.

3. In **Add RGIF Device** page, click "**Edit Configuration**" link.

4. Modify the content of adapter configuration based on below schema:

   a. **Layer7 adapter**

```
<xs:schema xmlns:tns="http://adapter.rgif.hp.com/"

elementFormDefault="qualified" targetNamespace="http://adapter.rgif.hp.com/"

xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:simpleType name="NonEmptyString">

<xs:restriction base="xs:string">

<xs:minLength value="3"></xs:minLength>

</xs:restriction>

</xs:simpleType>

<xs:element name="adapter">

<xs:complexType>

<xs:sequence>

<xs:element type="tns:NonEmptyString" name="name"

maxOccurs="1" minOccurs="1" nillable="false" />

<xs:element name="device" maxOccurs="unbounded"

minOccurs="1">

<xs:annotation>

<xs:documentation>Device adapter name is filled here
```

```
</xs:documentation>

</xs:annotation>

<xs:complexType>

<xs:sequence>

<xs:element type="tns:NonEmptyString" name="deviceId"

maxOccurs="1" minOccurs="1">

</xs:element>

<xs:element type="xs:string" name="location"

maxOccurs="1" minOccurs="1">

<xs:annotation>

<xs:documentation>Device adapter ID is filled here. ID must be

unique....

</xs:documentation>

</xs:annotation>

</xs:element>

<xs:element type="xs:string" name="externalUrl"

maxOccurs="1" minOccurs="0">

<xs:annotation>

<xs:documentation>Mangement URL for device</xs:documentation>

</xs:annotation>

</xs:element>

<xs:element type="xs:string" name="name" maxOccurs="1"

minOccurs="1">

<xs:annotation>

<xs:documentation>Mangement Console URL for device

</xs:documentation>

</xs:annotation>

</xs:element>

<xs:element name="snmp" maxOccurs="1" minOccurs="1">

<xs:annotation>

<xs:documentation>Device name is filled here</xs:documentation>

</xs:annotation>
```

```
<xs:complexType>

<xs:sequence>

<xs:element type="xs:string" name="address" />

<xs:element type="xs:anyURI" name="aggregator" />

<xs:element type="xs:short" name="port" />

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="interface" maxOccurs="unbounded"

minOccurs="1">

<xs:complexType>

<xs:sequence>

<xs:element type="xs:string" name="name" />

<xs:element type="xs:string" name="url" />

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

<xs:key name="deviceId">

<xs:selector xpath="tns:device/tns:deviceId" />

<xs:field xpath="." />

</xs:key>

</xs:element>

</xs:schema>
```

b. **IBM Datapower adapter**

```
<xs:schema xmlns:tns="http://adapter.rgif.hp.com/"

elementFormDefault="qualified" targetNamespace="http://adapter.rgif.hp.com/"
```

```
xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:simpleType name="NonEmptyString">

<xs:restriction base="xs:string">

<xs:minLength value="3"></xs:minLength>

</xs:restriction>

</xs:simpleType>

<xs:element name="adapter">

<xs:complexType>

<xs:sequence>

<xs:element type="tns:NonEmptyString" name="name"

maxOccurs="1" minOccurs="1" nillable="false" />

<xs:element name="device" maxOccurs="unbounded"

minOccurs="1">

<xs:annotation>

<xs:documentation>Device adapter name is filled here

</xs:documentation>

</xs:annotation>

<xs:complexType>

<xs:sequence>

<xs:element type="tns:NonEmptyString" name="deviceId"

maxOccurs="1" minOccurs="1">

</xs:element>

<xs:element type="xs:string" name="location"

maxOccurs="1" minOccurs="1">

<xs:annotation>

<xs:documentation>Device adapter ID is filled here. ID must be

unique....

</xs:documentation>

</xs:annotation>

</xs:element>

<xs:element type="xs:string" name="externalUrl"

maxOccurs="1" minOccurs="0">
```

```
<xs:annotation>

<xs:documentation>Mangement URL for device</xs:documentation>

</xs:annotation>

</xs:element>

<xs:element type="xs:string" name="name" maxOccurs="1"

minOccurs="1">

<xs:annotation>

<xs:documentation>Mangement Console URL for device

</xs:documentation>

</xs:annotation>

</xs:element>

<xs:element type="xs:string" name="domain" maxOccurs="1"

minOccurs="1">

<xs:annotation>

<xs:documentation>Device name is filled here</xs:documentation>

</xs:annotation>

</xs:element>

<xs:element name="interface" maxOccurs="unbounded"

minOccurs="1">

<xs:complexType>

<xs:sequence>

<xs:element type="xs:string" name="name" />

<xs:element type="xs:string" name="url" />

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element type="xs:string" name="restEnforcement" maxOccurs="1"

minOccurs="1"/>

</xs:sequence>
```

```
</xs:complexType>

<xs:key name="deviceId">

<xs:selector xpath="tns:device/tns:deviceId" />

<xs:field xpath="." />

</xs:key>

</xs:element>

</xs:schema>
```

Creating Contract Management Artifacts

From the Catalog tab, expand the Create menu from list of artifact types and select **More**.

In the Runtime Policy Enforcement list, select **Contract ID Management**.

Fill in the form exactly as in the following image:

Note: Not all devices support all means of identity retrieval methods, but both L7 and XI50 handle HTTP headers.

Defining Default RGIF System Settings

1. Navigate to **Administration** ->**Domains**. Select your working domain and click on
   RGIF Settings tab. Here you can define default setting for newly created proxies.Note that these

can be overridden within a service SLO.

2. Enter the **Identity Retrieval** field and click **Save**. The other fields need not be updated.

SLM Retrieval Configuration

These settings are used to configure the repository contacts proxy devices in order to gather information on contract service level metrics (SLM).

Automatic proxy reuse

Repository is capable in configuring multiple contracts to use the same proxy object if these apply the same set of policies. If you want to have a different proxy object (endpoint at the end) for each contract,then uncheck this setting.

Destroy orphan proxies

Repository deletes unused proxy objects.

> **Note:** When a service WSDL/endpoint is changed proxy objects are automatically updated without any further action. By default this update is performed by an automatic action defined within the service lifecycle.

Hide functional endpoints if managed are available

This setting affects the user interface only; if a functional endpoint has it's managed counterpart available then the functional one might be hidden; this is useful when you don't want regular users to access the functional endpoint directly but through PEP only. This setting affects the main display of endpoints/implementations only, users will be still able to access the functional endpoint via relationships (navigator, rest, ui relationship browser)

Allow manual / non-governed deployment actions for non-admin users

Uncheck this if you don't want users to use the manual deployment wizard and force them to follow the defined contract lifecycle process.

Protocol handler regexp

By default the automatic publishing process makes the proxy object available through all the protocol handler the proxy device has available. Here you can define regexp filter for the protocol handler name that will define which protocols new proxies will be available to.

Path template

This settings shows how managed endpoint urls are constructed. You can specify your own default path which can contain following place holders:

${environment}	name of the environment where functional endpoint resides.
${contractID}	contract ID which is the managed endpoint. To utilize this functionality you should uncheck the Automatic proxy reuse checkbox.
${path}	path of the original functional endpoint
${date}	current date in $(YYYYMMDD) format

Proxy layout

The contract to be deployed might be used to consume multiple different services. Some devices allow to create a single proxy object to handle multiple services per a proxy object. In that case the layout might be:

1. **shared** - there will be created a proxy object per single contract

2. **per service** – there will be create new proxy object per service available under the contract

Identity retrieval

Defines the method of association of messages by the proxy object with contracts.

Individual Contract Monitoring

It is possible to monitor individual contract performance even when multiple contracts share the same proxy object. To see the report of contract performance open the 'Reports' tab and in the left menu choose the 'Contract SLO Monitoring' link. A window will pop up:

You can see how the message per day metric evolves in time. The data are gathered from devices in predefined intervals (see the Contract Deployment Settings documentation) and data older than a day are compacted so there is just one record for a day/contract within Systinet database.

Certain adapters also need some preconfiguration to support this feature so referrer to the documentation of the adaptor you are using to get more information.

When you click on the 'Analyze link' in the header of a service Provider you can also compare service capacity (messages processed since certain date in the past) consumed by different contract of the service. Check the screenshot below:
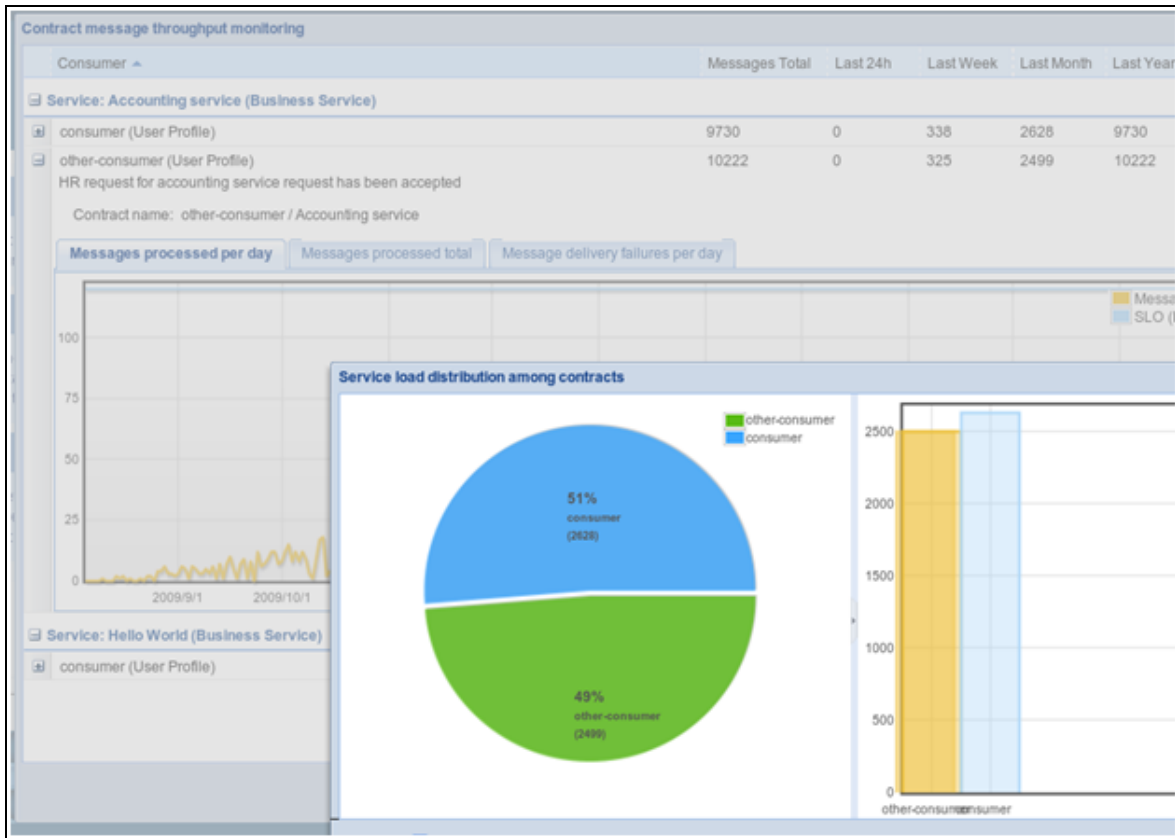
Setting Contract Development Lifecycle Process

From the Administration page, select Lifecycle Processes from the Administration list.

- Open Default Contract Lifecycle Process:
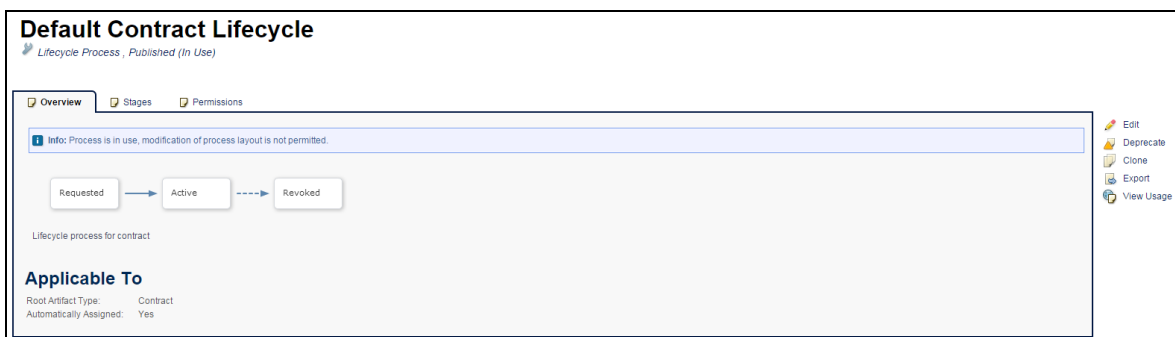
- Go to **Stages** tab and select Active stage. Add Deploy/undeploy automatic action for this stage, check the "Deploy to Production" environment:
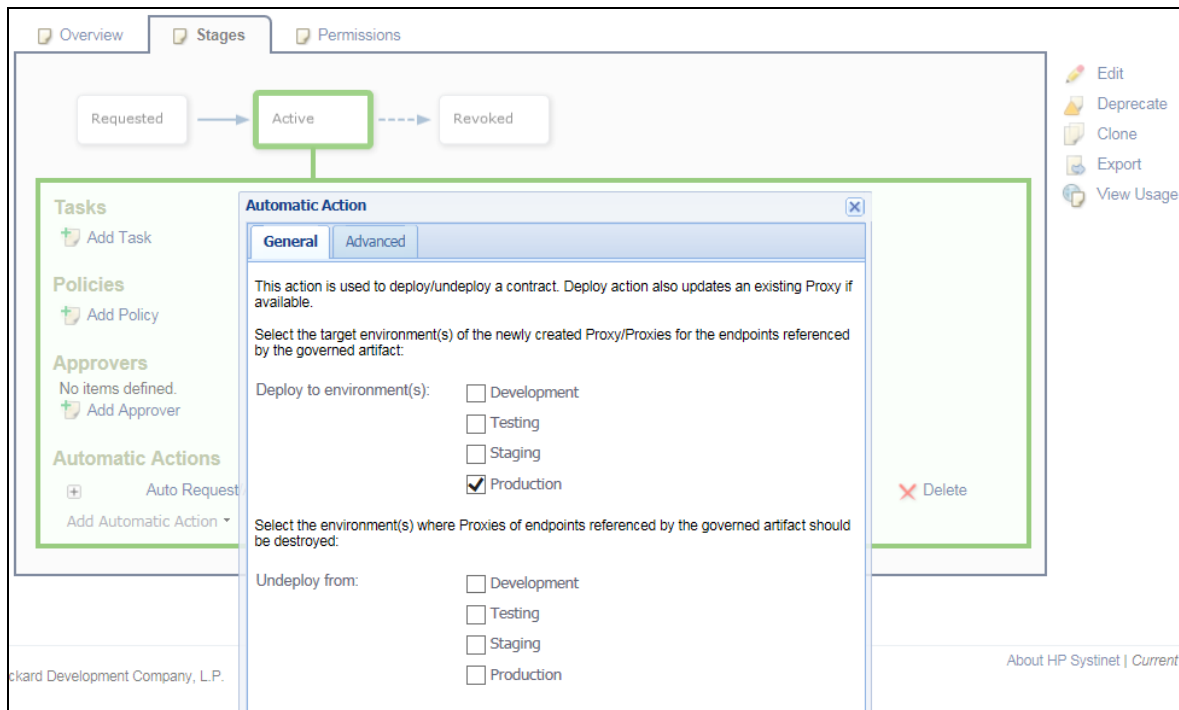
- Publish the process.

Publishing a Service

For demo purposes there is demo service available under PLATFORM/demo/proxy_mgmt/hello_
service and start it.

Check the service WSDL under http://localhost:6060/demo/basic/HelloService/wsdl.

Go to catalog and under "Import" click "File" and fill the form as in the following image:
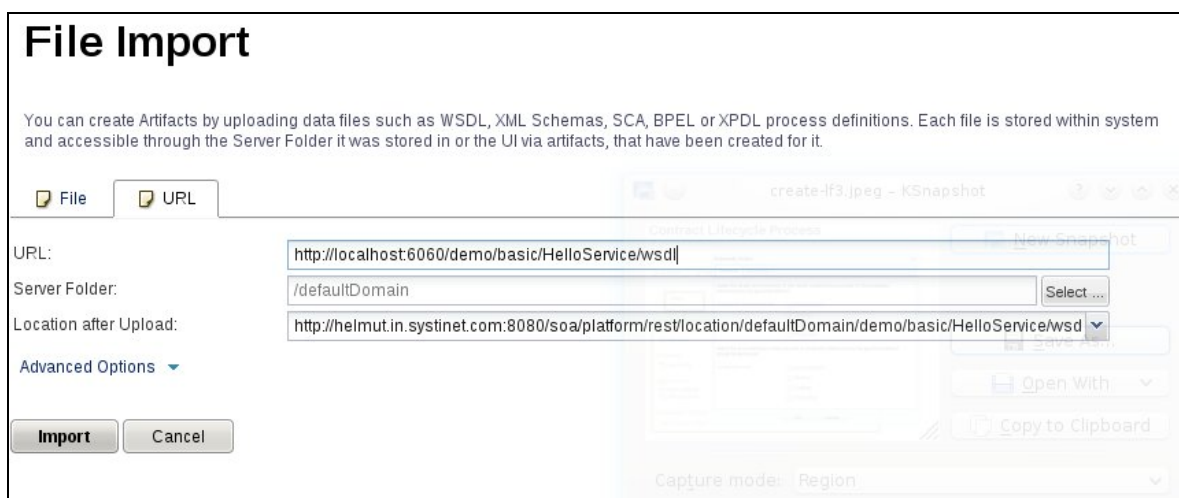
Defining Service SLO

Open the Hello service and create a Default SLO artifact for it. (Open the Hello service detail, switch to "Details" tab and in the bottom there is a "Create" link) under SLO.

Name it "Default SLO" and leave the settings untouched – defaults will be used. Using the SLO you can override the settings from the section 2.3 and you can associate policies with a contract.

Creating Contracts

Create a contract request for the hello service. You will need to make the service consumable first. (see more in the contract management documentation on this).

Don't forget to associate the proper SLO with the contract.

Request approval in the "Stage" lifecycle stage . The approval will be automatically granted. On the background the proxy object for the hello service will be created.

The contract overview page will look like the following:

Accessing Service Through Proxy

Start the SOAP UI tool available from http://www.soapui.org/

Create new project and as initial WSDL/WADL use the wsdl url property from the contract page view in the Systinet repository as depicted on (Fig. A).

Notice that there is a ContractID associated with the contract. As we have chosen the ContractID HTTP header style of identification of contract on the proxy object we will have to pass this string in the "ContractID" http header to the proxy.
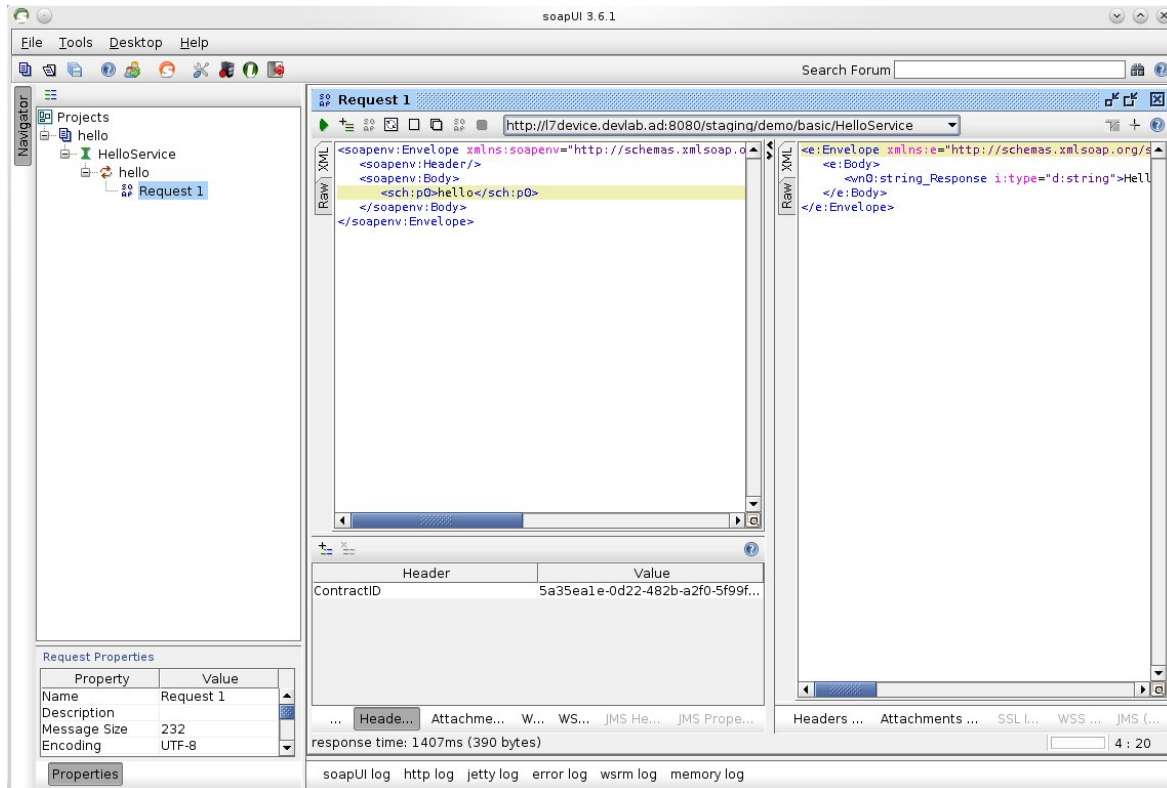
Working with Policies

SLO objects might be associated with runtime policies. Runtime policy is an artifact that is created by importing it / synchronizing it with a policy from an RGIF device. Policies are handled differently by different devices. You need to check the particular device manual to be aware of it's specifics and limitations.

Policies that can be downloaded from the device into a repository, must follow this naming convention:their names must end with the "_policy" string. Otherwise the device ignores them.

- "Creating Policies" below

- "Using Policies" on the next page

- "Policy Versioning" on the next page

Creating Policies

To import a policy from the device into repository:

- Click on the **Administration** tab.

- On the left menu under **RGIF Run-time** click **Runtime Policies**. Following page appears:

Runtime Policy Synchronization Status
Report

Here you can see only runtime policies on the Device that are recognized by Systinet (i.e having "_policy[_version_x]" suffix).

Legend:

UNSYNCHRONIZED   Policy has been found on the device but it is not known to the repository.
SYNCHRONIZED   Policy has been imported into the repository and it contents is the same as on the device.
CHANGED   Policy has been imported into the repository but it's contents is different than on the device.
OBSOLETE   Policy has been imported into the repository but it can't be found on any device.

You can delete individual policies but only the repository representation is removed. The policy on the device remains untouched.

🔄 Synchronize Into Repository   ✗ Delete From Repository

Name ▲	Version	Device Native ID	Synchronization Status	☐
Device: L7 Server				
⊞ bbktrinh	2	bbktrinh	Unsynchronized	☑
⊞ bbktrinh_l7_201	1	bbktrinh_l7_201	Unsynchronized	☐
⊞ dthung	Not versioned	dthung	Unsynchronized	☐
⊞ Hoa333	Not versioned	Hoa333	Unsynchronized	☐
⊞ Layer7 Integration Policy1	Not versioned	Layer7 Integration Policy1	Unsynchronized	☐
⊞ Layer7 Integration Policy2 ver1	1	Layer7 Integration Policy2 ver1	Unsynchronized	☐
⊞ Layer7 Integration Policy2 ver2	2	Layer7 Integration Policy2 ver2	Unsynchronized	☐

- You will get a list of all policies available on all devices known to Systinet. Those marked as **Synchronized** are already imported into Systinet others are not. To import those policies from Device to Repository, you can select them and click on **Synchronize Into Repository** button. The whole definition of the policy is uploaded into repository and can be later uploaded into a different device (this is done automatically by Systinet if it is needed).

Using Policies

You can associate runtime policies with SLO artifacts that are linked to contracts. Once the contract is deployed it's associated policies are configured into the proxy object created for the contract.

Note that policies linked to a SLO are also divided into groups by environment. Each link has a property called useType (it is available using DQL) which is equal to the value from the environment taxonomy. By default Systinet includes a preconfigured user interface where only Stage and Production environments are available for association with policies on the SLO overview page.

You can customize Systinet to support policies for other environments as well. All you need to do is to switch to UI customization mode and duplicate the setup of the production environment policy links, and change the environment labels and taxonomy key.

Policy Versioning

Systinet Repository integration with RGIF devices supports policy versioning – you will need to use the following naming convention:
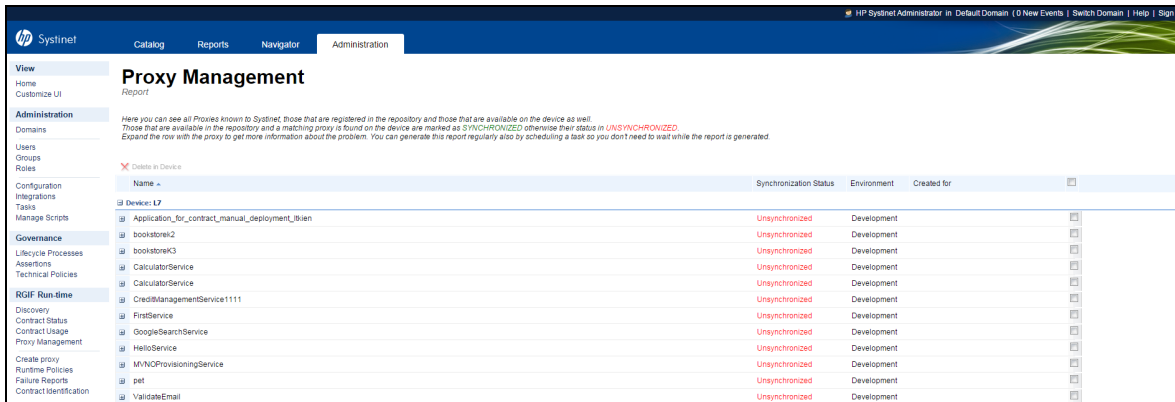
- Policy objects on the devices that have names ending with _policy are thought to be non versioned.

- Policy objects that have names ending with policy_version_{version number} (like policy_version_1, policy_version_22) are thought to be versioned.

Non versioned policies are expected to be the same across all the devices and changing it in one place means that it should be changed in all other places where it is used. It is the responsibility of the administrator to assure this. When a contract using such policy is deployed, the policy already available on the device is automatically used. If it is not available the last synchronized revision of the policy artifact is used to reconstruct to policy object on the device.

Using versioned policies on the other hand gives you the possibility to use different version of the same policy across multiple contracts. Also you will be able to track those versions using the UI.

Managing Proxies

As an administrator you can review proxies and their status by opening "**Administration tab**" and under "**Integrate Run-time**" click "**Proxy Management**". Following report appears:

Proxies are categorized as:

Synchronized	Proxies marked as Synchronized are both correctly registered in the repository and in the devices.
Unsynchronized	Proxies marked as Unsynchronized indicate that there is a mismatch between the information in the repository and in the device.

The best way to resolve such a state is by deleting the proxy and rebuilding it manually.

> **Note:** Rogue services can be detected using the dialog described in the previous section; these are those marked as unsynchronized and available on the device but not in repository.

Creating Manual Proxies

You can create proxies manually using a wizard when you can specify all the parameters for the new proxy object.

To create a proxy manually:

1. Use one of the following:
   - Navigate to **Administration** tab-> **RGIF Run-time** menu on left side and click **Create proxy**.

   - The Deploy button on a contract page if the '**Manual deployment for non administrators**' is allowed in the Contract management setting for a particular domain.

   The wizard starts with a selection of the artifact you want to create the proxy object for. Usually it is a contract but it can be some other artifact that directly or indirectly references endpoint

artifacts.

2. Specify the parameters for the new proxy:
   a. Specify the frontend protocol used by the proxy. It must be preconfigured on the device first. The newly created endpoint will use this protocol. To create a frontend protocol on a device you need to follow the proper device documentation.

   b. Provide a unique name for the new proxy object. You can find the proxy object by this name in both the repository and the RGIF device.

   c. Choose a preconfigured contract id retrieval methods. To define a new identity retrieval method see (link).

3. Finally you are prompted to confirm all the settings before the proxy is finally created. You can also customize the URL of the new managed endpoint here.

Working with Proxies

SLO objects might be associated with runtime policies. Runtime policy is an artifact that is created by importing it / synchronizing it with a policy from a proxy device. Policies are handled differently by different proxy adapters. You need to check the particular device manual to be aware of it's specifics and limitations.

It is common that the policies that can be downloaded form the device into repository need to follow some naming conventions, typically their name must end with the "_policy" string., otherwise the adapter ignores them.

Enabling XI50 Remote Configuration

You will need to explicitly enable remote management of the XI50device.

1. Log into the default domain as **admin**.

2. Open the "**Objects**" menu on the left

3. Scroll down to "**Device management**"

4. Click "**XML management interface**".

5. Fill the upcoming dialog as follows and click apply.

Enabling L7 Remote Configuration

You will need to explicitly enable remote management of the L7 device. In the left bottom corner click on the device name and open the popup menu. Click **Create internal service**. The following screen is displayed. Click **Finish**.
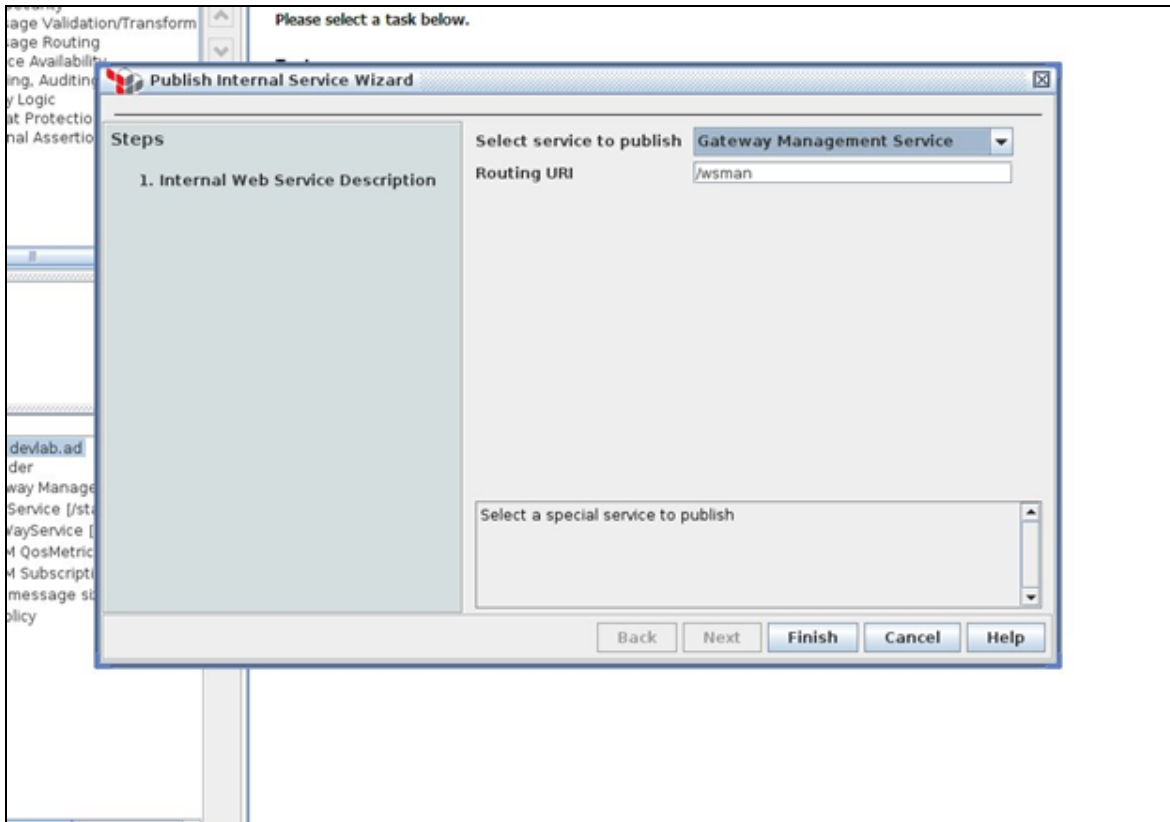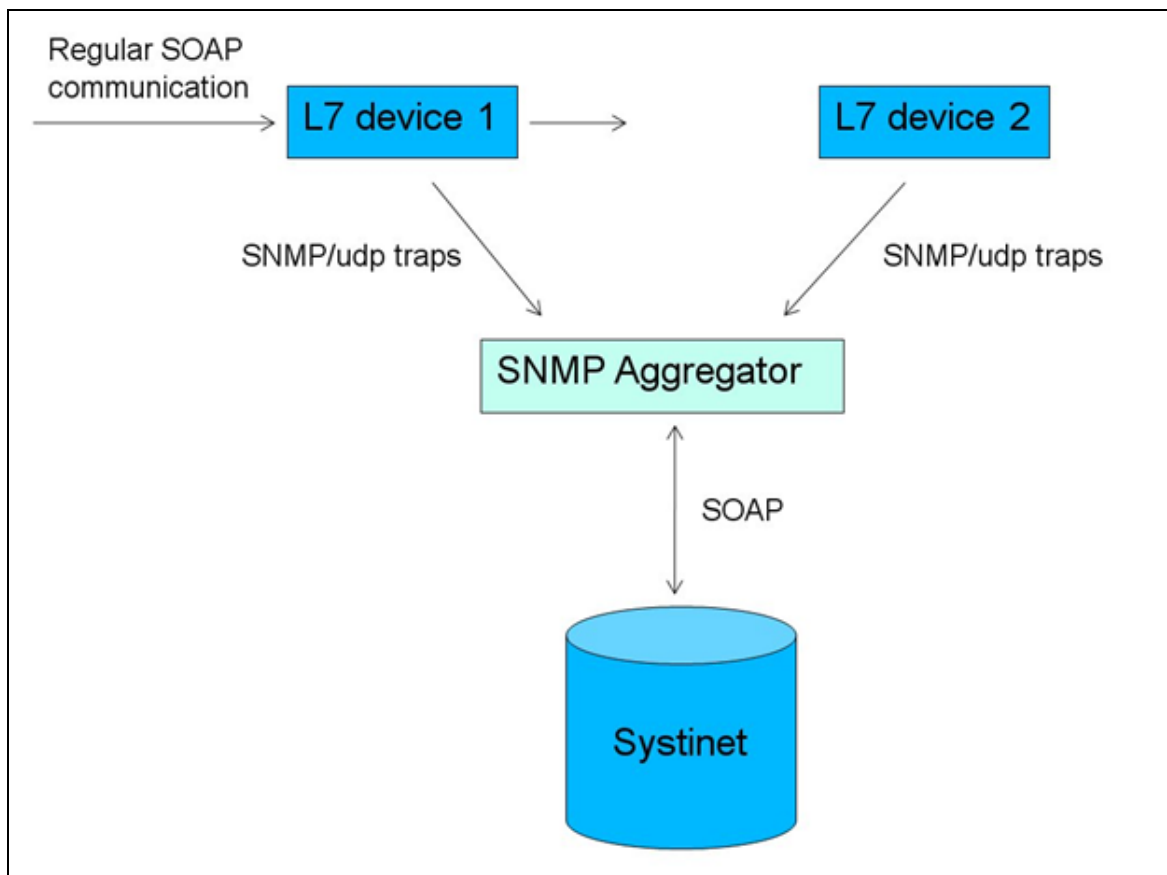
Environment limitations and setup

1. The Layer 7 device requires that you run Systinet with Java version 1.6 or later.

2. Systinet can be run on JBoss versions 6.2 or 7.1.

L7 SLM Handling

Layer 7 device doesn't support per contract SLM monitoring directly, however Systinet provides functionality how to get over this obstacle. See the following figure:

You can see that L7 device is generating SNMP traps for the messages passing through the device, these traps are aggregated in a separate SNMP aggregation server which is part of Systinet installation. Systinets then regularly fetching the aggregated information about contracts from the aggregator server. The format of the SNMP trap message which the aggregator understands is:

{contractID= 6441d6c0-9705-473e-89fa-ce23946be003;deviceId= 8441d6c0-9705-473e-09fa-ce2e946be0ee;type=OK}

The 'contractID' is the contract UUID in Systinet repository, 'deviceId' is the device UUID in the repository and type is one of the following fixed strings: OK/ERROR/THROTTLED.

To start the aggregator, go to PLATFORM_HOME/bin and start the following:

snmp_aggregator.sh –capture

This starts the server with the default settings – listening on the port 1620 for SNMP traffic, listening on port 6080 for communication with Systinet. You can override these defaults with extra command line parameters, check snmp_aggregator.sh –help

When you use the aggregator for the first time it is useful to enter the –debug parameter which is logging all the incoming SNMP traffic.

You also need to make Systinet aware of the aggregator; go to Contract Deployment Settings and check the SLM settings:

The SNMP Host and SNMP Port is used to configure L7 devices where the SNMP traps will be sent to. The Aggregator is the address of the aggregator web service interface where information about contract performance can be obtained. All the SLM settings are defined in the context of the top level domain – when changed within a sub domain configuration the top level domain will be updated as well. This true for all the other adapter not just Layer7.

Customizing L7 Proxy Objects

You might customize the L7 service policies created by Systinet. Open the Contract deployment settings and check the Adapter specific settings/L7 proxy template. See the figure below:
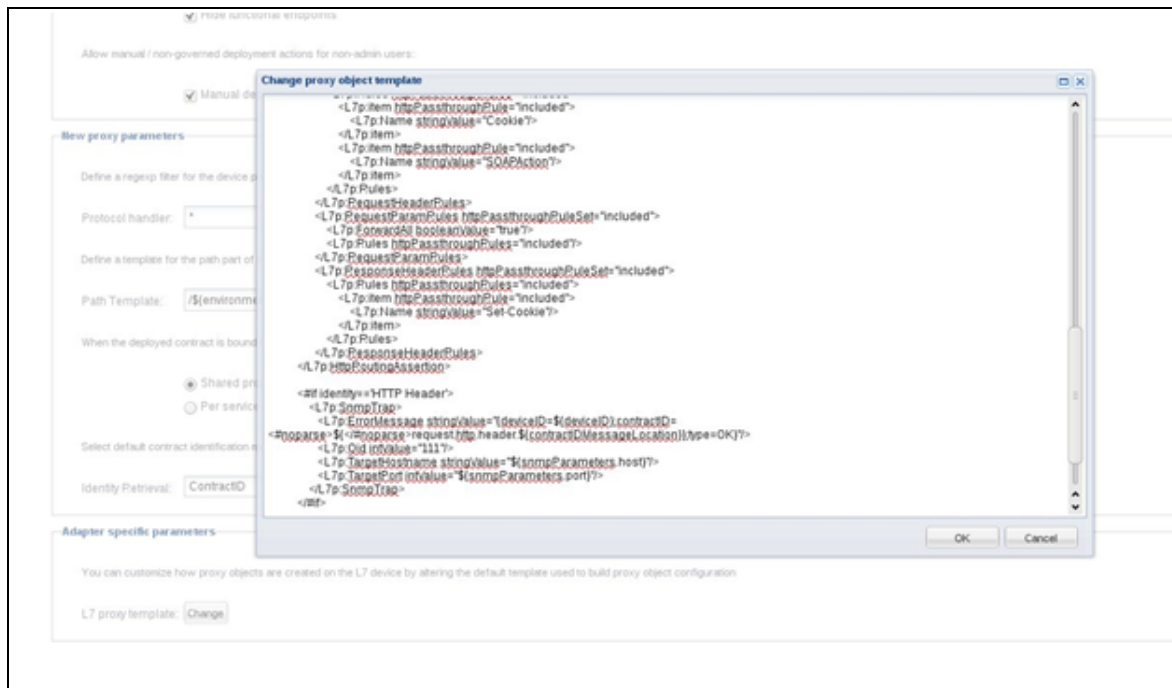
The template itself is based on FreeMaker. Refer to http://freemarker.sourceforge.net/ to get more information about the syntax of the template.

Chapter 11: Product Integration

This part contains the following topics:

Concepts

Tasks

- UDDI Integration Tasks

- BSM / UCMDB Integration Tasks

- ALM Integration Tasks

Product Integration Overview

Systinet integrates with other products to serve as a central point of governance and to view all the information about your content.

Product Integration covers the following main uses cases:

- Discover and import the content of the integrated products so that you can govern and provide that content within Systinet.

- Publish and share the content of the Catalog with integrated products so that you can apply the features of those products.

- Access the information provided by the integrated products within Systinet.

The administrator is responsible for managing product integration within Systinet. For details, see "Product Integration Management" in the *Administration Guide*.

The exact details for each product vary but server management is handled in a generic way. The administrator creates a server artifact using a URL and a set of default credentials. Depending on the product and the configuration of Systinet, other users can use these default credentials to access the server or store their own in their credentials store. For details, see "How to Manage Integrated Products" in the *Administration Guide* and "How to Manage Your Credentials" on page 52.

Systinet provides integration and support for the following products:

- **UDDI Registry**

  Use UDDI Registries as central storage for your service infrastructure or for specific environments such as development and production registries. Integration with UDDI Registries enables integration with other products (for example, SAP) through the registry.

  You can import, export, and synchronize your service artifacts with entities in the registry. For details, see "UDDI Registry Integration" on page 295.

- **HP Business Service Management**

  Discover services stored in *HP Business Service Management* (BSM) and *Universal Configuration Management Database* (UCMDB) and enter them into governance. These discovered services are then monitored for changes and you can synchronize any changes from UCMDB into Systinet.

  Use BSM to monitor the performance of your services and view information generated by BSM on shared services in service detail pages. For details, see "BSM / UCMDB Integration" on page 325.

  > **Note:** HP Business Availability Center (BAC) 8.02 is no longer supported.

- **HP Application Lifecycle Manager**

Use *Application Lifecycle Manager* to track your service testing and view information generated by ALM on shared services in Systinet. For details, see .

Systinet also integrates with *LDAP* for user management and *SiteMinder* for user authentication. The management of these integrations is administered during installation or with the Setup Tool. For details, see "Setting Up LDAP Integration" and "Setting Up SiteMinder Integration" in the *Administration Guide*.

HP Software provides plugins for IDEs that enable the use of Systinet features within development environments. In particular, Systinet Plugin for Eclipse can apply to 3rd-party Eclipse-based tools (such as *SAP Netweaver Developer Studio* and *Tibco Business Studio*) as a further integration point. For more details, see *Plugin for Eclipse* and *Plugin for Visual Studio*.

Systinet Product Family

Systinet is part of a larger family of SOA products designed to cover the entire range of SOA requirements.
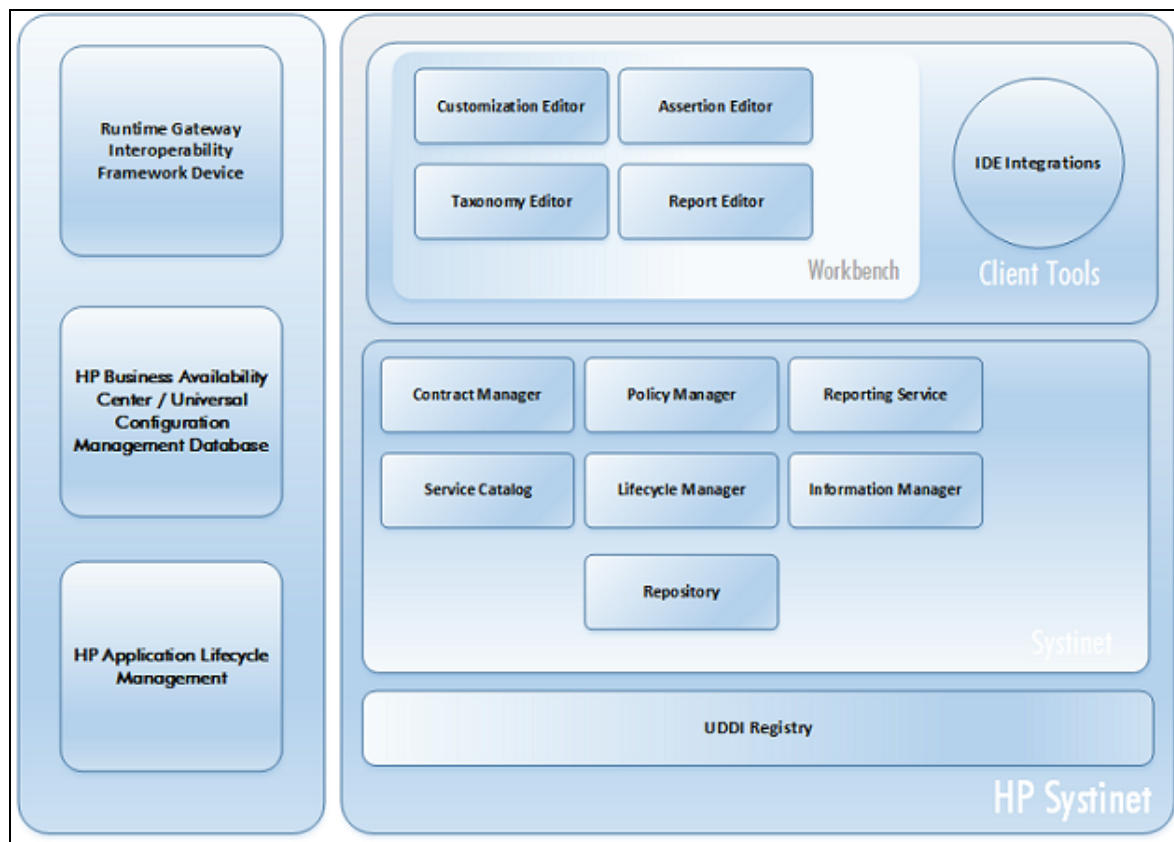
Standard Components

Systinet is designed to enable application governance to be established in a production environment with the following standard components:

- **Catalog**

  Provides an entry point for the provision and management of services.

- **Information Manager**

  Provides a suite of governance tools and low level access to the repository.

- **Reporting Service**

  Monitors and reports on your data.

- **Repository**

  Organizes and manages your data.

- **Lifecycle Manager**

  Enables you to control the service lifecycle and establish governance procedures for each lifecycle stage.

Optional Components

In addition, your Systinet license can include the following components:

- **Policy Manager**

  Enables your environment to conform to your business policy.

- **Contract Manager**

  Enables the management of provider-consumer relationships.

Tools

HP Software provide a set of tools for use with Systinet:

- **Workbench**

  A suite of editor tools distributed as an Eclipse development platform or as a plugin for Eclipse.

  Workbench includes the following components:

  - **Customization Editor**

    Enables you to customize the architecture model and in Systinet. For details, see the *Customization Editor Guide*.

  - **Assertion Editor**

    Enables you to create the building blocks of policies. For details, see the *Assertion Editor Guide*.

- **Taxonomy Editor**

  Enables you to create customized taxonomies to categorize your artifacts according to your needs. For details, see the *Taxonomy Editor Guide*.

- **Report Editor**

  Enables you to create customized reports. These can be deployed as a reporting extension and then used in Systinet. For details, see the *Report Editor Guide*.

- **IDE Integrations**

  A set of plug-ins for different development environments offering Systinet integration functionality.

  HP Software provide the following IDE integration products:

  - **Systinet Plug-In for Eclipse**

    Enables you to search the Catalog, generate service clients and skeletons from Systinet resources, perform local resource validation against Systinet policies, and publish local resources to the Catalog. For details, see *Plugin for Eclipse*.

  - **Systinet Plug- In for Visual Studio**

    Enables you to search the Catalog , generate web references from Systinet resources, and publish local resources to the Catalog. For details, see *Plugin for Visual Studio*.

Integrated SystinetProducts

Systinet integrates with the following products:

- **HP SOA Registry Foundation**

  Publish and index your services in a central location. For details, see the *Registry Product Documentation*.

- **HP Business Service Management** (formerly *Business Availability Center*) and **HP Universal Configuration Management Database**

  Discover services stored in UCMDB and access service availability statistics generated by BSM to monitor actual performance against your service level objectives.

- **HP Service Test Management**

  Register your SOAP services in ALM for testing and validation.

UDDI Registry Integration

Systinet enables you to use UDDI Registries as central storage for your service infrastructure or for specific environments such as development and production registries.

Systinet splits responsibility for registry integration between administrators and users with permissions to transfer content between the Catalog and an integrated registry server.

The administrator manages integrated registry servers and must create an integrated server artifact before UDDI integration functionality is available for use. For details, see the following topics in the *Administration Guide*:

- "How to Manage Integrated Products"

- "How to Synchronize Registry Taxonomies"

> **Caution:** If HTTPS is used for Systinet–HP SOA Registry Foundation communication, then it is necessary to import the registry certificates into the application server certificate store. For details, see "SSL Certificates" in the Administration Guide.

Users with appropriate permissions can perform the following content management actions involving an integrated registry server:

- "How to Import from Registry" on the next page

- "How to Synchronize with Registries" on page 303

- "How to Export to Registry" on page 305

- "How to Delete Data from Registry" on page 318

A UDDI registry is an implementation of the UDDI specification, for example HP SOA Registry Foundation. The UDDI specification has three major versions, commonly named v1, v2 and v3. Systinet is interoperable with UDDI v3 compliant registries.

The UDDI specification defines four major entities:

- **Business Entity**

  A business entity represents a business unit, company, department, and so on. It contains one or more company names, contacts, and provided services. A business entity in registry corresponds to an Organizational Unit in the Systinet SDM model.

- **Service**

A service represents a logical service. Services cannot stand alone, they must always be part of a parent Business Entity. A service in registry corresponds to an Implementation in the Systinet SDM model.

- **Binding Template**

  A binding template represents technical services. It includes information needed to create and run client applications. A binding template in registry corresponds to an Endpoint in the Systinet SDM model.

- **tModel**

  A tModel represents an arbitrary resource, that cannot be described by the structures above. For example; specification, documentation, (part of) WSDL document, policy or taxonomy. Therefore there is not a common map of tModel to an SDM model artifact.

> **Tip:** The mapping for certain types of tModel can be defined in Systinet Workbench. For details, see the *Customization Editor Guide*.

The UDDI specification defines interoperable standards for the exchange of data about web services, their interfaces, implementations, deployments, and responsible contacts. For details, see http://uddi.xml.org.

For details about *HP SOA Registry Foundation*, see the Registry Product Documentation.

How to Import from Registry

Systinet enables you to import services and associated entities from a UDDI Registry.

To Import Data from a UDDI Registry:

1. In the Catalog tab Import menu, place the cursor over **UDDI** to display the integrated UDDI servers and select the registry you want to import from to open the Import from UDDI dialog box.

2. **Optional:** Click **(Change)** to modify the UDDI connection credentials. You can add new credentials and preserve them in your credentials store. For details, see "How to Manage Your Credentials" on page 52.

3. Select to display **My Entities** or **All**. My Entities refers to the credentials you use to connect to the Registry.

4. Use the entity type tabs to select the UDDI entities to import. Use the name filter to locate a particular entity.

> **Note:** You can only select from one type of entity. Systinet imports all related entities

dependent on your selection. If you are importing a business entity, all its services are imported as well, and if you are importing a service, all its binding templates are also be imported. If a single business entity is imported then input a service name and description for the new business service artifact to associate with the imported items.

5. *Optional:* Expand **Advanced Options** and select whether the imported entities are automatically synchronized.

> **Note:** You can create a synchronization task to update Systinet with changes to registry data. For details, see "How to Synchronize with Registries" on page 303.

6. Click **Import** to start the UDDI Import task.

This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description
100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet.This prevents the client from timing out and assuming the request was lost.
200 OK	Standard response for successful HTTP requests.

201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.
226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.
301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.
306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.

307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" below below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.
406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.
409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.
414 Request-URI Too Long	The URI provided was too long for the server to process.

415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.
422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.
428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.

444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.
499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.

502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products
Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain

Content-Type	multipart/form-data	Administrator	Top-level Domain

Note: HP recommends updating Oracle Database schema statistics after importing large amounts of data. Old statistics may impact the performance of some data queries. Consult your database administrator.

To Update Oracle Schema Statistics:

- Execute the following command:

   EXEC DBMS_STATS.GATHER_SCHEMA_STATS (ownname => '&1',no_invalidate => FALSE,options => 'GATHER');

   This command does not require database admin privileges and can be run by the schema owner (ownname).

The report shows the artifacts created or updated by the import, the registry entity type and key and the result of the import for that artifact.

Any imported endpoints are categorized according to the following rules:

- If the endpoint is categorized by a keyed reference in the registry, the same environment category is applied to the endpoint in Systinet.

- If the endpoint is uncategorized in the registry, if **only** one environment category applies to the registry artifact in Systinet, the same environment category is applied to the endpoint in Systinet. If the registry artifact is uncategorized or categorized by multiple environments, no category is applied to the endpoint in Systinet.

For more details about environments, see "Environments" on page 111.

How to Synchronize with Registries

When Systinet integrates with a UDDI Registry, each artifact that corresponds to a UDDI entity contains a **Synchronization** component in its Artifact Details page Overview tab.

Screenshot: Synchronization Component
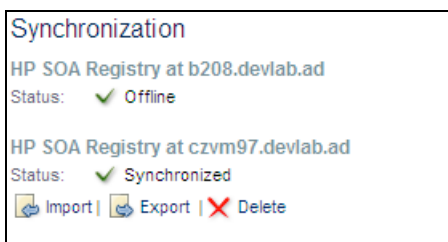
For each registry, the artifact synchronization status is shown and synchronization actions are available.

Synchronization Status

Synchronization Status	Description
Not Synchronized	Systinet artifact does not correspond to any UDDI entity. It is not exported to UDDI or imported from UDDI. This synchronization status corresponds to the NEW resource status.
Synchronized	Systinet artifact and corresponding UDDI entity are semantically the same, both are the same since the last synchronization. This synchronization status corresponds to the IDENTICAL resource status.
Local change	Systinet artifact has changed (while the corresponding UDDI entity has not) since the last synchronization. This synchronization status corresponds to the LOCAL CHANGE resource status.
Remote change	The corresponding UDDI entity has changed since the last synchronization. This synchronization status corresponds to the REMOTE CHANGE resource status.
Local+Remote change	Both the Systinet artifact and the corresponding UDDI entity have changed since the last synchronization. This synchronization status corresponds to the NEEDS MERGE resource status.
Remote Entity Not Found	The corresponding UDDI entity does not exist or has been deleted. This synchronization status corresponds to the UNREACHABLE resource status.
Offline	The UDDI registry is not available to check the synchronization status.

Use the following functions to synchronize Catalog and Registry data.

- **Import**

  Update the Catalog artifact with the data from Registry.

- **Export**

  Update the Registry data with the artifact from the Catalog.

- **Delete**

  Delete the data from the Registry.

The administrator can schedule an automatic synchronization task for each registry.

To Schedule an Automatic Registry Synchronization Task:

1. In the Artifact Details page of the Registry Server, click the **Schedule Automatic Sync** context action to open the Schedule Automatic Sync dialog box.

2. Set the schedule details as described in "Scheduling Tasks" in the *Administration Guide*.

The task can be managed from the Registry page or the Administration Tasks page. For details, see "Administration Task Management" in the *Administration Guide*.

By default, the task finds all artifacts imported from the registry that are marked for automatic synchronization and updates the existing Catalog content with any changes found in the registry.

How to Export to Registry

You can export artifacts to an integrated registry individually or in bulk.

To Export an Individual Artifact to a UDDI Registry:

1. In the Artifact Details page Overview tab, in the **Synchronization** component, for the Registry that you want to export to, click **Export** to open the Export to UDDI dialog box.

> **Note:** Only organizational unit, service, implementation and WS-Policy artifacts can be exported.

2. *Optional:* To use alternative credentials, click **(Change)** to open the Select Authentication dialog box and select alternative credentials.

> **Note:** For a successful export, the credentials used for registry sign-on must have the appropriate write permissions for the registry entities being created or amended.

3. *Optional:* To change the parent business entity, click **(Change)** to open the Select Parent Business Entity dialog box and select an alternative.

4. Click **Export** to start the export process.

   This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

   The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description
100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet. This prevents the client from timing out and assuming the request was lost.
200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.

226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.
301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.
306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" below below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.

406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.
409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.
414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.

420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.
422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.
428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.
444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".

451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.
499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.
502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.

507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products
Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain
Content-Type	multipart/form-data	Administrator	Top-level Domain

Note: Systinet automatically exports related sub-artifacts.

To Export a Set of Artifacts to a UDDI Registry:

1. In any table of artifacts, select the artifacts to export, and then select **Export to Registry** to open the Export to UDDI dialog box.

> **Note:** Artifacts associated with the selected artifacts are included in the export with the exception of endpoints. Only endpoints with the same environment category as the registry server artifact are exported.

2. Select the registry server to export to and optionally change the credentials to use. You can add new credentials and preserve them in your credentials store. For details, see "How to Manage Your Credentials" on page 52.

3. *Optional:* For artifacts other than Organizational Units, select a Business Entity to associate the exported entity with in the registry.

4. *Optional:* Expand **Advanced Options** and select **Overwrite Business Entities** to change any previously specified business entities for the artifacts selected for export.

5. Click **Export** to start the export process.

This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description
100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet. This prevents the client from timing out and assuming the request was lost.

200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.
206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.
226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.
301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.

306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" on page 307 below, but specifically for use when authentication is required and has failed or has not yet been provided.
402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.
406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.
409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.

414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.
416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.
422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.
428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.

444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.
499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.

502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products
Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain

Content-Type	multipart/form-data	Administrator	Top-level Domain

The report shows the artifacts exported, their registry entity type and key, and the result of the export for that artifact.

For exported binding templates, the WSDL repository URL is used by default with the following exceptions:

- If the registry artifact is categorized for an environment then an environment specific WSDL URL is used.

- If the registry artifact is not categorized for an environment then the WSDL URL from the repository REST location space is used.

To disable the use of repository URLs for WSDLs and enforce the use of the originURL (where the WSDL was imported from) the administrator must set the following property:

`platform.integration.uddi.useRepositoryUrlOnWsdlExport=false`

Consider this option if the repository is not your primary artifact store. To set this property, see "How to Manage System Settings" in the *Administration Guide*.

How to Delete Data from Registry

Entities in a synchronized registry can be deleted from the registry directly from Systinet.

To Delete an Entity from UDDI Registries:

1. In Artifact Details page Overview tab, in the Synchronization component, for the Registry that you want to delete the entity from, click **Delete** to open the Delete from UDDI dialog box.

   Note: The delete functionality is also available from the detail view of artifacts already deleted from the Catalog.

2. *Optional:* To use alternative credentials, click **(Change)** to open the Select Authentication dialog box and select alternative credentials.

   Note: For a successful deletion, the credentials used for registry sign-on must have the appropriate write permissions for the registry entities being deleted.

3. Click **Delete** to start the deletion process.

   This process executes as a bulk operation. An information bar opens informing you that the operation is in progress with a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes. For more details, see "How to Use Bulk Operations" on page 159.

> **Note:** Systinet provides some OOTB HTTP Status Code artifacts and HTTP Header artifacts which can be used while publishing WADL or Swagger files. The new HTTP Status Code or HTTP Header artifact will be created only when its code number or header value is found in the existing list.

HTTP Status Code Artifacts

The following is the list of OOTB HTTP Status code artifacts:

HTTP Status Codes	Description
100 Continue	This means that the server has received the request headers, and that the client should proceed to send the request body
101 Switching Protocols	This means the requester has asked the server to switch protocols and the server is acknowledging that it will do so.
102 Processing	This code indicates that the server has received and is processing the request, but no response is available yet.This prevents the client from timing out and assuming the request was lost.
200 OK	Standard response for successful HTTP requests.
201 Created	The request has been fulfilled and resulted in a new resource being created.
202 Accepted	The request has been accepted for processing, but the processing has not been completed.
203 Non-Authoritative Information (since HTTP/1.1)	The server successfully processed the request, but is returning information that may be from another source.
204 No Content	The server successfully processed the request, but is not returning any content. Usually used as a response to a successful delete request.
205 Reset Content	The server successfully processed the request, but is not returning any content. Unlike a 204 response, this response requires that the requester reset the document view.

206 Partial Content	The server is delivering only part of the resource (byte serving) due to a range header sent by the client. The range header is used by tools like wget to enable resuming of interrupted downloads, or split a download into multiple simultaneous streams.
207 Multi-Status	The message body that follows is an XML message and can contain a number of separate response codes, depending on how many sub-requests were made.
208 Already Reported	The members of a DAV binding have already been enumerated in a previous reply to this request, and are not being included again.
226 IM Used	The server has fulfilled a request for the resource, and the response is a representation of the result of one or more instance-manipulations applied to the current instance.
300 Multiple Choices	Indicates multiple options for the resource that the client may follow.
301 Moved Permanently	This and all future requests should be directed to the given URI.
302 Found	This is an example of industry practice contradicting the standard.
303 See Other	The response to the request can be found under another URI using a GET method.
304 Not Modified	Indicates that the resource has not been modified since the version specified by the request headers If-Modified-Since or If-None-Match.
305 Use Proxy	The requested resource is only available through a proxy, whose address is provided in the response.
306 Switch Proxy	No longer used. Originally meant "Subsequent requests should use the specified proxy.
307 Temporary Redirect	In this case, the request should be repeated with another URI; however, future requests should still use the original URI.
308 Permanent Redirect	The request, and all future requests should be repeated using another URI.
400 Bad Request	The server cannot or will not process the request due to something that is perceived to be a client error.
401 Unauthorized	Similar to "403 Forbidden" on the next page below, but specifically for use when authentication is required and has failed or has not yet been provided.

402 Payment Required	Reserved for future use. The original intention was that this code might be used as part of some form of digital cash or micropayment scheme, but that has not happened, and this code is not usually used.
403 Forbidden	The request was a valid request, but the server is refusing to respond to it. Unlike a "401 Unauthorized" above response, authenticating will make no difference.
404 Not Found	The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.
405 Method Not Allowed	A request was made of a resource using a request method not supported by that resource; for example, using GET on a form which requires data to be presented via POST, or using PUT on a read-only resource.
406 Not Acceptable	The requested resource is only capable of generating content not acceptable according to the Accept headers sent in the request.
407 Proxy Authentication Required	The client must first authenticate itself with the proxy.
408 Request Timeout	The server timed out waiting for the request.
409 Conflict	Indicates that the request could not be processed because of conflict in the request, such as an edit conflict in the case of multiple updates.
410 Gone	Indicates that the resource requested is no longer available and will not be available again.
411 Length Required	The request did not specify the length of its content, which is required by the requested resource.
412 Precondition Failed	The server does not meet one of the preconditions that the requester put on the request.
413 Request Entity Too Large	The request is larger than the server is willing or able to process.
414 Request-URI Too Long	The URI provided was too long for the server to process.
415 Unsupported Media Type	The request entity has a media type which the server or resource does not support.

416 Requested Range Not Satisfiable	The client has asked for a portion of the file (byte serving), but the server cannot supply that portion.
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field.
418 I'm a teapot	This code was defined in 1998 as one of the traditional and is not expected to be implemented by actual HTTP servers.
419 Authentication Timeout	Not a part of the HTTP standard, 419 Authentication Timeout denotes that previously valid authentication has expired.
420 Method Failure	Not part of the HTTP standard, but defined by Spring in the HttpStatus class to be used when a method failed. This status code is deprecated by Spring.
420 Enhance Your Calm	Not part of the HTTP standard, but returned by version 1 of the Twitter Search and Trends API when the client is being rate limited.
422 Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
423 Locked	The resource that is being accessed is locked.
424 Failed Dependency	The request failed due to failure of a previous request.
426 Upgrade Required	The client should switch to a different protocol such as TLS/1.0, given in the Upgrade header field.
428 Precondition Required	The origin server requires the request to be conditional.
429 Too Many Requests	The user has sent too many requests in a given amount of time.
431 Request Header Fields Too Large	The server is unwilling to process the request because either an individual header field, or all the header fields collectively, are too large.
440 Login Timeout (Microsoft)	A Microsoft extension. Indicates that your session has expired.

444 No Response (Nginx)	Used in Nginx logs to indicate that the server has returned no information to the client and closed the connection.
449 Retry With (Microsoft)	A Microsoft extension. The request should be retried after performing the appropriate action.
450 Blocked by Windows Parental Controls (Microsoft)	A Microsoft extension. This error is given when Windows Parental Controls are turned on and are blocking access to the given webpage.
451 Unavailable For Legal Reasons (Internet draft)	Defined in the internet draft "A New HTTP Status Code for Legally-restricted Resources".
451 Redirect (Microsoft)	Used in Exchange ActiveSync if there either is a more efficient server to use or the server cannot access the users' mailbox.
494 Request Header Too Large (Nginx)	Nginx internal code similar to 431 but it was introduced earlier in version 0.9.4 (on January 21, 2011).
495 Cert Error (Nginx)	Nginx internal code used when SSL client certificate error occurred to distinguish it from 4XX in a log and an error page redirection.
496 No Cert (Nginx)	Nginx internal code used when client didn't provide certificate to distinguish it from 4XX in a log and an error page redirection.
497 HTTP to HTTPS (Nginx)	Nginx internal code used for the plain HTTP requests that are sent to HTTPS port to distinguish it from 4XX in a log and an error page redirection.
498 Token expired/invalid (Esri)	Returned by ArcGIS for Server. A code of 498 indicates an expired or otherwise invalid token.
499 Client Closed Request (Nginx)	Used in Nginx logs to indicate when the connection has been closed by client while the server is still processing its request, making server unable to send a status code back.
499 Token required (Esri)	Returned by ArcGIS for Server. A code of 499 indicates that a token is required (if no token was submitted).
500 Internal Server Error	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable.

501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfil the request.
502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server.
503 Service Unavailable	The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server.
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.
506 Variant Also Negotiates	Transparent content negotiation for the request results in a circular reference.
507 Insufficient Storage	The server is unable to store the representation needed to complete the request.
508 Loop Detected	The server detected an infinite loop while processing the request.
509 Bandwidth Limit Exceeded	This status code is not specified in any RFCs. Its use is unknown.
510 Not Extended	Further extensions to the request are required for the server to fulfill it.
511 Network Authentication Required	The client needs to authenticate to gain network access. Intended for use by intercepting proxies used to control access to the network.
598 Network read timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network read timeout behind the proxy to a client in front of the proxy.
599 Network connect timeout error	This status code is not specified in any RFCs, but is used by Microsoft HTTP proxies to signal a network connect timeout behind the proxy to a client in front of the proxy.

HTTP Header Artifacts

Name	Header Value	Owner	Domain
Content-Type	application/json,	HP Software Developer	HP Software Products

Content-Type	application/xml	Administrator	Top-level Domain
Content-Type	application/x-www-formurlencoded	Administrator	Top-level Domain
Content-Type	multipart/form-data	Administrator	Top-level Domain

BSM / UCMDB Integration

HP Business Service Management (formerly Business Availability Center) (BSM) monitors run-time services and collects statistics on their performance. These statistics enable you to verify that a service meets its service level objectives. BSM uses the *Universal Configuration Management Database* (UCMDB) which provides visibility of the infrastructure and applications that support your business services.

Systinet splits responsibility for BSM / UCMDB integration between administrators and users with permissions to access UCMDB content and BSM performance statistics.

The administrator manages integrated BSM / UCMDB servers and must create an integrated server artifact before BSM / UCMDB integration functionality is available for use. For details, see the following topics in the *Administration Guide*:

- "How to Manage Integrated Products"

Note: Systinet can integrate with only one BSM/UCMDB server which is associated with the top-level domain.

Caution: If the Catalog already contains services discovered from BSM/UCMDB, do not delete the BSM/UCMDB server artifact. You may lose access to service discovery functions and discovered artifacts. If your BSM/UCMDB connection settings change, modify the properties of the existing BSM/UCMDB server.

Users with appropriate permissions can perform the following content management actions involving an integrated BSM / UCMDB server:

- "How to Discover BSM / UCMDB Content" on the next page

- "How to Resolve Matching Conflicts" on page 329

- "How to Mark Discovered Content as Rogue" on page 329

- "How to Ignore Discovered Content" on page 330

-

-

Systinet artifacts correspond to UCMDB entities as follows:

Systinet Artifact	UCMDB Entity
Organizational unit	Business Unit
Service	Business Service for Catalog
SOAP Service	Web Service

Tip: BSM can access services in Systinet directly if the connection to Systinet is set in *BSM for Systinet Settings*. This enables you to view SOAP Services in the Health Report page. For details, see the *HP Business Service Management* documentation.

How to Discover BSM / UCMDB Content

Systinet enables you to import services from the *Universal Configuration Management Database* (UCMDB) used by *HP Business Service Management* (BSM).

Note: To perform UCMDB discovery, a BSM/UCMDB server must be integrated with Systinet. For details, see "How to Manage Integrated Products" in the *Administration Guide*.

Tip: To take full advantage of this feature, HP recommends integrating UCMDB with *HP Discovery and Dependency Mapping Inventory* (DDMI). DDMI can identify your physical and virtual IT assets, collect hardware information, and identify installed and used software. When integrated with UCMDB, DDMI can automatically populate UCMDB with this information.

To Import Artifacts from BSM/UCMDB:

1. In the Catalog tab Import menu, click **UCMDB** to open the UCMDB Import page.

2. Click **Start Import**.

   A task progress information bar opens with a link to the discovery process report.

   The task consists of the following stages:

   a. Read the data from the BSM/UCMDB server.

   b. Store the discovered data in the Catalog.

   c. Update relationships between artifacts in the Catalog.

Discovery executes asynchronously so you can continue with other work while the task finishes.

The discover content process checks the Catalog for matching artifacts and resolves potential conflicts. The default rules for a conflict are determined by a configuration file. The default behavior checks for artifacts with matching `name` properties. For SOAP Services, Systinet checks for matching `serviceName` and `serviceNamespace` properties before checking the `name`.

A matching Service, SOAP Service, or Organizational Unit in the Catalog is assumed to be the same entity discovered in BSM/UCMDB. The artifact properties in the Catalog version are overwritten by the properties of the discovered entity. These properties are defined by a mapping file.

Any existing relationships with the existing repository artifact are preserved with the exception of previously discovered entities. For example, a Service created in Systinet is implemented with a SOAP Service previously discovered in BSM/UCMDB. The same service in BSM/UCMDB has the SOAP service removed and is then discovered. Systinet removes the relationship to that SOAP Service.

In the event that there are two or more artifacts in Systinet that conflict with a discovered entity, you must resolve the conflict manually as described in .

After discovery, the UCMDB Import page is populated with a list of content from the BSM/UCMDB server that is not already in governance in Systinet, or has not been marked as rogue or ignored.

Screenshot: UCMDB Import Page

Note: Any discovered endpoints are categorized with the same environment category applied to the BSM/UCMDB server artifact.

Note: Systinet 4.0 automatically creates Service artifacts associated with SOAP Services imported from BSM / UCMDB. In 3.x, these Service artifacts are only created when the imported SOAP Service is entered into governance. If you have SOAP Service artifacts which were migrated from HP SOA Systinet 3.x, then create appropriate Service artifacts and relations to the imported SOAP Services before entering them into governance.

The imported ungoverned content is only visible to you and you are expected to process the imported artifacts. For details, see the following sections:

- "How to Resolve Matching Conflicts" on the next page

  When the discovery process discovers multiple instances of matching artifacts already in the Catalog, you must resolve the conflict manually.

- "How to Mark Discovered Content as Rogue" on the next page

  Mark discovered content as requiring special attention for review later.

- "How to Ignore Discovered Content" on page 330

  Mark discovered content as not required and ignore them in subsequent discoveries.

- "How to Start Governance" on page 247

  Systinet does not govern discovered content automatically. To apply a lifecycle process to discovered content, use the Governance menu in the UCMDB Import page

> **Note:** These functions are not available if there is a discovery task in progress.

After discovery, services are monitored for changes and can be synchronized. For details, see "How to Synchronize Discovered Content" on page 330.

> **Note:** HP recommends updating Oracle Database schema statistics after importing large amounts of data. Old statistics may impact the performance of some data queries. Consult your database administrator.

To Update Oracle Schema Statistics:

- Execute the following command:

  EXEC DBMS_STATS.GATHER_SCHEMA_STATS (ownname => '&1',no_invalidate => FALSE,options => 'GATHER');

  This command does not require database admin privileges and can be run by the schema owner (ownname).

You can schedule UCMDB Import to execute at a set time or on periodic basis. In the UCMDB Import page, click **Schedule Automatic Import** and set the execution details. For details, see "How to Schedule Tasks" in the *Administration Guide*.

If a task already exists, the **Schedule Automatic Import** link is replaced by **Edit Schedule** and **Cancel Schedule** links.

To Stop a Running UCMDB Import task:

- In the UCMDB Import page, while the import task is in progress click the Stop context action.

The task does not stop immediately. The import of data stops, but Systinet ensures that the data imported is consistent by updating the relationships between stored artifacts. If you execute discovery again, the data not imported the first time is re-discovered and then imported.

How to Resolve Matching Conflicts

When UCMDB Import runs, Systinet checks the imported content to see if it matches any existing content in the Catalog. In the event that there is more than one matching artifact, these artifacts are indicated as a conflict in the UCMDB page and you must resolve them manually.

To Resolve Matching Conflicts:

1. In the UCMDB Import page, for the conflicted artifact, click **Merge Duplicates** to open the Merge Duplicates dialog box.

2. Use the **Detail** or **Comparison** links for each matching artifact to view their details or a detailed comparison with the imported artifact.

3. Do one of the following:

   ▪ Select **Do not Merge Artifact** to leave the imported artifact as a separate entity.

   ▪ Select a matching artifact to merge the imported artifact with it.

4. Click **OK** to perform the selected action.

If you selected to merge the artifact with a governed artifact, then it is removed from the UCMDB Import page.

How to Mark Discovered Content as Rogue

Some of the discovered artifacts may require special attention. Marking them as rogue adds them to a list of rogue artifacts which you can review at a later date. Systinet ignores them in subsequent discovery processes.

To Mark Discovered Artifacts as Rogue:

1. In the Catalog tab Import menu, click **UCMDB** to open the UCMDB Import page.

2. Select the artifacts to mark as rogue, click **Mark as Rogue**, and confirm your decision.

The UCMDB Import page re-opens and the rogue artifacts are removed from the list of discovered artifacts.

> **Note:** To review the list of rogue artifacts, in the UCMDB Import page, click **View Rogue**. In the list of rogue artifacts, you can **Revert** the artifact to its originally imported state or mark it as Ignored.

How to Ignore Discovered Content

Some discovered artifacts may be internal parts of third party products or low level infrastructure. Marking them as ignored makes them invisible in the Systinet UI and ignores them in subsequent discovery processes.

To Mark Discovered Artifacts as Ignored:

1. In the Catalog tab Import menu, click **UCMDB** to open the UCMDB Import page.

2. Select the artifacts to mark as infrastructure, click **Ignore**, and confirm your decision.

The UCMDB Import page refreshes and the ignored artifacts are removed from the list of discovered artifacts.

> **Note:** To review the list of ignored artifacts, in the UCMDB Import page, click View Ignored. In the list of ignored artifacts, you can Revert the artifact to its originally imported state or mark it as Rogue.

How to Synchronize Discovered Content

Artifacts discovered in BSM/UCMDB and entered into governance in Systinet are monitored for changes.

The Synchronization section of the Artifact Details page Overview and Metrics tabs of a discovered artifacts displays a BSM Server section and its synchronization status:

Screenshot: Synchronization Component

Click **Detail** to view more information about the status and source of the discovered content.

Click **Synchronize** to update the Systinet artifact with the latest changes from BSM/UCMDB.

How to Monitor BSM Performance Statistics

Any SOAP Service artifact monitored by BSM has an extra section in its Artifact Details page Metrics tab:

Screenshot: Performance and Availability Section

The Performance and Availability section displays statistics generated by BSM for the default time period. Select **Data Source** to refresh the information for the selected time period. Select **Real Data** to show metrics generated by actual users or **Synthetic Data** to show metrics generated automatically by SiteScope, an application monitoring product which you can integrate with BSM. Click **View Report in BSM** to display the BSM report for the selected time period.

Screenshot: BSM Health Report
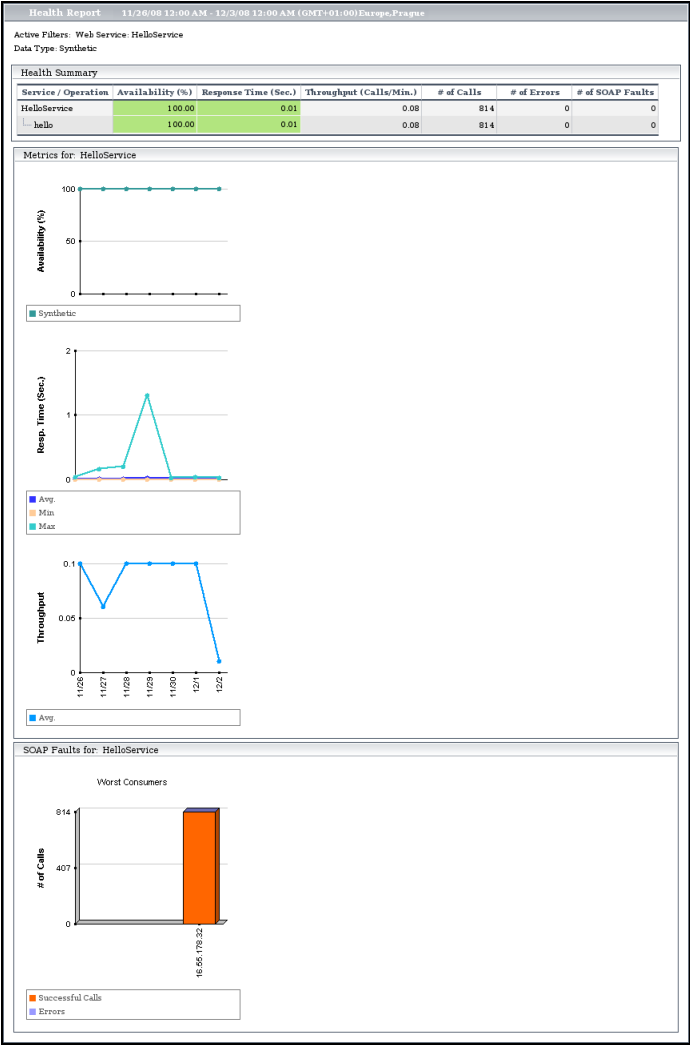
ALM Integration

Systinet enables to register the SOAP services in ALM 11 for testing or imports Requirements from ALM 12, and then monitors the testing statistics in Systinet.

To customize Systinet to work with later ALM versions:

1. Access the page **Administration** > **Configuration** > **System Settings**.

2. Download the property file: `platform.integration.alm.configuration`

3. Modify this file by adding a new `almServiceFetcher` for the new version of ALM.

4. Upload this file again to Systinet.

Systinet splits responsibility for ALM integration between administrators and users with permissions to register services in ALM and access ALM quality statistics.

The administrator manages integrated ALM servers and must create an integrated server artifact before ALM integration functionality is available for use. For details, see the following topics in the *Administration Guide*:

> **Note:** In order to make Systinet artifacts visible in ALM, you must set the Sharing Principal role to use the `system#everyone` group, share the artifacts you want to make visible in ALM, and set sharing for the appropriate stages of lifecycle processes that govern the artifacts. For details, see "How to Change the Sharing Principal" in the *Administration Guide*, "How to Share Artifacts" on page 102, and "How to Define Permissions" in the *Administration Guide*.

Users with appropriate permissions can perform the following content management actions involving an integrated ALM server:

- "How to Register Services in ALM" on page 337

- "Monitor ALM Quality Statistics of Registered Service"

- "How to Import Requirements from ALM 12"

- "Monitor ALM Quality Statistics of Requirement"

Systinet artifacts correspond to ALM entities as follows:

Systinet Artifact	ALM Entity
SOAP Service	Web Service
Requirement	Requirement

> **Tip:** ALM enables you to import services directly from Systinet. For details, see the *HP Application Lifecycle Manager* documentation.

> **Tip:** Systinet provides a set of default technical policies which you can use if you want to make ALM integration and the statistics it provides part of your validation process. For details, see "Default Technical Policies" in the *Administration Guide*.

ALM Server Customizations

You can customize the way that you view data collected from your ALM server by HP Systinet.The HP Systinet server provides three functions to get the service status from the ALM server.

The following procedures list the steps to add and customize your ALM server.

To add the ALM Server:

1. Click **Administration > Servers > ALM Servers > Add ALM Server**. The Add ALM Server page opens.

2. Fill in the required fields based on the type of ALM Server:

   - "Create ALM 11.5x Server Page"

   - "Create ALM 12 Server Page"

3. You can click the **Test Connection** button to test. After the test completes successfully, click **Save**.

To synchronize data with ALM Server:

- "How to Synchronize Imported Requirements from ALM 12"

To Customize the ALM Server:

1. Click **Customization > Customize** in the left navigation area.

2. Click **Recent Documents > GoogleSearchService** (the SOAP service). The selected SOAP service page opens.

3. Click **More > Metrics**. Initially, you will see a warning that no endpoints are defined for the Testing environment type.

4. Select the **Details** tab, and then under Environments, select **Edit**. The GoogleSearchPort_ Endpoint page opens.

5. In the Environment dropdown menu, select **Testing**, and then click **Save**.

6. Click **More > Metrics** and you will now see the ALM server quality presented graphically by default. However, you can customize this page to display different kinds of information.

7. Click **Customize** to view the **Artifact Detail Page** that you will use to embed your customizations.

8. You can choose to customize with any of the following scripts:

- Gets all ALM servers as JSON array by getAlmServers() function:

```
<customization xmlns="http://systinet.hp.com/2009/02/ui/customization"
xmlns:cust="http://systinet.hp.com/2009/02/ui/customization"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="cust artifactDetail.xsd">

<content>

<server id="alm_integration_server">

<import location="/scripts/almServerAPI.js"/>

</server>
```

```
<html id="alm_integration_client">

<include>

<script>

Ext.onReady(function () {

getAlmServers(almServersListener);

function almServersListener(result){

alert(result);

}

});

</script>

</include>

</html>

</content>

</customization>
```

- Gets all ALM beans from ALM server as JSON objects by getAlmBeans(serverUUID, beanProps, runtimeContext) function. Includes:

Name	Description
serverUUID	The ALM server UUID.
beanProps	The name and view name (config in property file of 'platform.integration.alm.configuration' above) of ALM bean as JSON array. For example, var beanProps = [{name:almReq, href:viewRequirement}, {name:almDefect, href:viewDefect}]
runtimeContext	The runtime context to get ALM bean, must include serviceUUID. For example, var runtimeContext = {serviceUUID:alm_integration_client.uuid}

```
<customization xmlns="http://systinet.hp.com/2009/02/ui/customization"
xmlns:cust="http://systinet.hp.com/2009/02/ui/customization"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="cust artifactDetail.xsd">

<content>

<server id="alm_integration_server">

<import location="/scripts/almServerAPI.js"/>

</server>

<html id="alm_integration_client">

<parameter name="uuid">${artifact._uuid}</parameter>
```

```
<include>

<script>

Ext.onReady(function () {

var serverUUID = "xxx";

var beanProps = "[{name:almReq, href:viewRequirement}, {name:almDefect,
href:viewDefect}, {name:almTest, href:viewTest}]";

var runtimeContext = {serviceUUID:alm_integration_client.uuid};

getAlmBeans(serverUUID, beanProps, runtimeContext, almBeansListener);

function almBeansListener(result){

alert(result);

}

});

</script>

</include>

</html>

</content>

</customization>
```

- Gets all ALM beans from all ALM server as JSON object by getServiceStatus(beanProps, runtimeContext) function. Includes:

Name	Description
beanProps	The name and view name (config in property file of 'platform.integration.alm.configuration' above) of ALM bean as JSON array. For example, var beanProps = [{name:almReq, href:viewRequirement}, {name:almDefect, href:viewDefect}]
runtimeContext	The runtime context to get ALM bean, must include serviceUUID. For example, var runtimeContext = {serviceUUID:alm_integration_client.uuid}

```
<customization xmlns="http://systinet.hp.com/2009/02/ui/customization"
xmlns:cust="http://systinet.hp.com/2009/02/ui/customization"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="cust artifactDetail.xsd">

<content>

<server id="alm_integration_server">

<import location="/scripts/almServerAPI.js"/>

</server>
```

```
<html id="alm_integration_client">

<parameter name="uuid">${artifact._uuid}</parameter>

<include>

<script>

Ext.onReady(function () {

var beanProps = "[{name:almReq, href:viewRequirement}, {name:almDefect,
href:viewDefect}, {name:almTest, href:viewTest}]";

var runtimeContext = {serviceUUID:alm_integration_client.uuid};

getServiceStatus(beanProps, runtimeContext, serviceStatusListener);

function serviceStatusListener(result){

alert(result);

}

});

</script>

</include>

</html>

</content>

</customization>
```

How to Register Services in ALM

After an ALM server has been integrated with HP Systinet, you can register SOAP services in ALM 11.5x.

Select the ALM 11.5x option, and then open the ALM services page manually and register the services by yourself manually. For more information, see the ALM documentation.

SOAP Services have an extra Quality section in the Artifact Details page Metrics tab:

Click **Register in ALM** to open a login page for ALM. For details about registering services from Systinet in ALM, see the *Application Lifecycle Management* documentation.

> **Note:** To enable the STM plugin in ALM 11.5x, get the shared artifacts from Systinet. If the Systinet server is installed with LDAP then make sure that **stmintegrator/changeit** user is added in LDAP.

> **Caution:** The SOAP service you want to register must be created in Systinet by uploading a

WSDL that defines it. For details, see "How to Publish Content" on page 127.

The service is registered in ALM using a URL pointing to an environment specific WSDL containing only endpoints (wsdl:port elements) for the specified ALM Server environment. If the ALM server is not categorized for an environment then the original WSDL content is used.

ALM 11.5x

For ALM 11.5x, you must log in to ALM and generate a new module from the registered service of Systinet.

Caution: To be able to register services in ALM you must install this *HP Application Lifecycle Manager* plug-in on the client:

- Microsoft Internet Explorer must be the default browser when accessing ALM.

- Access ALM and allow the client code to install (ActiveX) to the browser.

How to Import Requirements from ALM 12

Perform the following steps to import requirements from ALM 12:

1. Create a new ALM 12 Server page. For details refer to *Administration Guide > Product Integration Management > Integration UI Reference->Create ALM 12 Server Page*.

2. Go to **Catalog** tab > **Import** > **Requirements** in the left panel. The Requirement Import page displays.

3. Select the ALM Server, Domain and the Project from the drop-down list.

4. Click **Next**.

5. Select the requirement by enabling the checkbox and then clicking on **Import** icon.

Following are the Synchronization Status of the requirements:

Synchronization Status	Description
SYNCHRONIZED	Requirement details in repository are the same as in ALM Server.
NEW	Requirement has not been synchronized to the repository.
CHANGED	Requirement is synchronized from ALM Server and it is modified in ALM Server and Systinet.

REMOTE CHANGED	Requirement is synchronized from ALM Server and it is modified in ALM Server.
LOCAL CHANGED	Requirement is synchronized from ALM Server and it is modified in Systinet.

How to Synchronize Imported Requirements from ALM 12

If Systinet is synchronized with ALM 12 server then you can update the imported Requirements from that ALM 12 server with the latest version of corresponding Requirements.

To synchronize a single Requirement from its details page:

1. Open the details page of **Requirement** artifact that is imported from an ALM 12 server.

2. Click **Synchronization** from Synchronization section in the right.
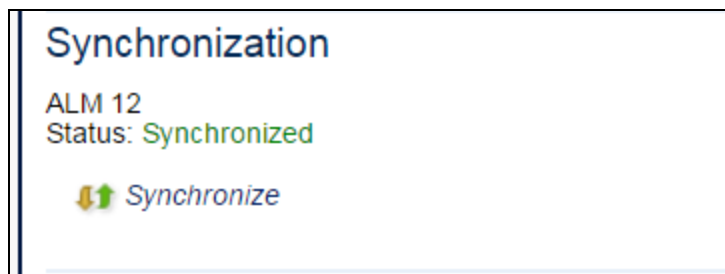
To synchronize a group of Requirements from Requirements collection page:

1. Open the **Requirements** collection page.

2. Select all the Requirements artifact that you want to synchronize.

3. Click **Synchronize** bulk operation. A Synchronize dialog box appears.
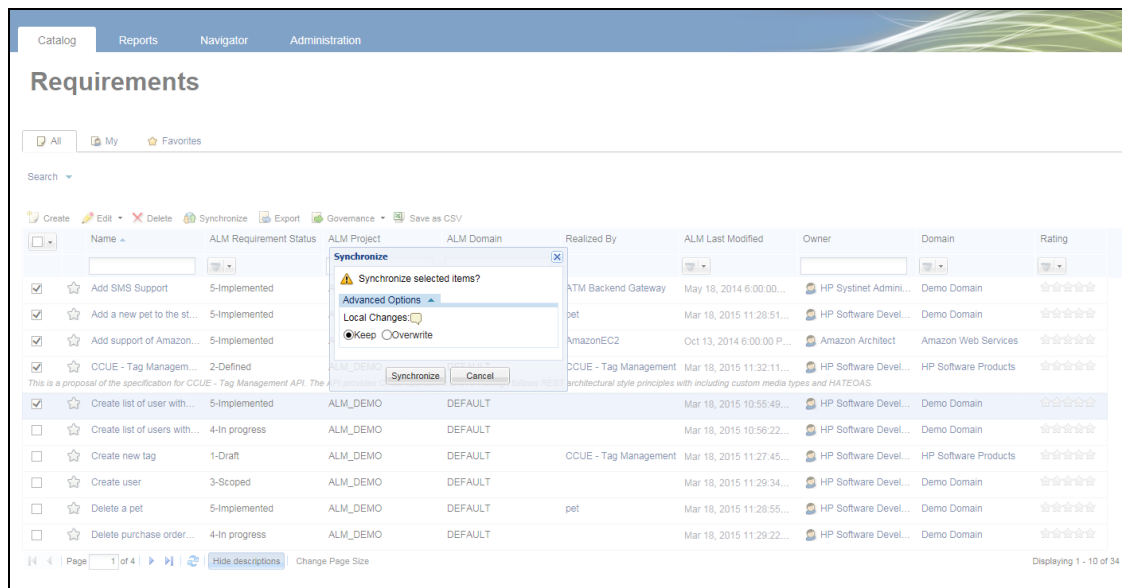
4. Select **Keep** or **Overwrite** local changes.

5. Click **Synchronize**.

To schedule synchronized imported Requirements from an ALM server:

The **Task** to synchronize imported Requirements from an ALM server is created automatically when an ALM server is created. You can use this task to schedule synchronized imported Requirements.

1. Go to **Administration** > **Tasks** > Open the details page of your ALM server Synchronization Task.
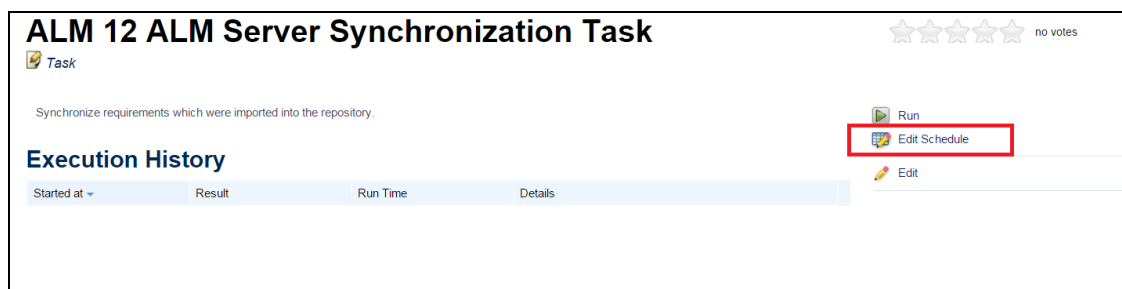
2. Click **Edit Schedule** to edit the task.

HP Software Integrations

Systinet supports integration with the following products:

Product	Version	Features
UDDI Registry	v3	Import/export/synchronize data. Import/Export taxonomies (HP SOA Registry Foundation only).
HP Business Service Management (BSM)	BSM supported versions are: • 9.20 • 9.21 • 9.22 • 9.23 • 9.24	uCMDB service discovery. Service entry to governance/synchronization. BSM health report monitoring. uCMDB supported versions are: • 9.05 • 10.00 • 10.10 • 10.11 • 10.20 **Note:** HP Business Availability Center (BAC) 8.02 is no longer supported.
HP Application Lifecycle Manager	11.50, 11.52, 12.00, 12.01, 12.20	Service test monitoring for ALM 11.50, 11.52 Requirement repository for ALM 12.00, 12.01, and 12.20
Layer7	7.1, 8.1	Service proxy and policy discovery. Contract deployment and status synchronization.
DataPower	6.0.1 7.2.0	Service proxy and policy discovery. Contract deployment and status synchronization.

Monitor ALM Quality Statistics of Registered Service

A registered SOAP service displays testing statistics in the Quality section of the Artifact Details page Metrics tab:

Screenshot: Quality Section
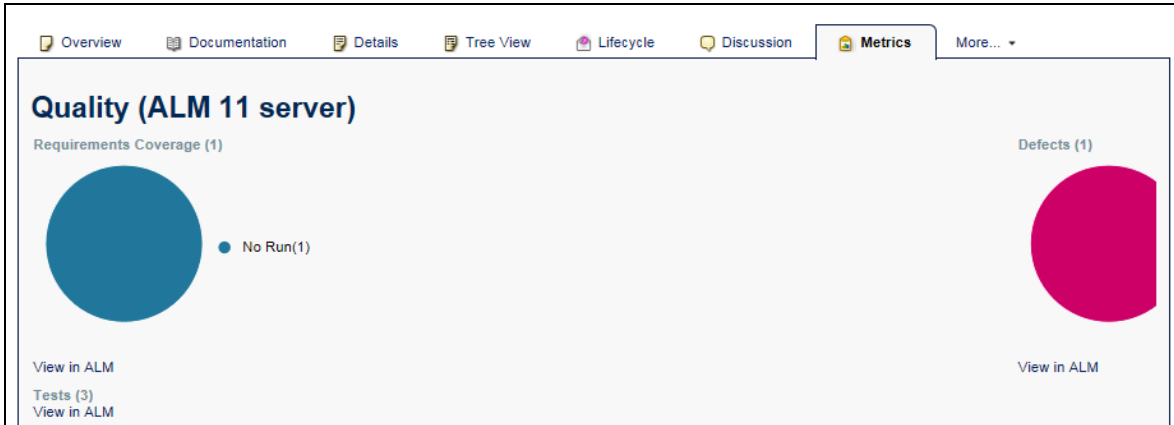
Use the requirement, test and defect links to open the relevant pages in ALM.

For details about registering services in ALM, see "How to Register Services in ALM".

Monitor ALM Quality Statistics of Requirement

The Metrics tab displays testing statistics in the Quality section of the Artifact Details page:
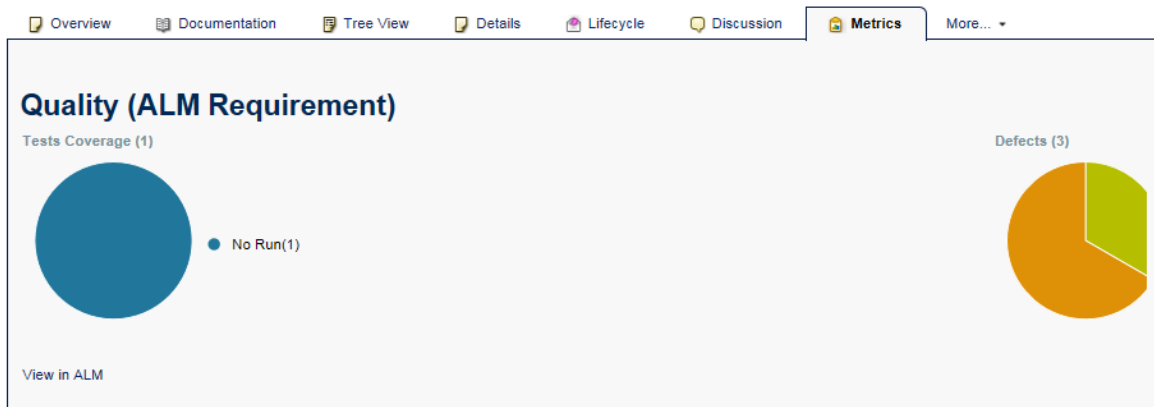
Use the View in ALM link to open the relevant pages in ALM.

Chapter 12: Reporting

This part contains the following topics:

Concepts

- "Reporting Overview"

- "Artifact Reports"

- "Custom Reports"

- "Policy Reports"

Tasks

- "How to Manage Content Feeds" on page 357

- "How to Review Artifact Reports" on page 345

- "How to Review Policy Reports" on page 354

- "How to Create Policy Reports" on page 354

- "How to View Custom Reports" on page 357

- "How to Add Custom Reports" on page 357

- "How to Add Reports to the Menu" on page 346

- "How to Clone Artifact Reports" (on page 1)

- "Customizing Dependency/Impact Report Design" (on page 1)

References

- "Artifact Report Types" on page 346

- "Datasource Configuration" (on page 1)

Reporting Overview

The initial page of the reporting module presents three predefined reports based on contract management, artifacts by lifecycle stage and an activity report. Clicking on contract types or lifecycles stages alters the pie chart to reflect the selection. You may also add relevant and interesting feeds to this page (see "How to Manage Content Feeds" on page 357 in User Guide).
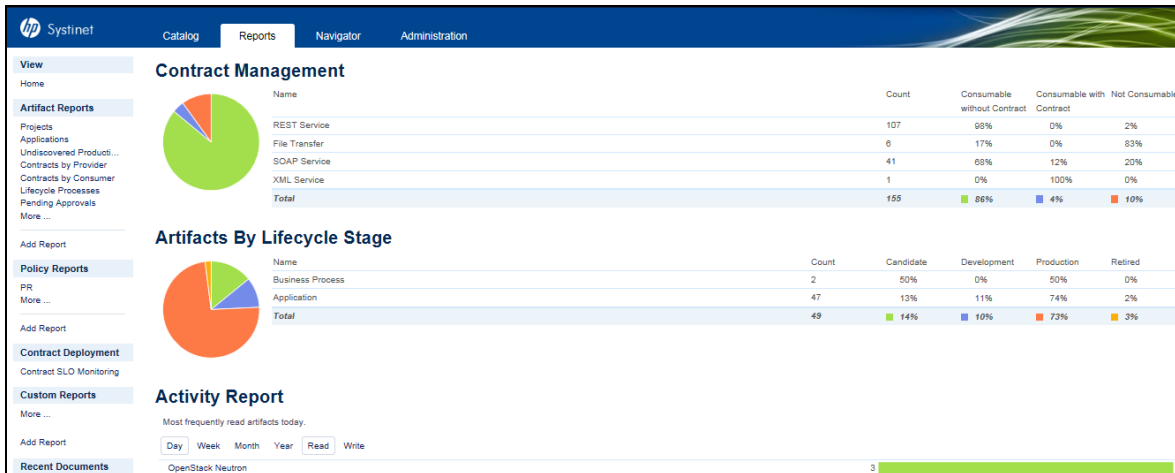
- **Contract Management**
  - Display of specific artifact types, number of artifact instances, number of consumable instances without/with contract, and number of non-consumable artifacts.

  - Administrator can customize which artifact types are captured in these reports.

  - The graph is for a selected line and can be changed by selecting the a different line from the report.

- **Lifecycle**
  - Displays the distribution of artifact instances between lifecycle stages.

  - Administrator can define which artifact types and which lifecycle stages are displayed.

  - The graph is for a selected line and can be changed by selecting the a different line from the report.

- **Users Activity**
  - Displays most frequently read or written artifacts over a specific time period (day/week or month).

Note: The reports on this page can only be modified by users with administrator rights.

There are three report types available in Systinet:
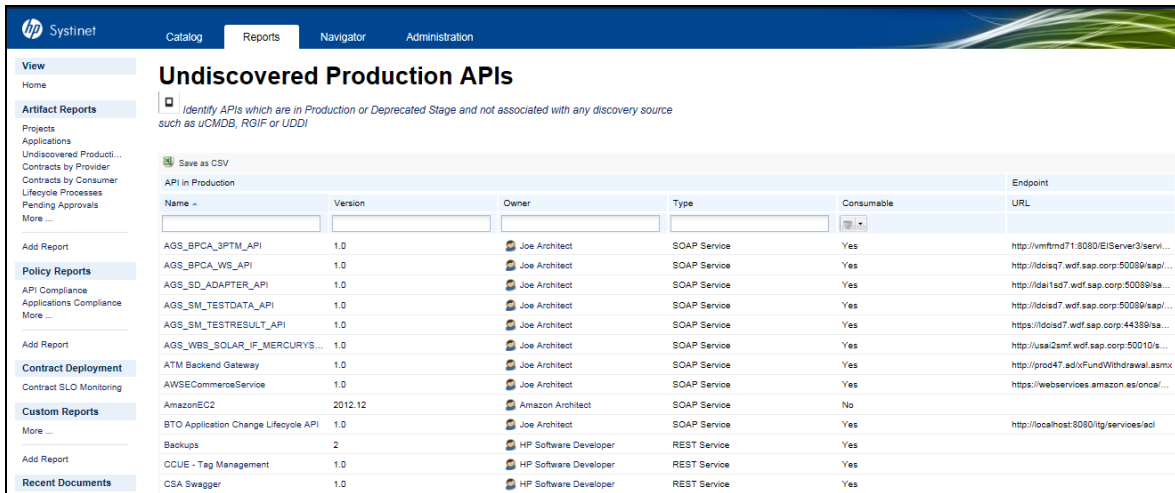
- "Artifact Reports" on the next page

- "Policy Reports" on page 352

- "Custom Reports" on page 355

Artifact Reports

Artifact reports are tabular reports based on DQL. Users can use them together with powerful table capabilities, such as filtering or sorting, in order to present information that cannot be returned via the Search form.

For example, the report *Services by Implementations* lists all services together with their providers, their implementations and implementation endpoints. A user can easily find all the Services from a specific provider that have Implementation in the production and Endpoint of specific type. Alternately, they can find all the Services from a specific domain that have a JMS endpoint.

The Artifact Reports menu contains a default set of reports which can be customized. See "How to Add Reports to the Menu" on the next page for more details.

Content	Description
Provider name	Name of the provider.
Email	Email address of the provider.
API in Production Name	Name of Applications during production.
Version	The version number of the artifact.
Type	Type of services.
Consumable	Select to make the XML service available to consumers.
Endpoint URL	URL for the actual implementation of the service.

How to Review Artifact Reports

All Artifact Reports are presented as a tables and can be sorted and filtered.

To sort a column in a report, click on the column name to switch between 'sort descending and 'sort ascending'. The direction of the sort is indicated by the direction of the light blue arrow next to the column name.

Clicking on the dark blue arrow on the right of a column reveals a dropdown box giving the same options for sorting as well as check boxes that will reveal or hide selected columns.

The *Services by Provider*, *Services by Implementation*, *Lifecycle Processes* and *Pending Approval* reports also offer options for searching within a column and the ability to filter certain columns for specific content.

All reports can be saved as CSV files which can be archived and viewed using standard spreadsheet applications by clicking the **'Save as CSV'** button.

How to Add Reports to the Menu
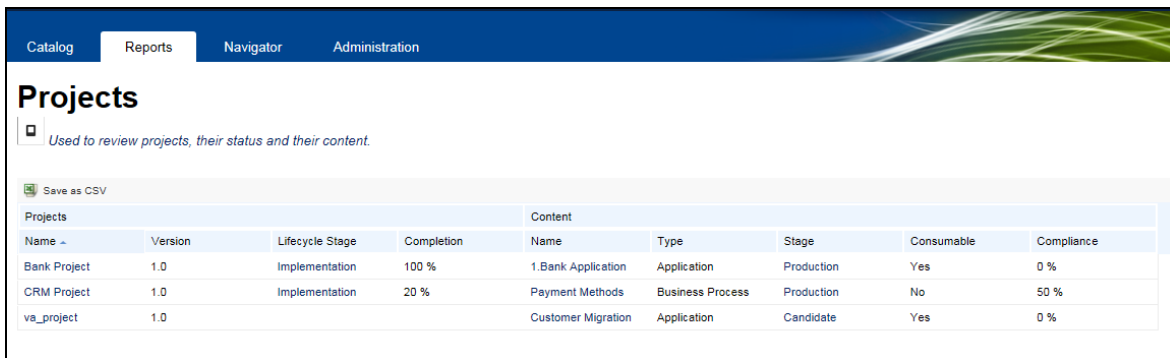
1. From Reports tab Artifact Reports menu, click **More**.

   Existing reports are listed and highlighting the pin icon makes the selected report visible in the menu.

2. Click **Close** to add or remove selected reports or **Restore Defaults** to return to installed status.

Artifact Report Types

Projects - Used to review projects, their status and their content.

Applications - Used to review existing applications, their components and review and filter them by their state (e.g. only production application, only applications with reusable services, etc.).

Processes - Used to identify process providers and the state of processes associated with them (e.g. providers of processes whose compliance is lower than 60%).

Processes By Implementation - Used to identify processes and their implementations like applications – process, implementation, endpoint (e.g. processes with production jms implementation, processes with implementations deployed in staging environment)
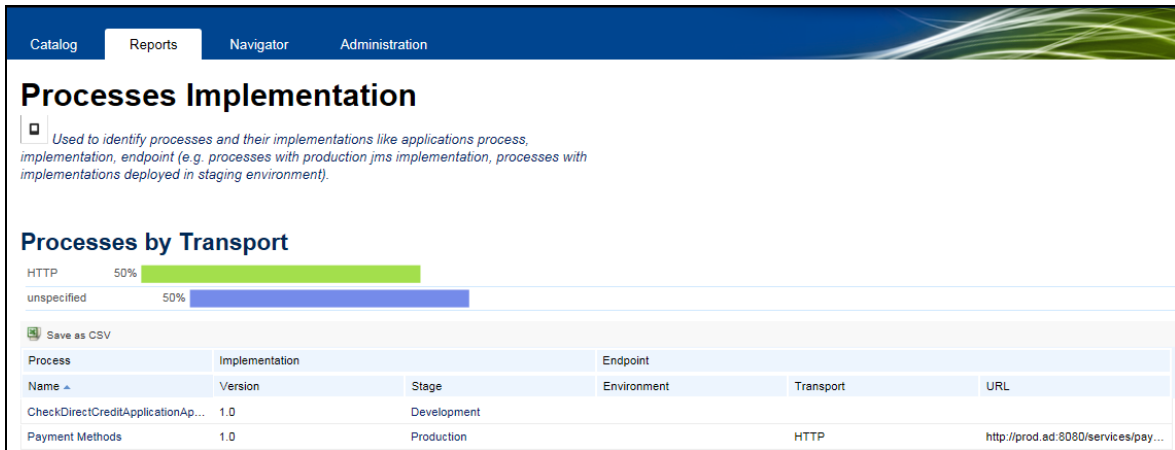
Services By Provider - Used to identify services associated with a provider (e.g. providers of service whose compliance is lower than 60%).
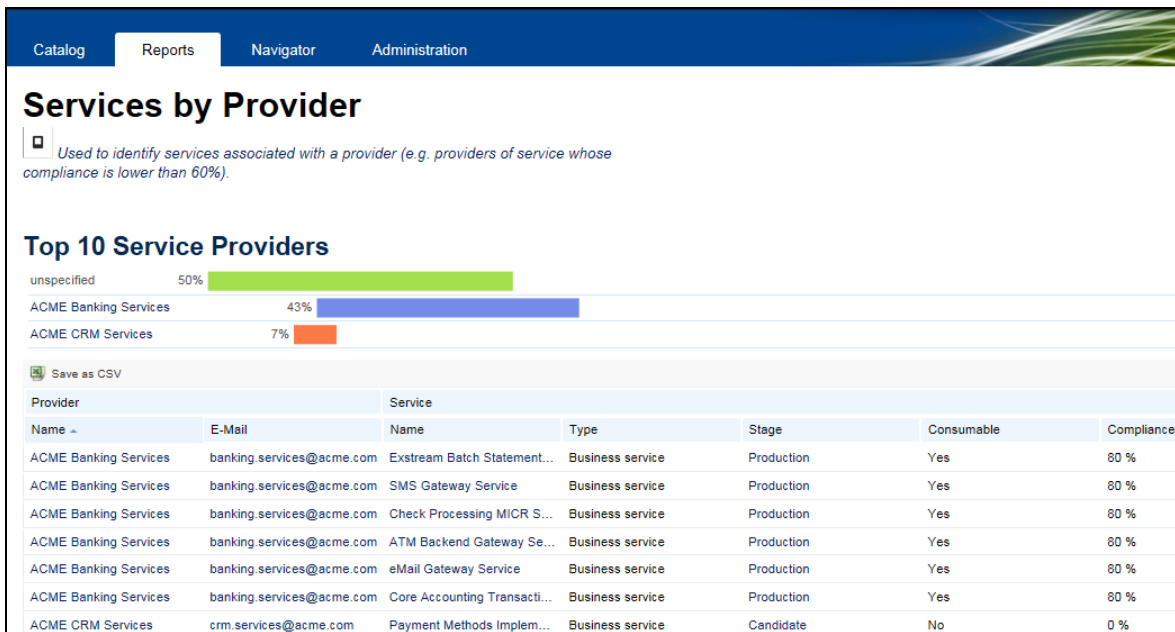
Services By Implementation - Used to identify implemented services, their locations and providers like process by implementation.
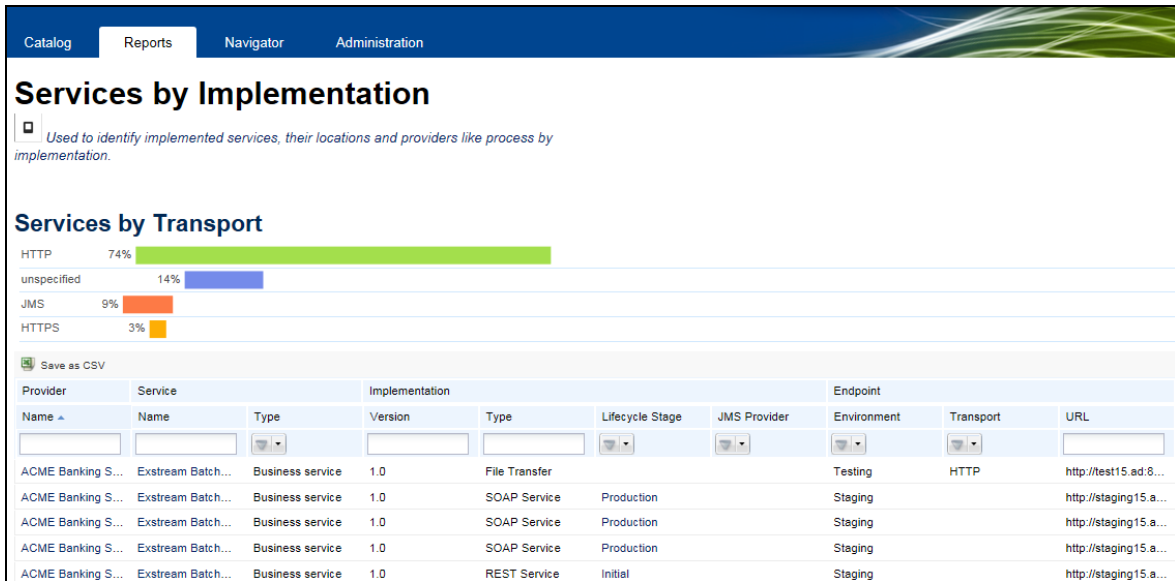
Services By SLO - Used to identify services by SLO, who provides them, and KPI data (e.g. review SLOs of services of a certain provider or providers of services offering gold SLOs).
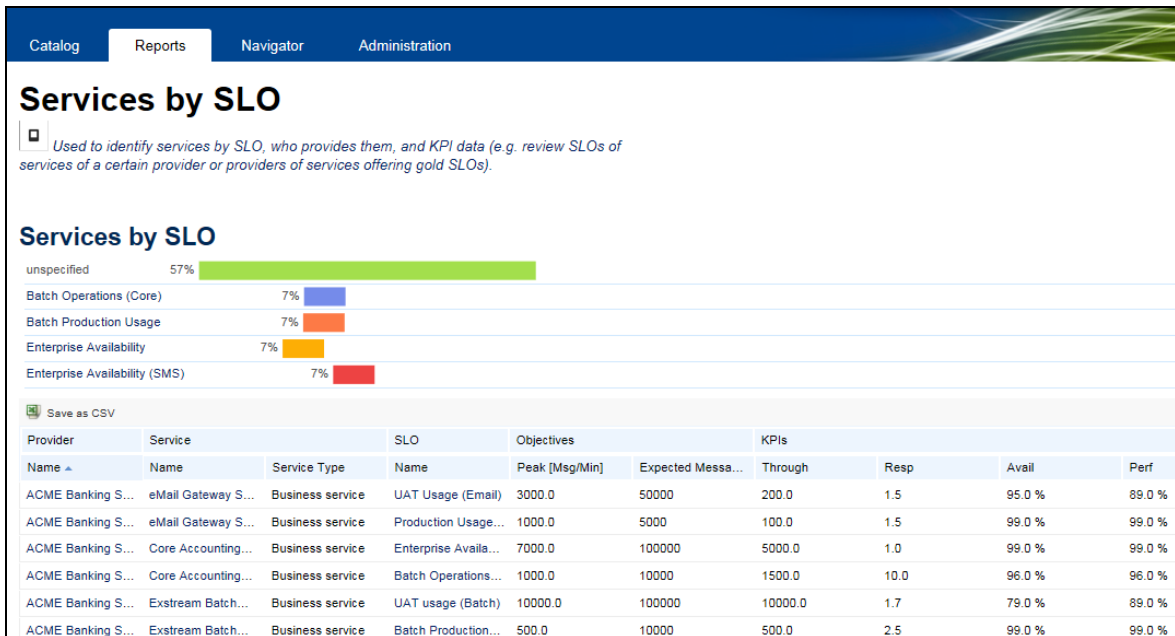
Undiscovered Services APIs - This artifact will list Application Interfaces (SOAP Services, REST Services, and so on) which are in Production Lifecycle stage and are not associated with a discovery source. For example, production services which are in production stage and are not associated with a discoverable source like UCMDB server or RGIF device.

Contracts By Provider - Used to identify contracts associated with a provider and the requestor of the contract (e.g. review providers who signed contracts for services in last 3 months).

Contracts by Consumer -Used to identify contracts associated with a consumer and provider of the contract (e.g. review consumers who requested contracts for services in last 3 months).
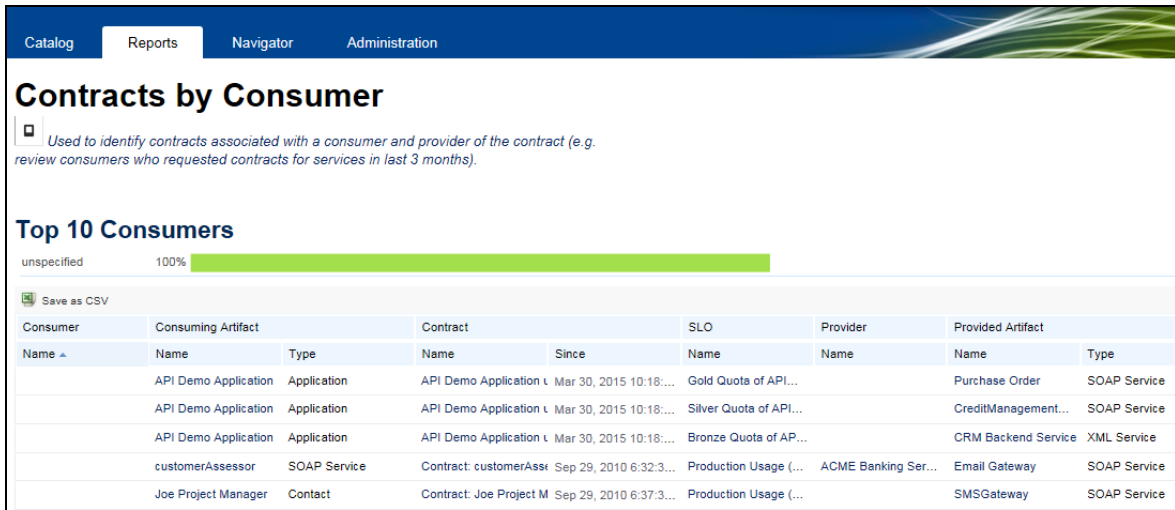
Contracts By SLO - Used to identify contracts associated with and SLO, who provides them and KPI data (e.g. review contracts with SLO guaranteeing more than 1000 msg/min).
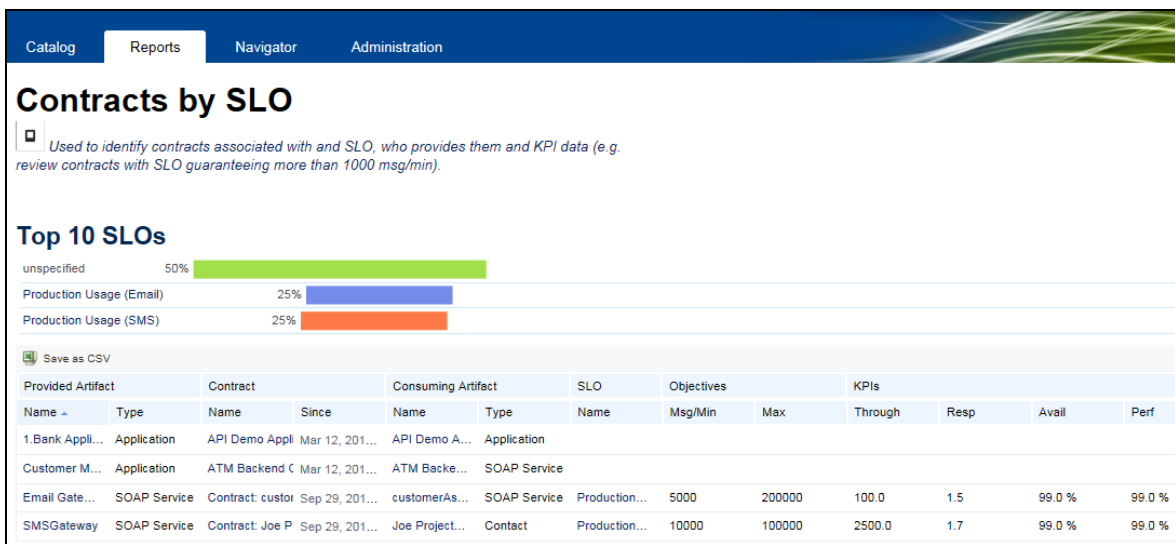
Pending Contracts - Used to review pending contracts and their providers and requestors (e.g. providers of services with pending requests for more than 6 months).

Lifecycle Processes - Used to identify artifacts as per what stage in the lifecycle process they are in.

Processes Summary screenshot showing Lifecycle Processes report.

Pending Approvals - Used to identify artifacts that are waiting for approval.

Policy Reports

Policy Reports display the compliance of artifacts selected by a repository search (or from external documents) against a set of technical policies. They can be used, for example, when Architects want to check the compliance of all WSDL documents containing a specific keyword against specified technical policies (e.g. WS-I Basic Profile Policy). Only policies defined in the current policy report are considered and eventual policies for the current lifecycle stage are ignored. Policy reports are also not affected by artifact compliance status calculations.

Policy Reports are recalculated automatically. By default they are refreshed daily, but Administrators can change the timing by scheduling an appropriate built-in task. The system can also recalculate reports based on a user's demand, but recalculation may be time consuming.

Systinet provides 2 new OOTB Policy Reports:

1. API Compliance Report

2. Application Compliance Report

API Compliance Report

This report is used to identify the application interfaces state. There are two ways to access the API Compliance Report page:

1. Go to **Catalog** homepage->**Manage APIs**->**API Compliance** or

2. Go to **Reports tab** homepage-> **Policy Reports** section in left menu > select the report API Compliance.

This displays the API Compliance homepage.

Application Compliance Report

This report is used to identify the application components state. There are two ways to access the Application Compliance Report page:

1. Go to **Catalog** homepage->**Applications**->**Application Compliance** or

2. Go to **Reports tab** homepage-> **Policy Reports** section in left menu > select the report Application Compliance.

This displays the Application Compliance homepage.

How to Create Policy Reports

1. From the **Reports** tab, click **Add Report** under **Policy Reports** from the left-hand menu.

2. Input a name for the new report and optional text describing the report.

   Under the **Repository Artifact** tab text may be added and used for search purposes.

3. Drop down boxes are provided for selecting *Artifact Type*, *Modification* dates, and *Domains*. Keywords can also be added and clicking on **View Top Keywords** will reveal a list of commonly used keywords that you may select from. See "How to Search the Catalog" on page 60 for further detail on this functionality.

   Clicking **Preview** will give you a view of what your report will look like and the column names in the preview can be used to sort the displayed data.

   The **External Document** tab allows the entry of a URL to document kept on another source along with the option to apply a name and password if the external source is protected.

4. Policies used during validation must also be added to the report by clicking the **'Add'** button and using the standard search functions to apply required policies.

5. Click **Save** to create the report.

How to Review Policy Reports

A policy report is created with three tabs. These tabs reveal an overview of the report, the report definition and report results.

Overview Tab - A graphical presentation of the report. Clicking on the **View results for all x artifacts** link will present the report results tab. The chart reflects overall compliance of artifacts while the table reflects compliance of each artifact against applied policies.

Report Definition Tab - Shows the Search Criteria and Technical Policies used for the report. Policies can be added or removed to modify the report.

Report Results Tab - Displays the results of the report in a table format that can be sorted or filtered.

Custom Reports

Custom Reports are based on report definitions as stored in the Reporting Server. These reports can combine multiple data sources. For example, it is possible to mix data from the repository with data from an external database.

While Custom Report definitions are stored in the Reporting Server and are managed with the Report Editor, artifact reports are customizable directly from the web UI in a user-friendly way.

Custom Reports are constructed in Report Editor and are available in the **Reports** tab under **Custom Reports** in the left-hand menu. For details, see *Report Editor Guide*Report Editor Guide.

Systinet provides 2 OOTB Custom Reports:

1. Top Reports

2. API Documentation

Top Reports

This report displays the most active providers, frequently consumed and often reused artifacts.

1. Select **Top Reports** from the **Report Definition** dropdown box.

2. Enter a name for the new report and an optional text describing the report.

3. Define a maximum number of rows for the report.

4. Click **Save** to generate the report.

API Documentation

This report provides information on the application interfaces artifact (for example, REST and SOAP services) using the API UUID.

1. Select **API Documentation** from the **Report Definition** dropdown box.

2. Enter a name for the new report and an optional text describing the report.

3. Enter the API UUID from the created artifact.

4. Click **Save** to generate the report.

Note: Top reports and API documentation reports can be viewed as PDF and HTML by clicking

the links on the right menu.

How to Add Custom Reports

1. From the **Reports** tab, click **Add Report** under **Custom Reports** from the left-hand menu.

2. Select a **Report Definition** from the list available in the dropdown box.

3. Input a name for the new report and optional text describing the report.

4. Click **Save** to generate the report.

Note: The above is a description of how to work with the default installation which provides a Top Reports report definition that displays the most active providers, most often consumed and most often reused artifacts. Depending on what customized reports may have been created, the creation process may differ.

How to View Custom Reports

Custom Reports are customized reports and the view depends on the configuration of the report. The standard options for viewers are:

Edit- allows user to change the name of the report, add a text definition or change the available parameters.

Delete - allows the user to delete the report

Recalculate- refreshes the report based on current data

View as PDF - creates a PDF file based on the currently generated report

View as HTML - creates a HTML file based on the currently generated report

These options can vary depending on user/role permissions.

How to Manage Content Feeds

Content feeds may be added to the bottom of the Reports homepage.

1. Click **Add Content Feed** to open the dialogue.

2. Enter a Title for your selected feed in the Title text box.

3. Enter the URL of the selected feed.

> **Note:** If the feed is from a secure site (including Systinet), a username and password may have to be entered.

4. Enter the number of items from the feed that will be displayed.

5. Enter the number of minutes that the feed will be updated.

6. Click **Save.**

Once the feed is visible on the homepage, it can be edited, manually refreshed or removed by clicking the appropriate links of the right hand side of the feed.

Send Documentation Feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on User Guide (Systinet 10.02)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to docteam_systinet@hp.com.

We appreciate your feedback!