# Cisco Process Orchestrator User Guide

Release 3.0
November 2013

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-30196-01

*Cisco Process Orchestrator User Guide*

# Preface

Cisco Process Orchestrator provides IT process automation capability within Cisco Intelligent Automation for Cloud, Cisco Intelligent Automation for SAP (resold by SAP under the name SAP IT Process Automation by Cisco), Cisco Network Operations Automation, and other custom solutions. Cisco Process Orchestrator's powerful process automation engine provides the logical constructs necessary to support even the most complex requirements to automate the administrative and operational tasks necessary to manage these systems.

## Audience

The Process Orchestrator's easy to use visual Process Editor allows processes to be rapidly designed and deployed—all with a minimum of training. Its powerful delegation model allows tasks that today consume the time of senior staff members to be redirected to other staff members or even other departments.

With Process Orchestrator, your IT organization can attain higher levels of operational excellence by increasing the deployment of operational and administrative best practices, improving the consistency with which process and policy is followed, and improves service by reducing operational and administrative errors.

## Related Documentation

For more information about the Cisco Process Orchestrator and related products, see the *Cisco Process Orchestrator Documentation Overview.*

## Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation* at: http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html.

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation, as an RSS feed and deliver content directly to your desktop using a reader application. The RSS feeds are a free service.


# Understanding Service-Oriented Orchestration and the Cisco Process Orchestrator

Cisco Process Orchestrator 3.0 introduces a new set of features providing Service-Oriented Orchestration that enable a paradigm shift vs. traditional run-book automation and IT process automation. This shift enables automation to align to the high-level services provided by IT, and models how a high level service is supported by a topology of lower level services, systems, and devices.

Planning for services and their desired state are the initial focus of automation design. Focus then moves to defining process actions against these services, with implementation of specific process workflows that traverse these services to act on lower level elements as a final implementation step.This enables a declarative approach to automation, focusing primarily on *what* is desired instead of *how* it is achieved.

**Note** The capabilities are additive and complimentary to traditional orchestration definitions and approaches, so Cisco Process Orchestrator provides both service-oriented and process-based orchestration. This means that you can still program Process Orchestrator as usual, so you can ease into the new concepts or not use them at all. This guide, however, will demonstrate the service-oriented orchestration approach.

This chapter contains an overview of Service-Oriented Orchestration and Process Orchestrator.



## Understanding Service-Oriented Orchestration

Cisco Process Orchestrator is an advanced orchestration engine belonging to the Run-Book Automation (RBA) or IT Process Automation (ITPA) class of products. Traditionally, tools in this category focus on a sequence of IT processes that achieve automation. The process is the focal point of automation. Processes act on lower-level IT elements such as devices, servers, or specific tools. The set of elements on which automation acts are typically delivered in the product and through adapters connecting to various layers of the IT technology stack. IT, on the other hand, is focused on services providing value to the business, which are much higher up the stack. The inability of RBA/ITPA tools to act on the business-level services in the environment becomes an inhibitor to delivery and creates a poor abstraction for users.

Service-Oriented Orchestration provides the agility to model and act on IT services. These features make creating orchestration active and dynamic, and allow for:

- Defining new, higher-level services in the system, and deploy new services quickly.
- In real-time, after these new types of services have been defined, creating real-time instances of those new services.
- Using events to watch for patterns in these services, enabling policy-driven automation.

Service-Oriented Orchestration combines several industry trends to synthesize a fresh approach to orchestration:

- Service modeling capabilities of a service catalog are now available in the orchestrator layer. Process Orchestrator provides fluid mechanisms to exchange service information with the Cisco Prime Service Catalog, advancing the integration of these systems.
- The feature delivers many of the capabilities of object-oriented design and programming into the RBA / ITPA world. The shift from traditional orchestration to Service-Oriented Orchestration is similar to the shift from procedural to object-oriented programming. Today, virtually all programming is done with object-oriented languages, and object-oriented design has transformed the industry to higher levels of productivity and quality. Service-Oriented Orchestration holds the same promise.
- The IT Infrastructure Library (ITIL) prescribes a service-centric approach for IT. Configuration Management Databases (CMDBs) model IT services and their relationship to other IT assets. Service-Oriented Automation allows automation to be driven through a model of current and potential IT services, aware of their relationships and interdependencies. In this aspect, the principles of service modeling in Process Orchestrator are essentially the same as modeling services within ITIL and CMDBs. While Process Orchestrator can integrate with an available CMDB, there is no requirement to have a CMDB to enable orchestration.
- The feature aligns to industry standards like the DMTF Common Information Model (CIM) and the Topology and Orchestration Specification for Cloud Applications (TOSCA).
- Model-based automation is becoming popular via script-based tools, especially in the configuration management space. Process Orchestrator combines the capability to model services with the openness to integrate with these tools to leverage their strengths. Moreover, the feature allows model-driven orchestration atop legacy tools to bring the full power of model-driven approaches to integrate with other IT tools.

# Key Benefits

Service-Oriented Orchestration allows automation to focus on higher level IT services, possibly specific to the customer's business and unknown by Cisco when the product is built. User interaction shifts to services on which to act and what you can do to them. The approach to defining automation shifts from having to first decompose automation into a sequence of *processes*, to decomposing high-level *services* into their components, and then defining actions that are possible on each of the component services.

This inversion in approach is simple, yet powerful. Cisco Services, partners, and customers can model services and extend others' services without coding. Extensions and automation can be packaged, shipped to customers or moved from development to test to production, versioned, and upgraded.

The approach delivers several key benefits:

- Greater ease of use in combining Process Orchestrator with Prime Service Catalog.

The Prime Service Catalog Adapter simplifies the exchange of service requests and service items to and from the orchestrator. It is easy to create targets from incoming service requests. Process Orchestrator helps push service items back to the catalog as needed by allowing users to pick the right moments in a flow to synchronize data. Using active catalog connection in the Prime Service Catalog adapters optimize user experience. The Prime Service Catalog adapter easily reads, writes, or creates service items, and is CP schema-aware. When a service item definition is created in the catalog, the desire is to set up integration rapidly. The property browser exposes the catalog definition to enhance ease of use. Get Service Item properties can be followed by Update Target to save properties in a single step. One can write directly through the Create Service Item or Update Service Item activities using property references.

- It allows simpler, more readable workflows.
- Data defines desired services, and the service instance drives automation to achieve the desired state. This separates the desired state from the implementation (the “what” from the “how”).
- It acts on the higher-level service rather than its technology elements. Services can span tools, and workflows can navigate service topologies to lower-level elements on which they act.
- It provides operational views of automation by service.
- It lets you monitor the environment vs. the service definition and bring them in line with policy.
- It provides federated storage, including push objects and relationships to and from Service Catalog, CMDB, and Service Assurance tools when needed.
- Automation becomes easier to extend and customize.

# Process Orchestrator Supports Service-Oriented Orchestration

Several features in Process Orchestrator combine to bring these capabilities:

- Service instances are **targets**. A **target type** allows you to define a new service; all new targets are created based on a target type. Target types can be based on the DMTF Common Information Model, but this is not required.

  Target types are service definitions that allow Process Orchestrator to deliver service-oriented automation, where the *service*, not the process, is the focal point. In service-oriented automation, the *content*, not the Process Orchestrator platform, defines the solution models; the platform is open to any model you want to produce.

  Some target types can be ‘abstract’, meaning they cannot be directly instantiated into targets but are only available for inheritance by other target types. In Process Orchestrator, these target types are marked as either ‘creatable’ or ‘not creatable’.

  Target types support inheritance, which allows you to extend a general type for a specialized need. In Process Orchestrator, you can select a ‘base target type’ for a target type that specifies that the target type inherits from the base target type.

  Target types have an extensible list of **properties** including field-customizable default values.

  In these properties, data can be stored that might come from Process Orchestrator in terms of defining an order or a desired state, or might include data that is collected from the environment. For example, periodic network device discovery can store information such as the operating system, version, which line cards are installed, and so on. Network automation can use that information in its workflows.

- **Relationships** allow modeling of topologies of lower level services assembled to offer higher-level services; that is, relationships allow lower-level objects to be bound together to achieve a larger whole. Automation can navigate from a higher-level object, such as a service, to lower-level objects against which they can perform automation.
  - Relationships allow you to navigate from one object to another in a workflow.
  - Target types define the property that models the relationship.
  - Target instances provide the link to a specific instance.
  - Relationships support inheritance.
  - Relationships can be set in the northbound web service like any property.
- **Processes** provide actions that can be executed against targets. Process Orchestrator allows you to view a target to see what actions are possible, or what automation is being done against those targets.

  The target types that a process can accept define the services on which a process can act.
- Targets and target types have **events**. These can be internal process events or the open Advanced Message Queuing Protocol (AMQP) protocol. Triggering processes in response to patterns in underlying processes provides policy.
- **Extensible by services, partners, customers**. The content, not the platform, defines service models. The platform is open to model any IT service topology within automation content. For example, some types, actions, properties, etc. may come from the packaged product, some from team 1, some from team 2, some from a partner, and some from a customer. Using this capability you can assemble a complex solution from parts. Each author controls the version control and lifecycle of the elements they deliver so they can deliver upgrades.
- A **consistent API**, even if elements come from different authors. Users and automation pack authors (see Automation Packs, page 1-15) can control which target types are published to the NorthBound Web Service (NBWS; see the *Northbound Web Services Guide*). For published objects, Process Orchestrator exposes the type uniquely in the NBWS. As content teams, services, partners, or customers extend types, these APIs are maintained to provide a consistent format for APIs across all types of objects. However, each type of object is uniquely supported.
- Authors can also configure which properties are exposed in the NBWS as optional parameters in the per-type WSDL.

# Aligned to IT Standards

While Service-Oriented Automation is open, allowing any model, it aligns to IT best practices such as the IT Infrastructure Library (ITIL). ITIL prescribes a configuration management database (CMDB) as a central tenet which allows high level services to be decomposed into their supporting lower level services and ultimately to the lower level systems and devices. In ITIL, each element in the model is called a Configuration Item since it is a unit of configuration. This provides a mapping of how lower level elements support business services. Service-Oriented Orchestration allows automation to be driven through a model of current and desired IT services, aware of their relationships and interdependencies. In this aspect, the principles of service modeling are substantially the same as modeling services within ITIL and CMDBs.

However, the implementation bypasses many of the negatives and weight of typical CMDB implementations. Process Orchestrator can integrate with a CMDB if it is there. Targets provide a natural synchronization point to exchange data with a CMDB, and target events allow that synchronization to be externalized from other processes. However, Service-Oriented Orchestration does not require a CMDB; one does not have to have a CMDB to enable orchestration to function. Often orchestration needs a reduced model vs. what is in the CMDB, and it is therefore much easier to keep the smaller data

set current. For example, resource management can be more efficient when externalized. Moreover, a CMDB typically represents actual elements and not potential elements, and since orchestration is often responsible for provisioning, the orchestrator may have data prior to its life in the CMDB, which it might write to the CMDB as it is instantiated. Even if Process Orchestrator service models leverage data in CMDBs, performance is improved by keeping this data in Process Orchestrator target properties to optimize queries. Process Orchestrator allows a local representation of only those services that relate to what you want to automate, so that the implementation is right-sized and lighter than complete CMDBs. Updates are more real time and targeted to the need of automation. A conceptual alignment to ITIL and CMDB principles creates the flexibility to right size the implementation and unify with CMDBs when needed.

Also, ITIL prescribes key process records such as alerts, incidents, and change requests that drive the IT operations process. Process Orchestrator provides a native representation for these concepts. This allows content to be abstracted from how these concepts are provided in a company. For example, content that detects an IT incident does not have to encode the specific service desk used by a particular customer.

Content that enriches incidents with further diagnostic data can also do so independent of tool selection. Process Orchestrator alerts, incidents, and change request records allow reusable content independent of tool selection and company-specific ITIL extensions. A separate body of content can integrate orchestrator incidents with a specific service desk like BMC Remedy. Processes can trigger when incidents are created or changed in general automation and push those changes to Remedy. Updates within Remedy such as incident closure trigger synchronization processes which update the Process Orchestrator incident. Processes in general automation can then proceed from incident resolution. For example, the incident can be verified to prove that the problematic behavior no longer manifests. Moreover, ITIL prescribes that these records have a relationship to the element which failed so one knows what services may be impacted. Process Orchestrator incidents provide a link to:

- The target defined by the target type for the service
- A secondary configuration item field to link arbitrary external CMDB references when a CMDB is present

# Related Service-Oriented Orchestration Terms and Concepts

The following table summarizes how Service-Oriented Automation blends aspects of these industry standards as well as common best practices like object-oriented programming. Users who are familiar with one of these domains might find it easiest to map the concepts with which they are familiar to Service-Oriented Orchestration features. As you can see, the feature synthesizes similar concepts that have very disparate terms across the industry. Process Orchestrator attempts to provide the most intuitive terms within orchestration, and especially with prior Process Orchestrator users.

*Table 1-1 Service-Oriented Orchestration Terms and Concepts Comparison*

| Object-Oriented Programming | IT Infrastructure Library (ITIL) | Cisco Process Orchestrator | Cisco Prime Service Catalog | Common Information Model (CIM) | Topology and Orchestration Specification for Cloud Applications (TOSCA) |
|---|---|---|---|---|---|
| Object Model | Service<br>Configuration Item Hierarchy | Collection of related targets | | Model | Service, Service Template, Topology Template |
| Class | Configuration Item type, or more loosely asset classes | Target Type | Service Item Definition, Service Standard Definition | Class | Node Type |
| Instance, object | Configuration Item, Service | Target | Service Item, Service Standard | Object derived from Managed Element | Node |
| Inheritance, Parent / child / ancestor / descendant Subclass / superclass, subtype / super-type | (loosely) Configuration Item categories and classification | Inheritance | Inheritance, Parent / child / ancestor / descendant | Inheritance (subclassing) | Inheritance, Parent / child / ancestor / descendant |
| Attribute, field | Configuration Item attribute | Property | Attribute | Property | Property |
| | Relationship type | Future: Relationship Type (achievable through Target Types - see above) | | | Relationship type |
| Links and associations Pointers | Configuration Item relationship | Relationship | Relationship | Relationship, association | Relationship |
| Method, operation | | Process<br>From Target Type: Process Action | Action, Associated Services, Tasks | Method | Plan |
| Event or Message | | Event<br>From Process: Trigger from Event | Event<br>Rules for “On Event”<br>Some use of the term “Trigger” | Indication (SNMP, Alert, Threshold, and others) | |
| | Selection by Configuration Item Category | Target Group, Target Selection | ... Rule | | |
| | Event (related to Configuration Item) | Alert (Related to Target) | | | |

*Table 1-1 Service-Oriented Orchestration Terms and Concepts Comparison (continued)*

| Object-Oriented Programming | IT Infrastructure Library (ITIL) | Cisco Process Orchestrator | Cisco Prime Service Catalog | Common Information Model (CIM) | Topology and Orchestration Specification for Cloud Applications (TOSCA) |
|---|---|---|---|---|---|
| | Incident (related to configuration item) | Incident (Related to Target) | | | |
| Change Request (related to configuration item) | | Change Request (Related to Target) | | | |
| | | Automation Pack | | | Cloud Service Archive (CSAR) |
| | | Files in an Automation Pack | | | Artifact |
| Class Interface | | No explicit term The collection of all processes that act on a Target Type, including their input and output parameters | | Class Interface | Interface |
| “This” reference in a method | | Process Target | | | |
| Abstract class | | “Targets of this type can be created” checkbox on a Target Type | | | Abstract type |
| Dynamic binding, overriding, polymorphism | | Achievable in processes | | | |

# Service Definition Examples

The following sections provide some service definition examples.

## A Distributed Application and Product as a Service

An example of the usefulness of Service-Oriented Orchestration is the ability to model a distributed application and provide its provisioning and ongoing operations. To the business, an application and its elements are not important by themselves; the application is important for the service it provides to the business. For example, an application may provide customer support, order entry, or partner billing. Let's pick one of these, say customer support. The customer support service in this example is provided by a distributed application that consists of a number of web servers behind a load balancer, a database, a cluster of application servers, and an LDAP repository for user credentials. Each application server ultimately runs on a UCS blade, which is in turn related to a UCS Manager. Moreover, each of these

topology nodes fit within some network, with each tier separated by firewalls, and these nodes require specific firewall configuration, so the application topology has relationships to a network topology. All of this combines to form the service topology.

In a Platform as a Service solution, the customer must enable not only provisioning but ongoing operations and eventually deprovisioning of services or applications, as opposed to IaaS where individual virtual machines are provisioned. The leap involves treating a service or a collection of virtual machines and their networks as a package. Applications or other services are provisioned and configured on these VMs and networks. It involves an ordering experience that allows customers to deploy and use those services.

To this end, there is a need to define service or application blueprints so they can model their services and make them orderable and support provisioning, operations, and de-provisioning of those services. These blueprints capture what the customer needs to deploy some application. Customers want to receive these blueprints from Cisco, or possibly from partners or a community. They also need to be able to build these blueprints themselves and move them from development to test to production.

Service-Oriented Orchestration allows one to define, package, ship, and receive these models, as well as to create instances of the service from that model. In this way, the company can drive a standardized deployment of the service across development, test, and production systems. For example, a target type could be created for a web server, while the specific web server target is an instance created from that web server target type. This might have a relationship to the Linux server that hosts the web server.

A level 1 help desk operator might not know or care what specific database or LDAP instance supports the customer support service, or what UCS blade actually runs a certain instance of the web server. They want to do things against the customer support service. Say they want to authorize a new user to access the customer support service and provision whatever they need. They would generally run an add-user process, acting on the customer support service.

A Build process that acts on the customer support service is responsible for provisioning of the whole of the service. This process might call the Build process for each target type in the topology, building the service up an element at a time. This process might use files, like an OVF for a server, to stand up the specific service element, or might invoke a tool such as Puppet or Chef.

Other examples of processes that act on the service or its elements are things like site disaster recovery, that may need to make changes to many parts of the customer support distributed application to move it to another site. These processes need to act on the customer support service as a whole, but leverage the topology and relationships to traverse to other actions, possibly calling similar processes that act on those elements. For example, a request to back up the overall service might invoke a backup of each server in the topology as well as the database.

Through Service-Oriented Orchestration, processes can act on the customer support service. As necessary, the process can traverse relationships to lower level infrastructure and tool elements on which it takes action. For example, a process that queries capacity might traverse to the database target, and perform Process Orchestrator activities against that database.

# A Cisco TelePresence Service

A TelePresence system has a number of components that automation can act on through SSH, SNMP, or a web service. Prior to Process Orchestrator 3.0, to act on a service, one was required to browse through all of the processes, filter by category or automation pack (see Automation Packs, page 1-15), then run a process and specify the terminal, SNMP, or web service target. One had to understand the underlying implementation to know what target to provide.

Now, using Service-Oriented Orchestration:

1. An external team provides an automation pack that defines a TelePresence target type, with:
   - Relationships defined for terminal, SNMP, and web service target types
   - Properties such as a phone number or escort name
   - Process actions for TelePresence systems
2. The automation pack can add properties such as a location property to the built-in network device target type.
3. The customer installs the automation pack and configures TelePresence service instances by calling a constructor process called **Create**.
4. The process not only creates the TelePresence target, but also creates the terminal, SNMP, and web service targets as well as the relationships that unify them into the model.
5. The end user can browse the target views or operations views and filter for targets with the TelePresence type. When they select a type, they can see:
   - All of the available user-startable actions (processes).
   - All automation running against the TelePresence system. These results are filterable by a time range or by a specific process.
6. When the end user runs a process action, internally the workflow traverses the relationship to find the SSH, etc. target required by the action. The user sees this and other automation running against the TelePresence system.

# Process Orchestrator System Components

The topics in the following sections describe the major architectural elements of the Process Orchestrator.

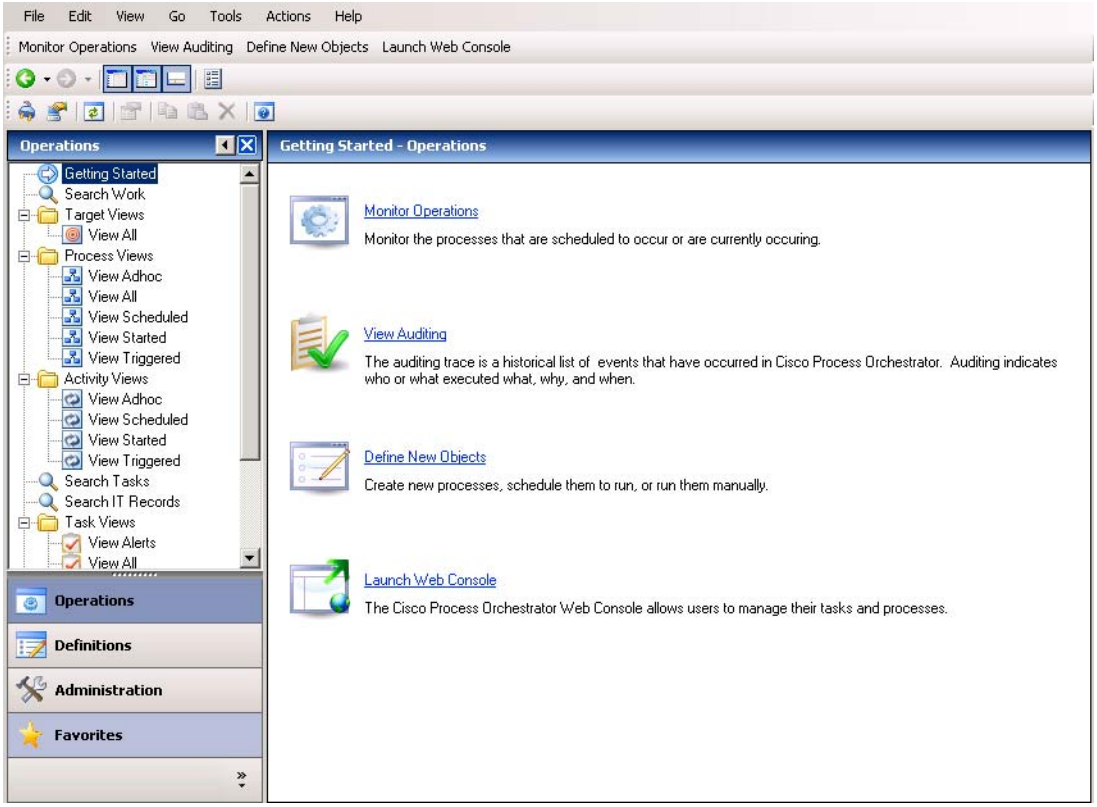
## Process Orchestrator Console

The Process Orchestrator console is a Windows form-based UI that is intended for advanced users that will be defining processes, target types, activities, and so on.

The console contains the following workspaces, each of which contains a group of objects that perform specific actions within the application.

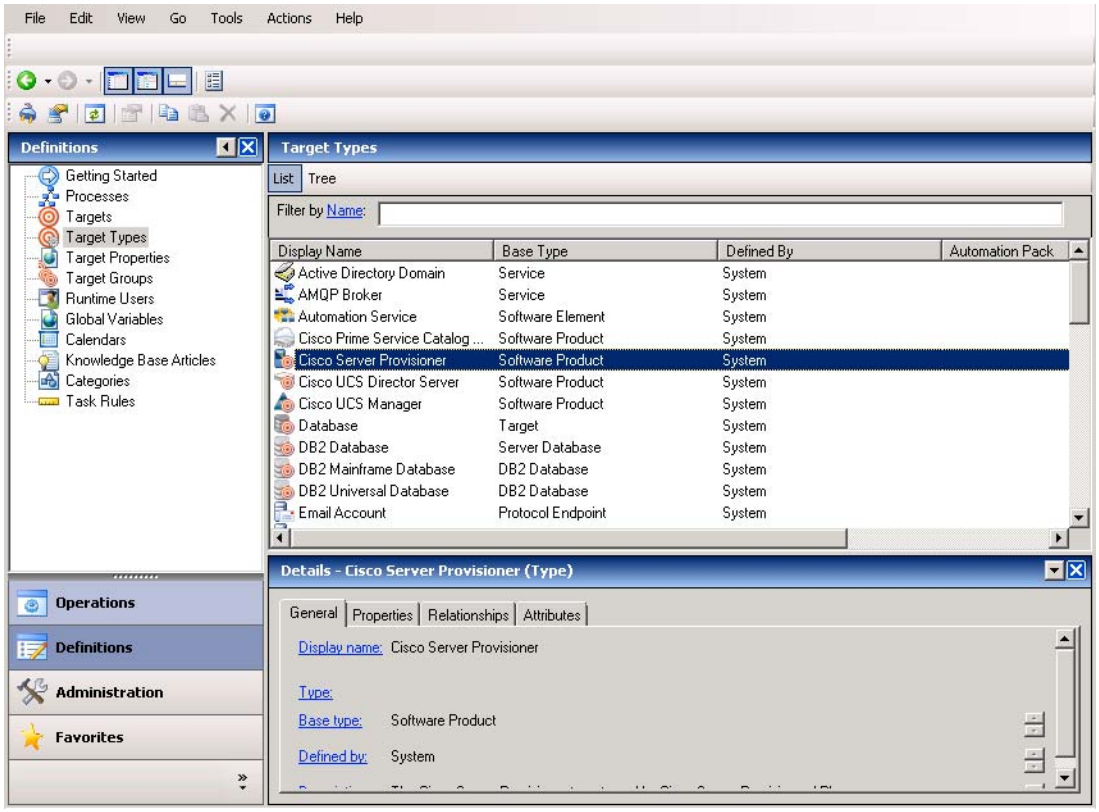
- Operations—Use the Operations workspace to monitor the orchestration that is executing or is scheduled to execute on various targets. You can also use this workspace to monitor the processes that are scheduled to execute, view processes that are currently running, and verify that processes have successfully completed.

***Figure 1-1 Operations Workspace***



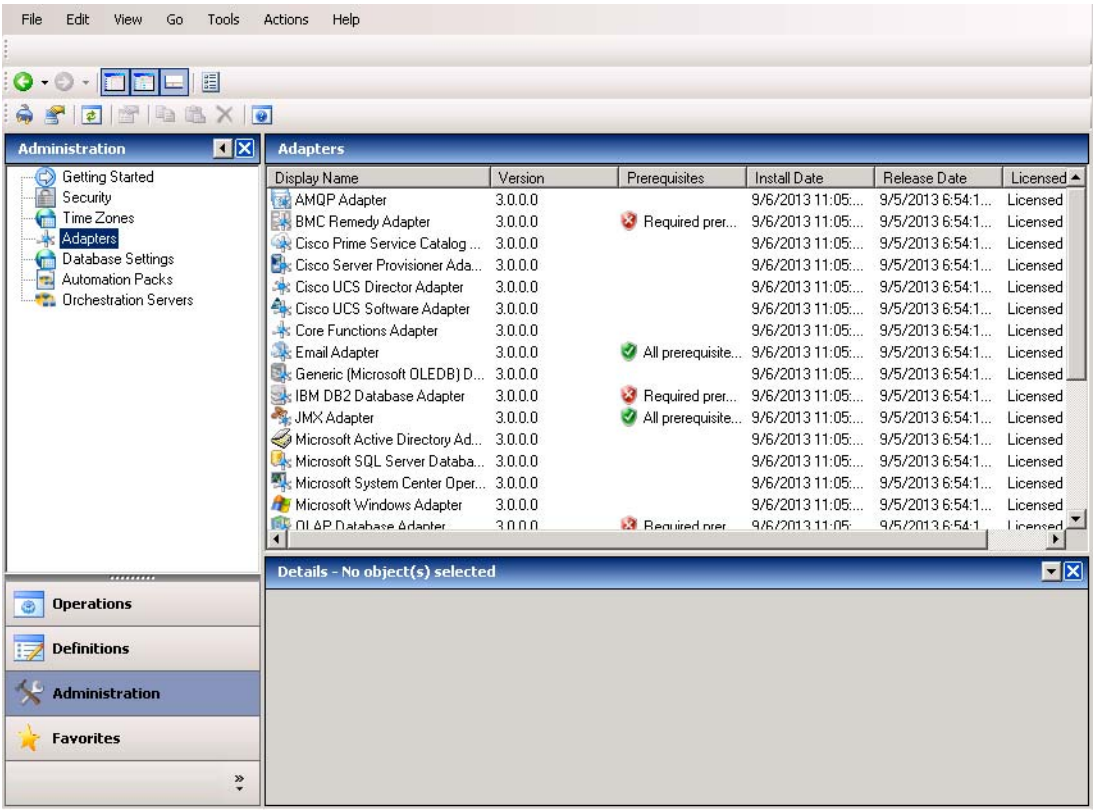
- Definitions—Use the Definitions workspace to view and modify processes and other related configurations that are used in executing automation.

***Figure 1-2 Definitions Workspace***



- Administration—Use the Administration workspace to perform administrative actions such as editing system-wide settings, managing security roles, configuring adapters and automation packs, and tuning database grooming and other settings.

***Figure 1-3 Administration Workspace***




# Web Console

The Process Orchestrator Web Console provides a web-based UI that is intended for occasional users. The vision is that while a broad set of users may be defining processes, pervasively everyone in IT and even non-IT individuals may need to interact with those processes. The web console focuses on this second group. It does not allow users to define processes but does allow them to start processes, monitor processes they started, or interact with human steps in processes called *tasks* (such as approvals). This interface is purposefully simplified to minimize the learning curve of occasional users.

**Note** The Web Console is typically not used when Process Orchestrator is used in conjunction with Cisco Prime Service Catalog. In these cases, end-user interaction occurs through the catalog rather than Process Orchestrator interfaces. The Web Console is typically only used in cases when Process Orchestrator is used independently.

The web console runs on a Microsoft IIS web server. Process Orchestrator environments can span Process Orchestrator servers, so if a deployment scenario requires breaking up automation for security, organizational, geographic, or purpose (such as development or test), the Web Console can provide a single point of access for occasional users.

**Related Topics**

Chapter 13, "Using the Web Console"

# Northbound Web Service Interface

Process Orchestrator provides a web service interface that allows external systems to invoke Process Orchestrator processes, check their results, and so on. Use the northbound web service interface when you:

- Have a need to integrate with or communicate between multiple Process Orchestrator environments
- Have your own support portal
- Use a service catalog
- Leverage Process Orchestrator underneath some other master orchestrator

For more information about the northbound web service interface, see the *Cisco Process Orchestrator Northbound Web Services Guide*.

# Command Line Interface

The Process Orchestrator command line interface is based on Windows PowerShell. PowerShell is the emerging standard for CLI interfaces on the Microsoft platform. The CLI provides basic operations that are needed for integration, including the ability to:

- Export or import automation packs
- Start a process with input variables
- Check the status of a process
- Enable and disable processes and targets

# Reporting

Process Orchestrator supports two reporting technologies:

- SAP BusinessObjects
- SQL Server Reporting Services

Report definitions are provided in the Process Orchestrator product and in some Cisco automation packs, and are imported into the chosen reporting platform.

**Related Topics**

Managing the Report Database, page 4-6

# Process Orchestrator Server (Process Engine)

At the core is the Process Orchestrator Server, which executes automated processes. This component is responsible for:

- Creating instances of running processes
- Orchestrating the execution of these processes
- Configuring multiple servers to work together for load balancing and in an active-active configuration to achieve high availability.

# Role-Based Access Control (RBAC)

Windows performs authentication of the user connecting to Process Orchestrator against Active Directory. The authentication is done prior to connection to the Process Orchestrator server, and the Process Orchestrator server simply relies on the results of that authentication. No special end-user account management is necessary for Process Orchestrator.

In Process Orchestrator, authorization is performed using a Role-Based Access Control System. Roles are a collection of permissions. Users are assigned to roles that give them the ability to perform the actions allowed by the role.

Typically, roles are defined according to a standardized job function within IT. Examples might include Level 1 Helpdesk, Level 2 Helpdesk, Cloud Technical Administrator, Network Configuration, SAP Basis Expert, and so on. Security groups already in the directory for the users in these job functions are then typically assigned to the roles.

# Process Database (Configuration & Audit)

This database is closely associated with a Process Orchestrator high availability environment (that includes one or more servers) and stores process definitions and other configuration information. In this environment, the Process Orchestrator Server (Process Engine):

1. Loads process definitions and other configuration information at startup. Then, as appropriate, it starts instances of processes.
2. Stores the state of running process instances into the Process Database. This persistence allows resiliency across server restarts and supports load balancing and high availability in an multi-server environment.
3. Stores auditing data in the Process Database to record change as users interact with the system, (such as to make a configuration change or invoke a process manually).

# Reporting Database (Data Warehouse)

This second database stores data needed for longer-term retention and storage. This can include a copy of auditing data that can groom more slowly than the Process database, as well as summary information for each process that ran. This data is organized for reporting, with flat relational tables, and can be summarized. Automation content can also add data, such as performance measurements, to the reporting database.

**Related Topics**

Reporting, page 1-13

# Adapters

Adapters are one of the extensibility mechanisms in the Process Orchestrator platform, and are only written by members of the Process Orchestrator product development team. The development team uses adapters to extend Process Orchestrator functionality to integrate with devices, environments, applications, or tools without undergoing core modification.

**Related Topics**

Configuring Adapters, page 4-13

# Automation Packs

Automation packs are the primary means of extending integrations in the field. Automation packs are prepackaged collections of process definitions, target types, variables, categories, targets, target groups, and other configurations needed to define a set of automated IT processes. Automation packs allow:

- Cisco or its partners to ship best practice automation to customers.
- Customers to get productive quickly, and to shift automation across systems, such as from a development system to a test system to production.
- Customers to export their own processes and share with each other in the community.
- The backup of process definitions and other product configuration.

Automation packs:

- Can be exported from one Process Orchestrator system and imported into another.
- Can ship separately from product releases.
- Are versioned and dependencies are recorded. Versions and dependencies are checked during import to ensure that the system will work after the import. For example, if your personal automation pack calls a process in “Sample Document A” automation pack v1.3, then you cannot import your automation pack into a system that has installed “Sample Document A” v1.2.

**Related Topics**

- Managing Automation Packs, page 2-2
- Chapter 11, “Authoring Custom Automation Packs”

# Advanced Message Queuing Protocol (AMQP)

The Advanced Message Queuing Protocol (AMQP) is an open standard application layer protocol for message-oriented middleware. It delivers message orientation, queuing, routing (including point-to-point and publish-and-subscribe), reliability and security. AMQP came out of the financial industry, and is proven highly scalable in demanding environments (see http://www.amqp.org/).

AMQP is the emerging standard for messaging and events in the cloud. For example:

- vCloud Director can publish vCloud Messages, also known as “blocking tasks,” “notifications,” or “call-outs,” related to different provisioning. An orchestrator can not only receive these events, but can also respond to them to delay execution.
- AMQP is supported with vCloud Orchestrator.
- OpenStack selected AMQP as the messaging technology for OpenStack.

AMQP is integrated with more than 70 developer platforms, providing a nice framework for event-oriented integrations.

AMQP enables event-driven capabilities of Enterprise Service Bus-style designs. It enables queuing, so automation can fetch messages when there is capacity. You can use asynchronous methods servicing requests as capacity allows when possible, and reserve real-time, synchronous methods only when absolutely needed. AMQP’s open platform is a natural choice for event and message design patterns

Process Orchestrator can trigger processes in response to messages placed on AMQP queues/exchanges. Processes can also read messages from queues/exchanges one at a time if they want to respond to messages one at a time rather than in parallel, to operate more as a queue. Processes can submit messages to queues/exchanges.

JMS is another possible integration enabled through AMQP. AMQP has providers to place JMS messages on queues/exchanges.

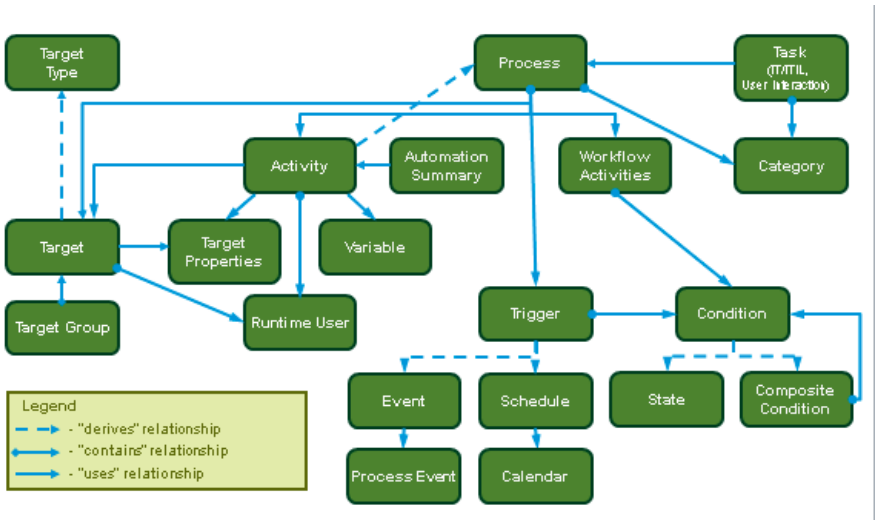
### Related Topics

Integrating Using the Advanced Message Queuing Protocol (AMQP), page 8-6

# Process Orchestrator System Elements

The following diagram shows the major functional elements of the Process Orchestrator system. These elements are discussed in the sections that follow.

***Figure 1-4 Process Orchestrator Functional Model***



# Target Types

Target types provide a way to define a service or other IT element that is not represented by any target type provided by an adapter. A target type can:

- Extend an existing adapter-provided target type
- Extend another target type
- Define a completely new target type

All new targets are created based upon a target type. Some target types are 'abstract', meaning they cannot be directly instantiated into targets but are only available for inheritance by other target types. In Process Orchestrator, these target types are marked as either 'creatable' or 'not creatable'.

A target type:

- Exposes (and inherits from its ancestor target types) properties and inter-target named relationships that can be read and set either manually or by an Update Target activity.
- Defines property default values.

### Related Topics

Chapter 6, "Authoring Target Types"

# Targets

Targets are *instances* created from a target type provided. For example:

- A terminal target allows SSH or telnet to some specific network device.
- A UCS Manager target allows connection to a specific UCS Manager in charge of a UCS system.
- A database target allows connection to specific supported databases.

A process or activity executes an action within some environment. The specifics of the definition of the connection to the environment are encompassed in the target. For example, a target for:

- An SSH command might be a specific UNIX system or network device.
- A database query might be a specific database
- An SAP ABAP activity might be a specific SAP system.

Processes can restrict the type of target which they can accept. For example, it is not appropriate to run a process to change a network device's configuration against a Windows computer.

A process workflow acts on a target. This allows the process workflow to perform actions against an external tool or environment. Although a default target instance can be specified directly by a process, more commonly it is associated to the process at run time.

Activities, which are the steps in a process (see Activities, page 1-20), can also specify a target. Typically, activities default to use the process target, but a step in a workflow might need to happen against a different system than the process target. For example, a process overall might deal with a network device, but a step in the process might need to update a database. Often workflows can determine the target on which an activity may act by using a relationship to the target for the process, or doing a query on some data such as a name to find a matching target instance.

### Related Topics

- Creating Targets, page 2-4

- Creating Triggers, page 8-1

# Target Groups

Target groups are collections of targets. Often automation might need to run against all machines in a collection, or against one of the machines in a collection. Target groups provide this functionality.

Adapters can provide target groups to leverage grouping definitions where they exist. For example:

- Active Directory OU—Customers are frustrated when they must recode their grouping information into yet another product. The Active Directory adapter seeks grouping information where it is defined. For example, an Active Directory target group looks up computers in some organizational unit (OU) in the directory.
- Target Type group—Using queries of their attributes, the Process Orchestrator provides type-based groupings of targets. For example, use a target group to group all targets of a given type, or to perform additional filtering to pattern match against a field in the target definition.
- Virtual group—A target group can be a virtual group. Use a virtual group to specify an explicit list of targets, providing the capability to manually select targets and establish group membership. A virtual group can also allow the inclusion of other target groups so that group membership can be defined hierarchically. For example, a virtual group called “Production switches” might include all members of the “Houston data center switches” group as well as all members of the “London data center switches” group, but not members of the “Engineering lab switches.”

A default target for a process can specify a target group along with a target selection algorithm:

- Typically, a target selection algorithm chooses one or more members of a target group to specify the target instances on which the process will act.
- At runtime the target selection algorithm resolves the then-present members of the target group and selects a target.
- When a user, CLI, or northbound API call interactively runs a process ad-hoc (on demand), the user can accept the default target specified in the process or override it with a specific target.
- Where a target selection algorithm resolves to multiple target instances, the engine spins up separate process instances for each target. Thus at runtime, a process instance has a specific target on which it will act, against which the process workflow encodes actions.

### Related Topics

Adding Targets and Target Groups, page 7-19

# Runtime Users

Many operating system and application actions require credentials. To run a SQL query against a database, for example, requires database credentials with read access to the relevant database tables. These credentials are defined in runtime users.

Targets have default runtime users. Often activities will leverage this feature to avoid specifying credentials activity by activity (see Activities, page 1-20). Cisco’s automation packs use this concept to reduce customer configuration and ship processes that customers do not need to alter.

Runtime user instances can be shared across targets and processes. For example, if a single set of credentials can be used to access a set of network devices, only one runtime user instance must be created. When it is time to change the credentials, users can go to the runtime users list and edit the single instance to change the credentials. This greatly reduces the configuration load when credentials tend to change often in some environments.

**Note** Runtime user credentials can be used in a process, but no process can retrieve credentials. If your process must access credentials, use hidden string variables (see Using Hidden String Variables, page 7-19).

The runtime user concept allows the product to implement delegation. For example:

1. An IT help desk operator comes to Process Orchestrator to run a process.
2. This operator is presented with a list of processes that Process Orchestrator's role-based access control allows them to run. These processes might include activities that require a level of security permission that the operator does not natively have.
3. The operator can perform actions as a part of the established process that are not possible for them to perform manually.

This concept can also be leveraged to reveal where operators make changes outside of a process. By examining auditing logs such as Windows logs for things being done under the operator's credentials rather than the Process Orchestrator runtime user credentials, it is possible to determine how the operator is doing things outside of process and determine how to close things down. So a side effect of Process Orchestrator automation is that customers might be able to tighten security in their environment.

#### Related Topics


- Defining Runtime Users, page 2-6
- Role-Based Access Control (RBAC), page 4-17
- Using Hidden String Variables, page 7-19


## Processes

Use Process Orchestrator to automate an IT processes by defining a process, then running instances of the defined process.

An element of the process is the process workflow. The workflow defines the automation steps (activities), the logic or flow between these steps, and how to flow data from one step to the next.

The engine manages the state and lifecycle of a process, bringing it into existence, running its steps, and finally terminating it. During and (by default) after process execution, the engine retains information so that operators can view the status of their running processes.

#### Related Topics

Creating a Basic Process, page 7-5

## Workflow Activities

Workflow activities provide the logic or flow aspects of the process workflow. Workflow activities are exposed in the Logic tab of the toolbox in the Process Editor.

**Related Topics**

Adding Logic Components to a Process, page 7-12

# Activities

Activities are the steps in a process. They are customized to perform integration with some environment. For example, an SNMP trap send activity would have a very different UI and set of properties from an SAP ABAP call, and these would have very different runtime characteristics as well.

**Related Topics**

Chapter 12, "Advanced Authoring Concepts."

# Conditions

Many workflow logic elements perform tests to control execution. Conditions implement these tests. For example, a Condition Branch can split execution to take one path if a condition exists, and another if it does not. A While Block can iterate execution while a condition exists.

Conditions can also be placed on a process trigger, allowing control of situations in which the process can run. For example, a scenario might require the process to run during two different time ranges, which would require two triggers: one trigger with an 8am-5pm condition and a totally different trigger with a 5pm-8am condition.

There are two basic types of conditions:

- State conditions

  State conditions evaluate some state. The Process Orchestrator provides a number of state conditions that can be perceived as a part of the base product, but technically are part of the adapter. For example, it provides a test to see if a variable has some value.

- Composite conditions

  A composite condition builds a compound condition from individual conditions. It allows combining conditions with AND logic, where all of the conditions must be TRUE for the composite condition to be TRUE, or OR logic, where any of the conditions can be TRUE for the composite condition to return TRUE.

  Compound conditions can be used in other compound conditions to produce complex logic, such as ((X AND Y) OR (A AND B)).

Process Orchestrator supports these types of conditions:

| Condition Type | Purpose |
|---|---|
| Variable Condition | Specifies a variable to be used as the condition under which the variable should evaluate as *true*. |
| Time Condition | Specifies conditions that evaluate to *true* on the days in the specified calendar and between the specified start and end time. |

| Condition Type | Purpose |
|---|---|
| Prior Process Instance | Determines if a process has occurred within a specific time interval, the condition will evaluate to *false*. If no process instance is selected, then the trigger will search for all process instances. |
| Compound Condition | Combines other conditions (time condition, prior process instance condition, variable condition, or another compound condition) into a single condition. After it has been created, a compound condition can be nested within another compound condition.<br>For more information about creating compound conditions, see Adding Conditions, page 7-22. |

# Triggers

Process instances can come into existence in the following ways:

- A process can be invoked manually.
- A process can be invoked by another process.
- A trigger can fire, which initiates the process.
- A process can be invoked using the northbound web service.

Triggers are events and conditions in the system that can fire off processes. The attributes of the trigger that fires off a process instance can be referenced in the process workflow. Processes often use this data to control execution.

Process Orchestrator supports two types of rule-based triggers: events and schedules.

## Events

The Process Orchestrator can monitor for events from the environment, and you can specify triggers that initiate processes when the subscribed event occurs. For example, an event might be an incoming SNMP trap or a fault on a UCS system.

### Related Topics

Chapter 8, "Working with Events and Triggers"

## Schedules

Schedules allow triggering processes at some time by leveraging another object called a calendar. Calendars define which days something can occur. Calendars can be selected days or sequences of dates such as weekly or monthly, they can represent dates like fiscal quarter end, or they can be combined hierarchically. Schedules then associate a time with a calendar. When the day is in the calendar, the time is evaluated. Times can be explicit or repeating (for example, hourly).

### Related Topics

Chapter 9, "Scheduling Processes"

# Tasks

Tasks allow a process to create an IT record or perform human interaction, and provide ownership and lifecycle. Process Orchestrator provides a list of tasks in the operator console. Managers can then set or change assignments to those on their staff. Assignment changes, state changes, and all other changes are audited, so one can definitively tell exactly who did what in the system.

There are two types of tasks: IT process records and human interaction.

### Related Topics

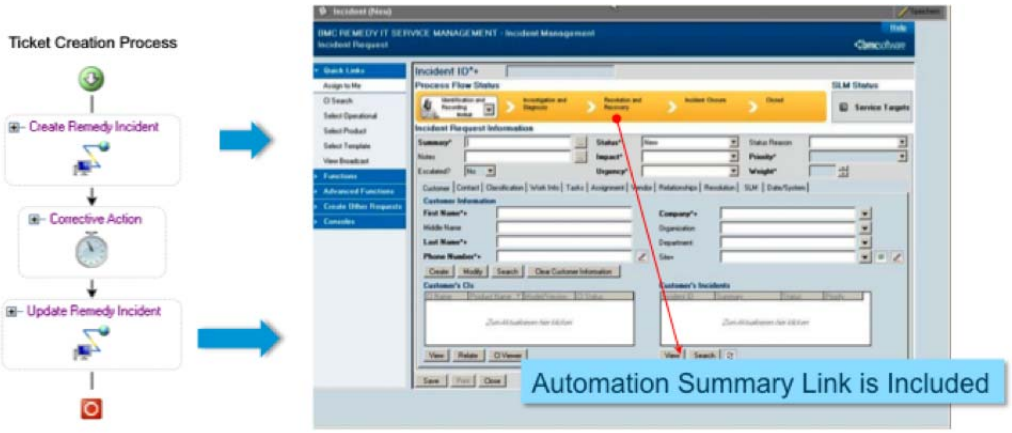
Authoring Task Activities, page 7-7

## IT Process Records

IT process records tasks include the alerts, incidents, and change requests. These records represent the core records within IT; they are the core ITIL (IT Infrastructure Library) records. While Process Orchestrator does provide a list of these records so that one can, for example, look at a list of alerts which automation created and manage their lifecycle, more often these records are used to synchronize the records out of Process Orchestrator to a separate tool.

For example, alerts typically need to be pushed to an enterprise event manager so that the alerts can be combined with others to present a view of IT system health. Incidents and change requests are typically the domain of service desks. Rather than managing these records in Process Orchestrator, many customers prefer to manage these records elsewhere.

***Figure 1-5 Synchronizing Alerts, Incidents, and Change Requests to IT Tools***




This construct allows processes to be independent of the event manager or service desk. For example:

1. A process such as one in a network best practice automation pack simply creates an alert, incident, or change request.
2. Optionally, this process can block waiting on the task to enter a completed state.
3. Separately, a Remedy automation pack includes a process that triggers when a new incident task is created, and creates the corresponding entry in the Remedy system.
4. The Remedy automation pack can synchronize data between the systems and when the incident reaches a terminating state in the Remedy system, the incident in Process Orchestrator is then automatically synchronized.

5. When the incident in Process Orchestrator is closed, if the process in the network best practice automation pack blocked, it can continue. For example, the process might go on to prove that the problem that generated the incident in fact no longer manifests, closing the loop to prove that the remediation did in fact fix the problem.

With this abstraction, the network best practice automation pack does not need to be dependent on the specifics of which service desk is involved, and no dependency exists between the automation packs.

### Human Interactions

Human interaction tasks are steps in a workflow where a human needs to take action. They include Approval, Input Request, Guided Operation, and Review tasks.

- Approval tasks consist of steps where the workflow needs someone's indication of what route to take. They may be simple yes or no questions, or they might give any arbitrary set of choices.
- Guided Operation tasks consist of steps in a workflow that someone must perform manually. For example, a step in a network automation workflow might be to replace a cable. While this step cannot be automated, it might be a small part of an overall workflow that can be automated.
- Review tasks allow the assignment of a document or automation summary review. When someone indicates the review is complete, the workflow can continue. Moreover, the person who indicated they completed the review is recorded in auditing. Many systems might send a notification to let someone know that they need to, for example, look at a report, but do not close the loop to see that the review actually occurred. Process Orchestrator allows closing the loop.
- Input Request tasks allow gathering data as the workflow executes.

## Categories

Process Orchestrator processes, tasks and several other elements of the functional model can be placed into categories. Categories in Process Orchestrator work just like Microsoft Outlook categories with respect to tagging objects for grouping in the UI. Objects such as processes can be in multiple categories. For example, a process can be both a network best practice and a security best practice.

Categories are fundamentally used to make visibility easier. For example, in process definitions views or views of running processes users can filter the view by a category.

Categories can reference other categories in a hierarchical fashion, in that they can include objects in subcategories.

#### Related Topics

Chapter 10, "Authoring Categories"

## Task Rules

Task rules provide a list of rules that act on new tasks or task changes to handle cases such as assignments, changing alert or incident severities, or setting customer-specific categories on tasks. They provide an easy-to use, simple, yet powerful mechanism for routine activities customers need to perform for alerts, incidents, and other tasks. This feature allows the management of tasks created within Cisco automation packs without requiring that customers understand or change the automation pack itself. By performing these actions by rule, one change can affect tasks created by multiple processes. For example, it is possible to assign all SAP Incidents to a single person with a single rule.

Some use case details include:

- Task Assignment—A new user installs and runs the SAP Automation Packs. Automation Pack processes create Alerts and Incidents. The user needs to assign the alerts and incidents to individuals who own that specific area of SAP. It is difficult to anticipate every way the customer may have broken up their SAP management responsibilities within the team. Virtually every customer who uses Process Orchestrator for SAP customizes the product for assignments. The user can go to a list of Task Rules and create a rule to make the assignment. In this list, they can see what other assignments have been made.
- Task Notification—Beyond notifications, when an owner is assigned, often the same or separate people must be notified in response to incidents. For example, if a job fails that is a financial job, perhaps both the technical owner of SAP job management and someone in the financial team should be notified; ownership of the incident should not be assigned just to the financial team. Note that Cisco's implementation task rules set a "people to notify" property on the task. The Task Rule identifies who to notify and a process would send the actual email.
- Change Alert or Incident Severity—Cisco's SAP automation pack assigns the severity of each detected condition when it creates the alerts and incidents. However, customer environments and requirements are unpredictable. It is very common for customers to change the severity of their alerts and incidents to match the business importance of that issue relative to others in their systems.
- Add a Custom Category to a Task—If a user needs to see a certain type of review, alert, or incident within a restricted view, categories are a natural way to group these alerts. Default alert categories are established in the automation pack, but users might want to add their customized categories to Tasks generated by automation packs.

# Variables

Process Orchestrator allows users to define variables. There are several types.

- Global variables span processes. All processes can reference or update them. Process Orchestrator often stores data such as performance thresholds in global variables so that customers can edit them easily without modifying process definitions, and automation packs can then update processes without affecting customer settings.
- Process variables are created from within a process and can be used as a reference value to store or pass a value between executions of a process or between steps within a single process. Process variables can also be used to collect input parameters from the user or parent process. These variables are only available from within the defined process and cannot be accessed or referenced by objects outside of the process.

There are several types of process variables:

- Definition variables are created in the process definition and only that process can reference them. The value is persistent and shared across instances of that process.
- Input variables are also created in the process definition. They specify data (parameters) which can be passed into the process. Input variables can optionally be marked as required.
- Output variables are also created in the process definition. They specify data which the process can return to its caller.
- Local variables are also created in the process definition and only that process can reference them. However, a new instance of the variable is established for each process instance, so that one instance's interaction with the variable cannot affect another instance of the process.

**Related Topics**

- Defining Target Properties, page 6-4
- Adding Process and Global Variables, page 7-13

# Automation Summaries

An automation summary is a collection of data that summarizes the actions taken and data retrieved during the execution of the process. To publish an automation summary:

1. Add a Create Automation Summary activity to the process workflow.
2. Configure which activities from earlier in the workflow to publish in which sections of the document.

Technically, automation summaries are XML files that are published on a share or web server. The path or URL is then available to be linked from anywhere. This architecture makes them available for integration into virtually any context.

Automation Summaries are a key integration tool. For example, they can be placed in incidents in the Service Desk to reveal the analysis of the automation that was performed, or they can show how an incident was resolved.


# Getting Started

This topic explains how to connect to your first server, import automation packs, and configure the specifics of your IT environment.

## Connecting to Your First Server/Environment

This section contains instructions for connecting the console to your first Process Orchestrator server.

**Before You Begin**

- There are several user groups assigned to Process Orchestrator's built-in security roles. Before attempting to access the console, you must be a member of one of these user groups or another group to which your administrator has assigned custom role privileges. If you do not have access or the appropriate rights, please contact your local IT administrator.

To connect to your first Process Orchestrator server:

**Step 1** Choose **Start** > **Programs** > **Cisco > Cisco Process Orchestrator > Cisco Process Orchestrator Console**.

**Step 2** In the Select Server dialog, select a server name and click **Connect to Server**.

**Step 3** To configure the console server connection, choose **Tools > Options**, click the **General** tab, and update the parameters as necessary.

**Note** For additional information about the property fields, see the online help.

# Managing Automation Packs

## Importing Automation Packs

The Automation Pack Import Wizard imports the automation packs (see Process Orchestrator System Components, page 1-10). The wizard automatically launches:

- Only when the first server is installed in a high availability environment.
- After the installation is complete and the automation pack initialization is completed.
- The wizard does not fully launch until *after* the automation server has started.

To import automation packs *after* the Process Orchestrator has been installed, you can either:

- Use the Cisco installation path for shipped automation packs, **Start > All Programs > Cisco > CiscoProcess Orchestrator > Automation Packs Import Utility**.
- Run the Automation Pack Import Wizard from the console (see the following procedure).

#### Before You Begin

- To import an automation pack, you must be a Process Orchestrator Administrator.
- If you click **Cancel** during the import, the wizard will close and the automation pack will not be imported.
- Report definition files are extracted onto a local hard drive by the import process. Import the reports *after* the wizard completes.

To use the Automation Pack Import Wizard from the Process Orchestrator console to import automation packs:

**Step 1** Choose **Administration > Automation Packs**, right-click and choose **Import**.

**Step 2** Select the automation packs to be imported and click **OK**.

You can import multiple Cisco automation packs from multiple directories. The Automation Pack Import Wizard will verify that all of the dependent automation packs already exist in the database, or in the list of automation packs to import, before allowing the import to continue.

The automation packs that are available for import depend on the product license.

**Step 3** When the Welcome to the Automation Pack Import Wizard panel displays, click **Next**.

**Step 4** On the General Information Panel, you can optionally check **Disable all imported processes** to indicate that all processes from this tab should be disabled by default after being imported.

**Step 5** Review the information about the automation pack, then click **Next**.

**Step 6** Some automation packs might include additional steps in the wizard to configure elements of that automation pack. Follow the instructions to perform each configuration.

**Step 7** Some automation packs might include supplemental files that ship in the automation pack. These files are typically used to configure external systems, such as report definitions, that can be imported into your chosen reporting technology. If the automation pack includes these elements, you will see a Data Extraction panel. On this panel:

a. Verify the default location for where the Process Orchestrator-provided data files should be copied.

b. Under **Select data to extract**, check the appropriate check boxes to indicate which data files should be copied, then click **Next**.

**Step 8** The Review Prerequisites panel displays the status of each prerequisite:

- The green check mark verifies that the prerequisite was met on the computer.
- The red X determines that the prerequisite is not met on the computer. When this occurs, the import progress is stopped and cannot continue until all prerequisites have been met.

**Note** If all prerequisites are passed, the wizard automatically continues to the next panel which displays the status of the automation pack objects being imported. After the objects have been imported, the importing of the automation pack is complete.

**Step 9** If other automation packs were selected to be imported, the wizard will relaunch for the next automation pack, starting at Step 4. Otherwise, a Finish page will display the information about the automation packs that were imported. Click **Close**.

**Related Topics**

- Creating an Automation Pack, page 11-3
- Exporting an Automation Pack, page 11-9
- *Cisco Process Orchestrator Installation Guide*

# Extracting Data Files

Data files are embedded in some Process Orchestrator-provided automation packs. By default, you can extract these data files during the automation pack import process. However, if you decide not to extract the files at that time, use the following steps to extract the files in the automation pack to a location on the computer.

To extract the data files:

**Step 1** Choose **Administration > Automation Packs**, right-click and choose **Extract**.

**Step 2** Highlight the appropriate automation pack, then click **Open**.

**Step 3** In the Data Extraction dialog box, verify the default location for where the selected files are to be copied.

**Step 4** Check the check box to the left of the appropriate data file to indicate that the data file should be extracted, then click **OK**.

# Overwriting Existing Objects

Use the Overwrite Objects dialog box to confirm whether to overwrite existing objects that currently reside in Process Orchestrator. The Overwrite Objects dialog box displays only if both the following are true:

- The Cisco Process Orchestrator server contains existing objects with the same unique ID.
- The current user is the author of the object.

To overwrite the existing version of the object, check the check box to the left of the object, then click **OK**.

If the object belongs to a third-party automation pack, the object will be updated automatically. Any changed value of the object will be preserved.

## Deleting Automation Packs

If you import the wrong automation pack, use the following steps to delete it from Process Orchestrator.

**Note** Automation packs that have objects with dependencies cannot be deleted. First delete the dependent objects, then return to try deleting the automation pack again.

To delete an automation pack:

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Delete**.

**Step 2** On the Delete Confirmation dialog box, uncheck the **Select the checkbox to delete all member objects** check box to keep all member objects in the product.

**Note** This checkbox is available *only* to the automation pack's authors. If this check box remains checked, all objects in the automation pack will also be deleted.

**Step 3** Click **OK** to delete the automation pack.

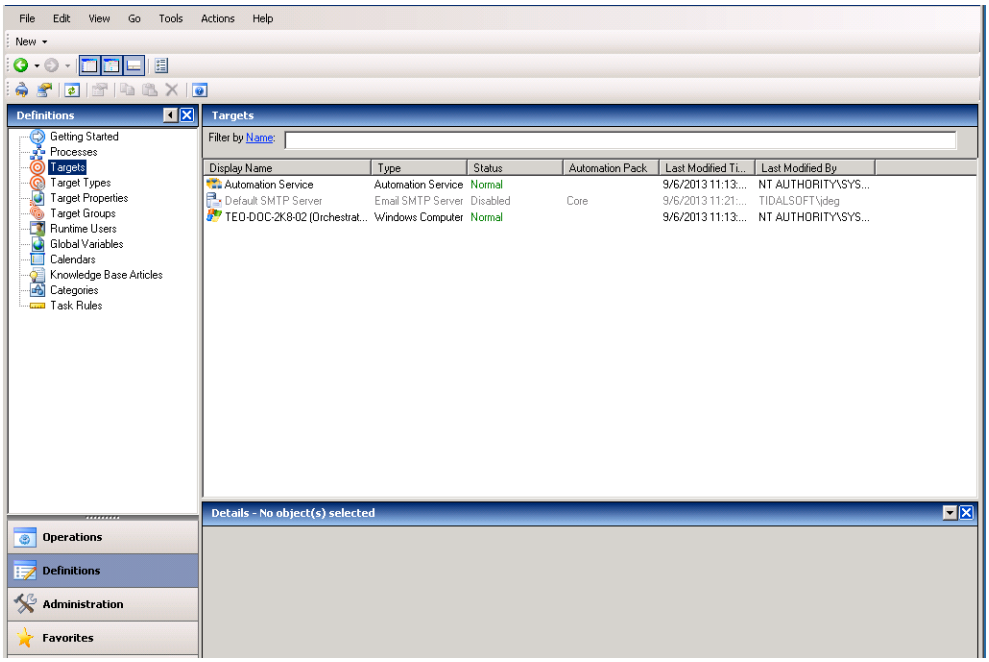
# Creating Targets

Targets specify where certain processes, activities, or triggers will run. You can define a target once and then reuse it in multiple processes. For more information about targets, see Targets, page 1-17.

**Note** Not all targets can be created manually. Some targets are discovered automatically based on the information specified by other targets.

The Definitions > Targets view displays all the existing defined targets. You can also use this view to create new targets, modify the properties of a target, and delete targets.

*Figure 2-1 Definitions—Targets*




To create a target:

**Step 1** Choose **Definitions**, right-click **Targets**, choose **New > [Target]**, and select a target type from the drop-down list.

**Step 2** Click the target-specific tabs and enter the appropriate information.

**Note** See the online help for information about defining the individual properties.

Any connection and credentials information will be verified by the wizard or property sheet for that target type. If the connection is *not* possible with the specified parameters, an appropriate error is displayed and must be corrected before the target can be created.

### Related Topics

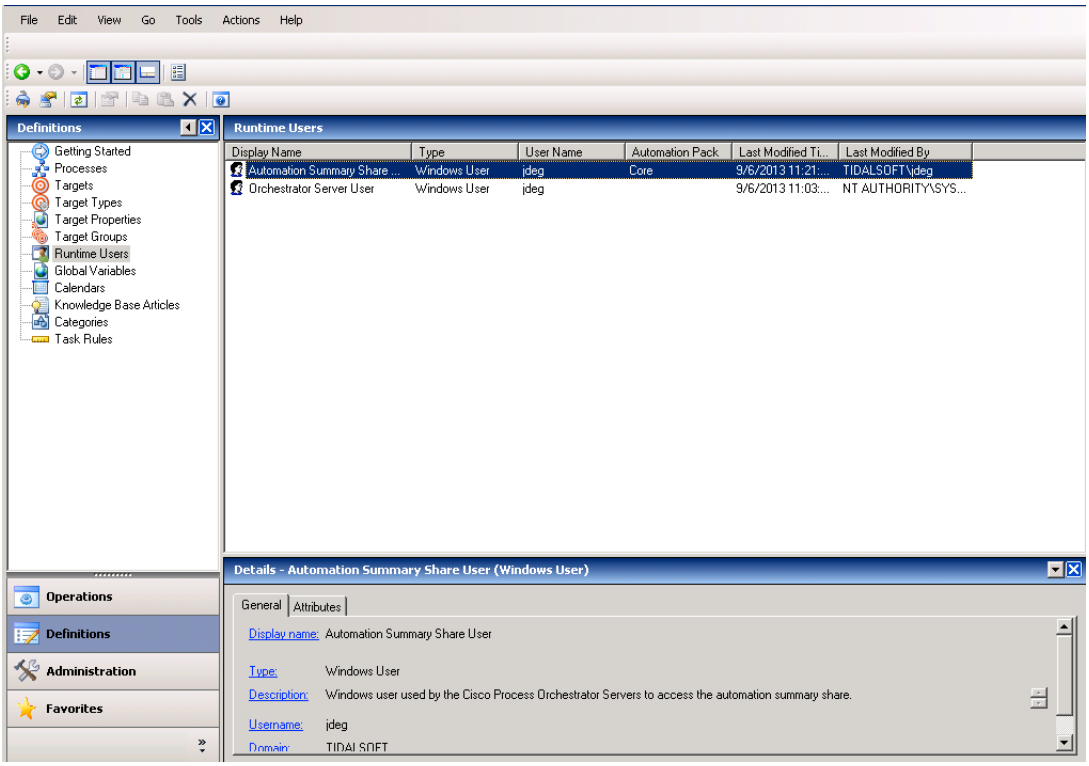
Deleting an Object, page B-6

# Defining Runtime Users

Runtime users hold the security credentials that are assigned to processes and activities. For more information about runtime users, see Runtime Users, page 1-18.

Use the Definitions > Runtime Users view to display the configured runtime users, add new runtime users to define new connection credentials, update runtime users such as to update a password, or delete runtime users.

***Figure 2-2 Definitions > Runtime Users***




To define a runtime user:

**Step 1** Choose **Definitions > Runtime Users**, right-click and choose **New > [Runtime User]** for the appropriate runtime user account.

**Step 2** On the [Runtime User] Properties property sheet, click the **General** tab and enter the appropriate runtime user-specific information. The property pages displayed depend on the type of runtime user selected.

**Step 3** The **Used By** tab displays objects that reference the runtime user. This tab will remain blank until the runtime user is used by an object.

**Step 4** The **History** tab displays the history of actions taken against the runtime user. This tab remains blank until after the initial creation.

**Step 5** To view information about an object, highlight the object, right-click and choose **Properties**.

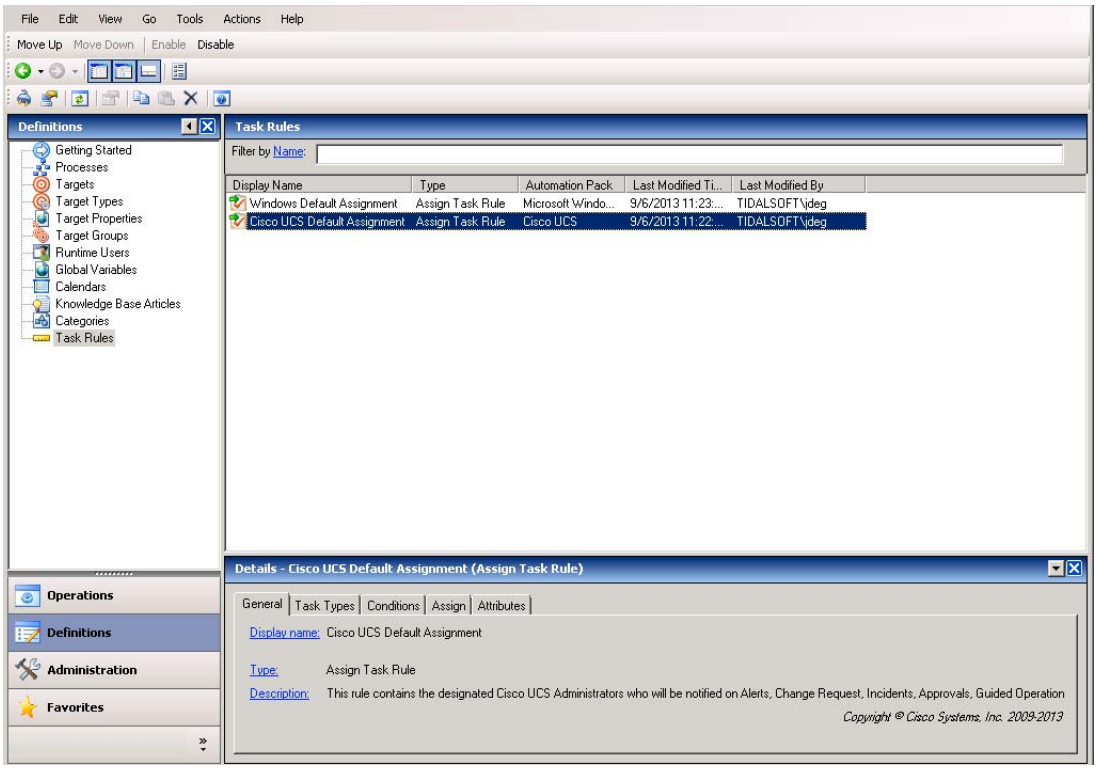
**Step 6** Click **OK** to close the property sheet.

# Defining Task Rules

Task rules are used to manage task assignment and notifications for various customer-specific tasks. By defining a task rule, you can tune the content you receive to your specific company without having to edit processes in the automation pack.

The Definitions > Task Rules view displays all tasks that have been configured to assign, notify, or update tasks based on specific settings. The display can be filtered to display task rules by properties such as, name, description, and automation pack.

***Figure 2-3 Definitions Workspace > Task Rules***




You can view default assignments, notifications, and properties to be updated according to the following defined rule types:

| Task Rules | Description |
|---|---|
| Assign Task Rule | Assigns users to the task |
| Notify Task Rule | Adds an entry to the notification list of a task |
| Update Task Rule | Specifies the properties to update in a task |

Whenever a task is created, Process Orchestrator goes through the settings and conditions of each task rule that is listed and enabled. If the conditions and settings in the rule are satisfied, the task rule is executed.

**Note** Task rules are executed according to the order in which they are displayed in the list.

### When Task Rules Execute

Task rules can not be manually run by the user. Task rules can only execute when a task is created within Process Orchestrator processes and rule conditions are met. If the task-created trigger (for any task type) and rule conditions are met, the action of the task rule is executed in the order in which it is listed in the Definitions—Task Rules view.

Any processes that are triggered based on a Task Created trigger will only execute after the appropriate task rules have executed.

### Notifications

In addition to notifications that occur when a user is assigned, often the same or other people must be notified in response to a task. The task rule notification identifies which person or group to notify and a process emails the notice.

Each task contains a list of notification recipients in addition to task assignees. The notification task rule adds to this list of notification recipients. A process can react to a task create event or task change event and then appropriately notify the notification recipients by email or any other mechanism.

### Automation Pack Rule Management

Task rules can be included in automation packs to easily transfer default task rules from one system to another, such as transferring task rules from a development system into a test or production system.

Some Cisco automation packs include not only processes that generate tasks, but also separate processes that handle notifications. You can create task rules to manage these objects without editing the process or task so that your customization to the tasks occurs before these notification processes run.

## Creating a Task Rule

### Before You Begin

By default, only users with administrative rights can create and update task rules from the Definitions > Task Rules view. However, the security settings can be changed, if necessary, by the Process Orchestrator administrator.

To create a task rule:

**Step 1** Choose **Definitions > Task Rules**, then on the toolbar, choose **New > [Task Rule]**.

**Step 2** On the **[Task Rule]** property pages, define the properties.

**Note** The property pages might display as display-only if the task rule definition is shipped as part of the product or you do not have the appropriate rights.

- Click the **Task Types** tab to select the task types to be executed by the rule.
- Click the **Conditions** tab and click the appropriate panel to indicate the type of condition equation to be used to trigger the task rule.
- Click the **Assign** tab and specify the assignees for task rule.

**Step 3** Click **OK** to close the property sheet.

You can create the following task rules:

*Table 2-1 Summary of Available Task Rules*

| Task Rule | Purpose |
|---|---|
| Assign Task Rule | Specify the users to be assigned to the task after the task rule has executed.<br>To modify this rule:<br>• To add, edit, or remove assignees, click the **Assign** tab on the [Task Rule] Properties property sheet. |
| Notify Task Rule | Specify the recipients to be notified after the task rule is executed.<br>To modify this rule:<br>• To add, edit, or remove recipients, click the **Notify** tab on the [Task Rule] Properties property sheet. |
| Update Task Rule | Specify the properties to be updated on the task after the task rule has executed.<br>To modify this rule:<br>• To add task properties to update after the task rule is executed:<br>a. Click the **Update** tab, click **Add** to add a new Property drop-down list to the Properties to Update section.<br>b. From the Property drop-down list, select the item to update in the task.<br>• To remove this rule:<br>The Remove button removes the last property added to the list. For example, you cannot remove *Property 4* without removing *Property 5* from the list.<br>To keep *Property 5*:<br>a. Update *Property 4* with the information from *Property 5*.<br>b. Click **Remove** to remove *Property 5* from the list. |

# Adding Conditions to a Task Rule

Use the following steps to add a condition to a task rule.

**Step 1** Choose **Definitions > Task Rules**, highlight the appropriate task, right-click and choose **Properties**.

**Step 2** On the [Task Rule] Properties property sheet, click the **Conditions** tab, then click the appropriate panel to indicate the type of condition equation to be used to trigger the task rule (see Creating a Task Rule, page 2-8).

**Step 3** Click **OK** to close the property sheet.

# Modifying Task Types in a Task Rule

Use the following steps to modify the list of task types affected by the task rule.

**Step 1** Choose **Definitions > Task Rules**, highlight the appropriate task rule, right-click, and choose **Properties**.

**Step 2** On the [Task Rule] Properties property sheet, click the **Task Type** tab, then use one of the following methods:

- To add a task type to the list, check the check box to the left of the appropriate task type.
- To remove a task type from the list, uncheck the check box to the left of the appropriate task type.
- To include all the task types into the task rule, click **Select All**.
- To exclude all the task types, click **Unselect All**.

**Step 3** Click **OK** to close the property sheet.


# Monitoring Operations

Use the Operations workspace to monitor the processes that are scheduled to execute, view processes that are currently running, and verify that processes have successfully completed. The workspace also provides auditing information related to system events generated by Cisco Process Orchestrator servers and adapters during operation.

The Operations workspace displays:

- The status of processes and activities executing, scheduled, or recently executed by the application.
- Alerts, incidents, and approvals in the product.
- System messages about status and errors within the Process Orchestrator server or adapters.

The options provided by this workspace include:

| Navigation Item | Function |
|---|---|
| Getting Started | Displays task shortcuts to view process instances, process instance activities, and the other system activity in the results pane.<br>For more information, see Getting Started > Operations, page 3-3. |
| Search Work | Displays the results of the query for processes and activities based on the specified filter options.<br>For more information, see Searching Tasks, page 3-15. |
| Target Views | Displays the defined targets and their associated process instances that are in progress, scheduled or have been completed.<br>For more information, see Monitoring Target Views, page 3-4. |
| Process Views | Includes all the defined processes and their associated activity instances that are in progress, scheduled or have been completed.<br>For more information, see Monitoring Process Views, page 3-6. |
| Activity Views | Includes a set of predefined views that displays information about processes, including the status of activities within the processes.<br>For more information, see Monitoring Activity Views, page 3-8. |
| Search Tasks | Displays the results of queried tasks based on the specified filter option.<br>For more information, see Searching Tasks, page 3-15. |
| Search IT Records | Displays only the alerts, incidents, and change requests created in Process Orchestrator.<br>For more information, see Searching IT Records, page 3-16. |

| | |
|---|---|
| Task Views | Displays all task activities and tasks that have been assigned to a specific user or group. Only users with administrative rights can create tasks from this view in Process Orchestrator.<br>For more information, see Monitoring Task Views, page 3-12. |
| Auditing | Displays the system logs for system events that have occurred within Cisco Process Orchestrator.<br>For more information, see Monitoring Auditing Information, page 3-16. |

The Process Views folder includes all the defined processes and their associated activity instances that are in progress, scheduled or have been completed. From this view, you can see detailed information about all processes that have been triggered, executed adhoc, or are scheduled for execution.

You can review both process and activity instances in the Process Views folders when querying the properties and status of process instances. Each Process View has two separate results panes:

- The top results pane displays the process definitions for the selected view. This view also allows you, if you have the appropriate rights, to access the Process Editor to modify the configuration of the process, to start the process, and enable or disable the process.
- The bottom results pane is automatically filtered to show *only* the instances of the process whose definition is selected in the top panel.

The Definitions > Processes view displays all defined processes that are in the application. This includes all processes that have been authored in the Process Editor and the processes that have been imported in automation packs.

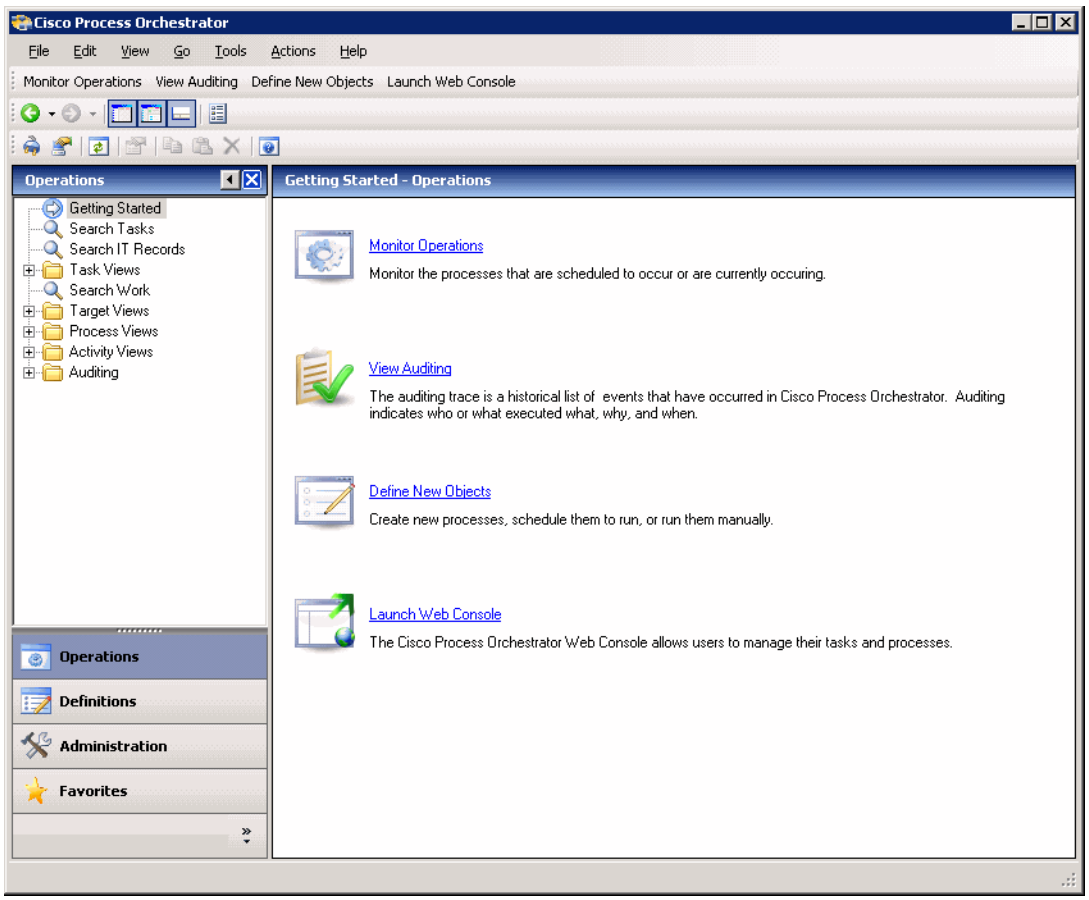
The following topics discuss some typical monitoring tasks:


- Getting Started > Operations, page 3-3
- Monitoring Target Views, page 3-4
- Monitoring Process Views, page 3-6
- Monitoring Activity Views, page 3-8
- Searching Work, page 3-11
- Monitoring Task Views, page 3-12
- Searching Tasks, page 3-15
- Searching IT Records, page 3-16
- Monitoring Auditing Information, page 3-16

# Getting Started > Operations

The Getting Started > Operations view displays task shortcuts to view process instances, process instance activities, and the other system activity in the results pane.

***Figure 3-1 Operations > Getting Started***




The following tasks can be performed from this view:

| Navigation Item | Function |
|---|---|
| Monitor Operations | Launches the Activity Views folder which includes the options to display information about the activities that are in progress, scheduled or have been completed.<br>See Monitoring Activity Views, page 3-8 |
| View Auditing | Launches the Auditing view which displays the system event history for actions that have occurred within Process Orchestrator.<br>See Monitoring Auditing Information, page 3-16. |

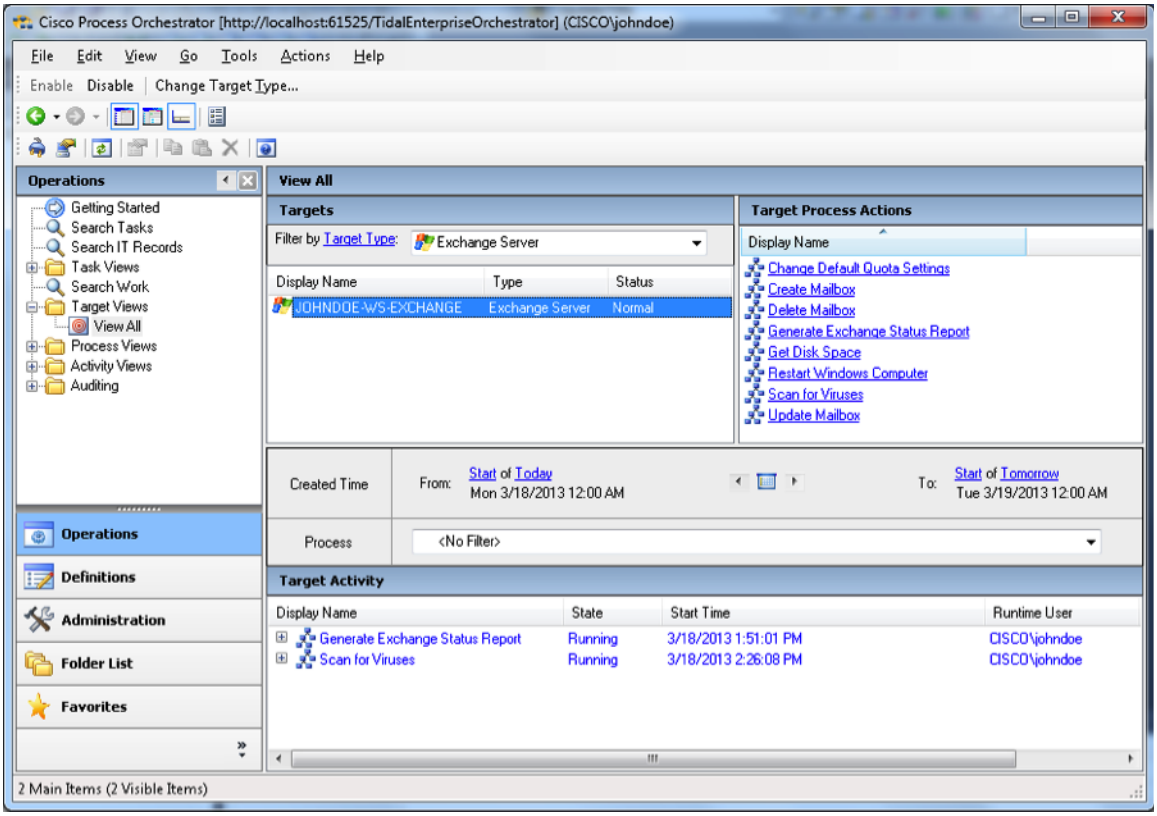
| | |
|---|---|
| Define New Objects | Launches the Getting Started—Definitions view which displays the options for creating a new process and defining new objects to be associated with a process.<br>See Chapter 7, "Authoring Processes." |
| Launch Web Console | Launches the Process Orchestrator web console to provide users with a web-based view of the scaled-back Operations workspace on the expert Process Orchestrator console.<br>See Chapter 13, "Using the Web Console." |

# Monitoring Target Views

Target views allow you to view ongoing automation in the context of the targets or services on which it acts. You can filter the list of all targets to focus in on a specific type, or by the name of a target such as an Cisco Intelligent Automation for Cloud service request. After selecting a target, you can see the processes that act on that target type and all process activity related to that target.

The Operations > Target Views folder displays the defined targets and their associated activity instances that are in progress, scheduled or have been completed.

***Figure 3-2 Operations > Target Views***



# Monitoring Target Information

To monitor target information:

**Step 1** Choose **Operations > Target Views > View All**.

From this view, you can see detailed information about all targets, regardless of whether processes have or have not been executed against these targets. This view contains the following information:

- The Target Process Actions pane (top right) contains items that perform specific actions associated with the target selected in the Targets pane.
- The Target Activity pane (bottom) displays the processes that ran or are scheduled to run against the target selected in the Targets pane.

**Step 2** Select a target (in the top pane) for which you are interested in viewing the target activity. The list of processes appears in the Target Activity pane.

**Step 3** You can filter the results by process name, a range of dates, by a start or end period, by days to offset, by a specific time period, and so on.

**Step 4** To view additional (read-only) information about a process, highlight it and use one of the following methods to launch the Process Viewer:

- From the Target Activity pane, double-click the appropriate process instance.
- From the Details pane, click the hyperlink of any item on the General or Run Options tabs.

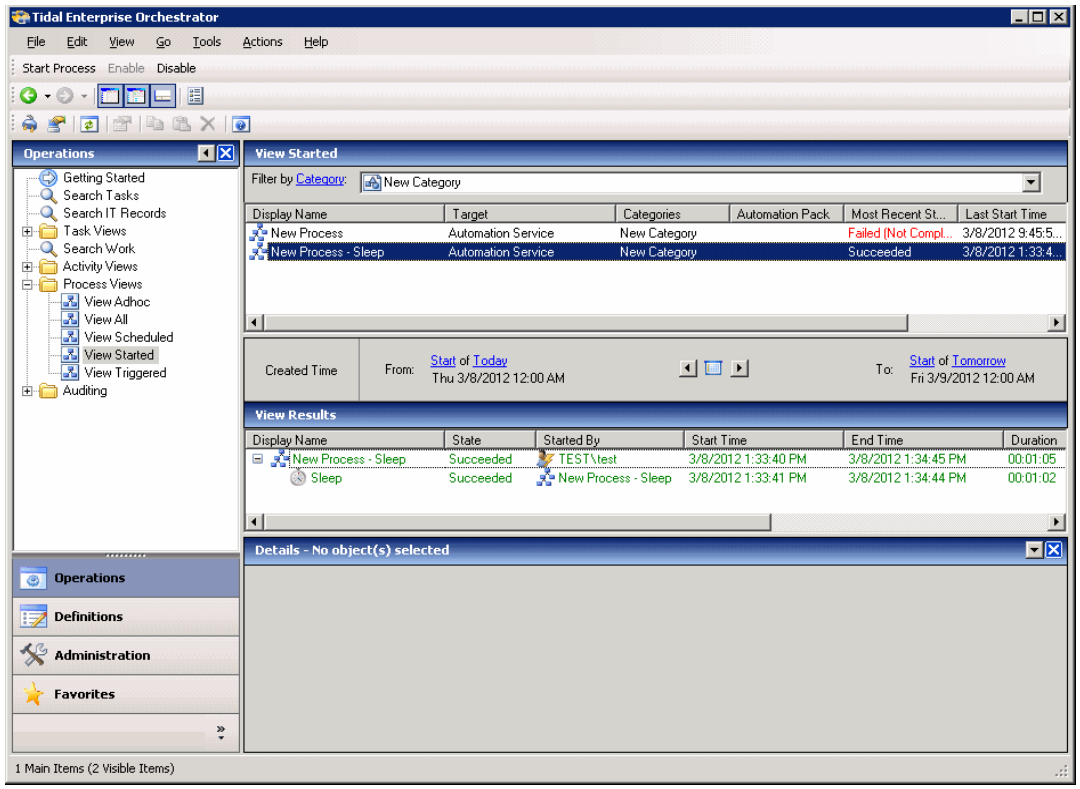
For more information about the information displayed in the Process Viewer, see Process Viewer, page A-8.

# Monitoring Process Views

The Operations > Process View folders display all processes that have been started, scheduled to start, started by a trigger, or started ad hoc. Each Process view display can be filtered to display by the name, category, or column view.

Use the Definitions > Processes View to display the defined processes.

***Figure 3-3 Operations > Process Views***




**Note** To add, remove, or sort column headings on the display, see Configuring Columns, page B-4.

# Monitoring Process Information

The Process View displays all the defined processes and their associated activity instances that are in progress, scheduled or have been completed. From this view you can view detailed information about all the processes that have been triggered, executed adhoc, or are scheduled for execution.

To view process information:

**Step 1** Choose **Operations > Process Views**, then choose one of the following options.

| Process View | Description |
|---|---|
| View Adhoc | Displays all process or activity instances that were executed manually and are in progress, have successfully completed, or failed |
| View All | Displays all process, activity, and scheduled process and activity instances that are in progress, have successfully completed, or failed |
| View Scheduled | Displays all process or activity instances that are in progress, have successfully completed, or failed and are also scheduled to execute |
| View Started | Displays all process or activity instances that were executed (manually or automatically) and are in progress, have successfully completed, or failed |
| View Triggered | Displays only processes that were “triggered” (not processes that were manually executed or scheduled). |

**Step 2** Select a process definition (in the top panel) for which you are interested in viewing the instances. The process information appears in the View Results pane.

**Step 3** You can filter the results by a range of dates, by a start or end period, by days to offset, by a specific time period, and so on.

**Step 4** To view additional (read-only) information about an instance, highlight it and use one of the following methods to launch the Process Viewer:

- From the Results pane, double-click the appropriate process instance.
- From the Details pane, click the hyperlink of any item on the General or Run Options tabs.

For more information about the information displayed in the Process Viewer, see Process Viewer, page A-8.

# Canceling Running Process Instances

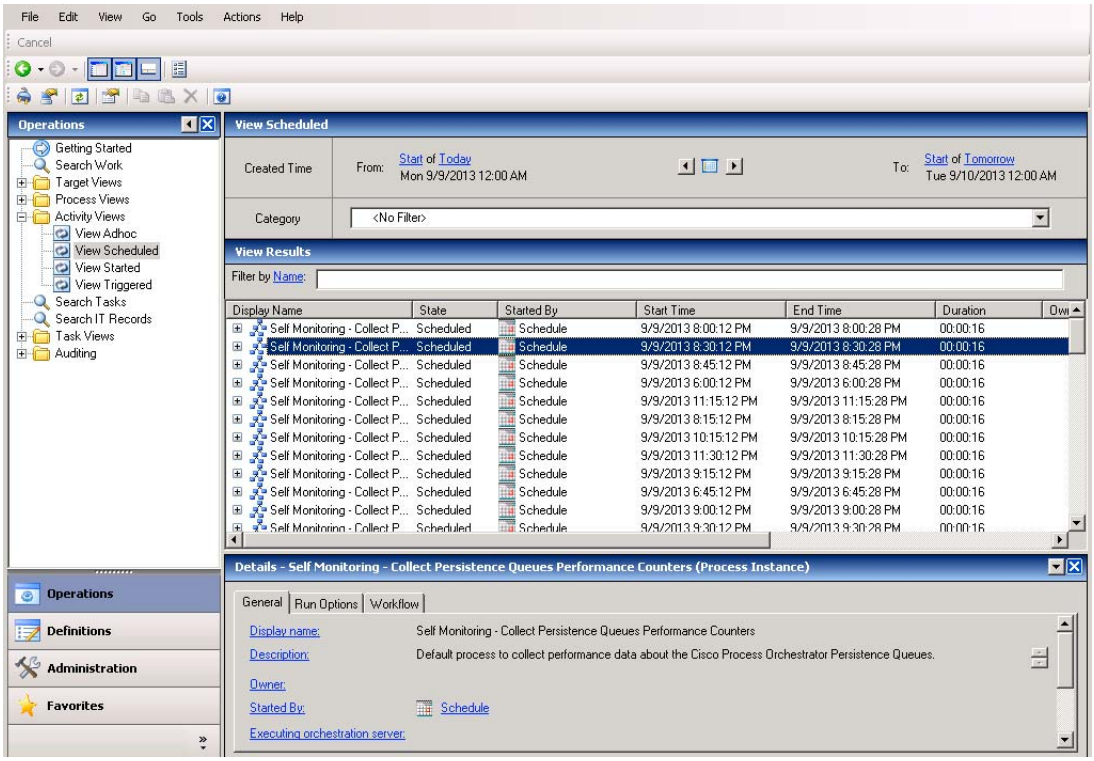
You can only cancel running process instances from the bottom View Results pane on any of the Process Views. You cannot cancel processes from the top Process results pane.

# Monitoring Activity Views

The Operations > Activity Views folders display processes and their associated activities that have been started, scheduled to start, started by a trigger, or started ad hoc. Each Activity view display can be filtered to display by the date/time of execution, category, or column view.

From this view, detailed information is provided about all the activity instances that are started, running and scheduled to run. If you have the appropriate security rights, you can create and modify tasks from this view.

***Figure 3-4 Operations > Activity Views***




# Viewing Activity Information

Activity Views include a set of predefined views that display information about the status of activities within the processes. From this view, you can view activities that have been triggered, executed adhoc, view all activities, and view all scheduled activities.

Certain activities display additional activity instance information. For example, there will be certain activities that generate information based on the defined properties of the activities. In those situations, the activity instance properties displayed from the Operations > Activity View will show the read-only configuration properties as well as the generated results of the configuration properties.

To view activity information:

**Step 1** Choose **Operations > Activity Views**, then choose one of the following options.

| Views | Descriptions |
|---|---|
| View Adhoc | Displays all process or activity instances that were executed manually and are in progress, have successfully completed, or failed during the selected time period |
| View Scheduled | Displays all process or activity instances that are scheduled to execute during the selected time period |
| View Started | Displays all process or activity instances that are in progress, have successfully completed, or failed during the selected display time period |
| View Triggered | Displays all process or activity instances that were triggered for execution (manually or automatically) and are in progress, have successfully completed, or failed during the selected time period |

A list of activity instances appears in the Results pane.

**Step 2** You can filter the results by category, a range of dates, by a start or end period, by days to offset, by a specific time period, and so on.

**Step 3** To view additional (read-only) information about an instance, highlight it and use one of the following methods to launch the Process Viewer:

- From the Results pane, double-click the appropriate process instance.
- From the Details pane, click the hyperlink of any item on the General or Run Options tabs.

For more information about the information displayed in the Process Viewer, see Process Viewer, page A-8.

# Modifying Activity View Format

To change an Activity View format, highlight an activity instance, right-click, and choose **View > [Format].** Each of the activity views can display in one of these formats:.

| Activity View | Description |
|---|---|
| Flat | Displays all process instances and activity instances on the top-level with no expandable tree view. |
| Workflow | Displays the process instances and only one workflow instance (even if it has run multiple times) in an expandable tree-view format. |
| Activity (Default view) | Displays the process instances and all related activity instances (no matter how many times it has run) in an expandable tree-view format. |

# Understanding Activity View Indicators

The process and activity views have certain indicators that provide status information.

### Status Indicators

The State column displays the status of the individual process and activity. The following indicators definitions display in the Results pane.

| Status | Description |
|---|---|
| Succeeded | Process has completed successfully |
| Running | Process is in progress |
| Failed (Not Completed) | The process has failed and did not complete the process execution |
| Failed (Cancelled) | The process has been cancelled manually |

### Color Indicators

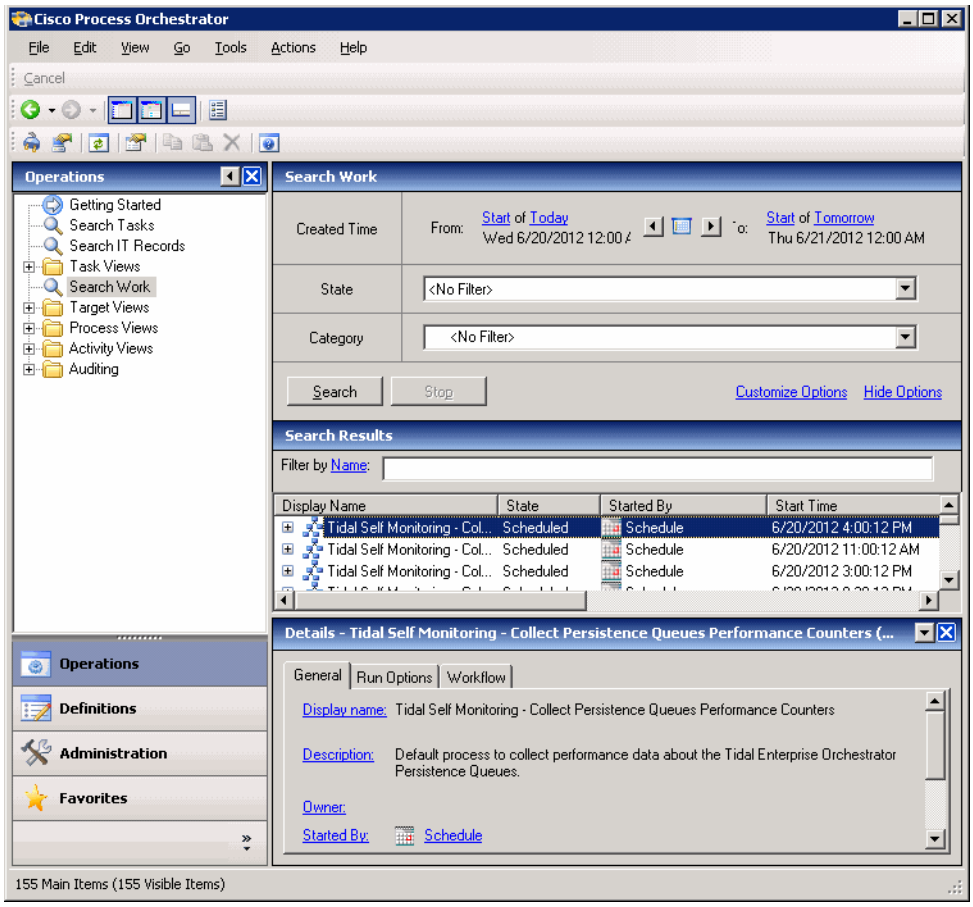
The colors associated with the individual activities determine the status of the process and activity instances. To customize these colors, choose **Tools > Customize**. For more information about customizing the color indicators, see the online help.

# Searching Work

The Operations > Search Work view displays the results of the query for processes and activities based on the specified filter options. The Search Work view displays all processes and activities that are scheduled to be performed during the selected display time period.

On the Operations workspace, choose **Search Work**.

***Figure 3-5 Operations Workspace > Search Work***




To customize the information to be displayed in the Search Results table:

**Step 1** Choose **Operations > Search Work**.

**Step 2** In the Search Work dialog box, click **Customize Options**.

**Step 3** In the Select Search Filters dialog box, click the check boxes to the left of the appropriate options. If a check box remains unchecked, the results for that filter option will not display in the Search Results table. Click **OK.** The query results appear in the View Results pane.

**Step 4** You can filter the results by category, a range of dates, by a start or end period, by days to offset, by a specific time period, and so on.

**Step 5** To view additional (read-only) information about an instance, highlight it and use one of the following methods to launch the Process Viewer:

- From the Results pane, double-click the appropriate process instance.
- From the Details pane, click the hyperlink of any item on the General or Run Options tabs.

For more information about the information displayed in the Process Viewer, see Process Viewer, page A-8.

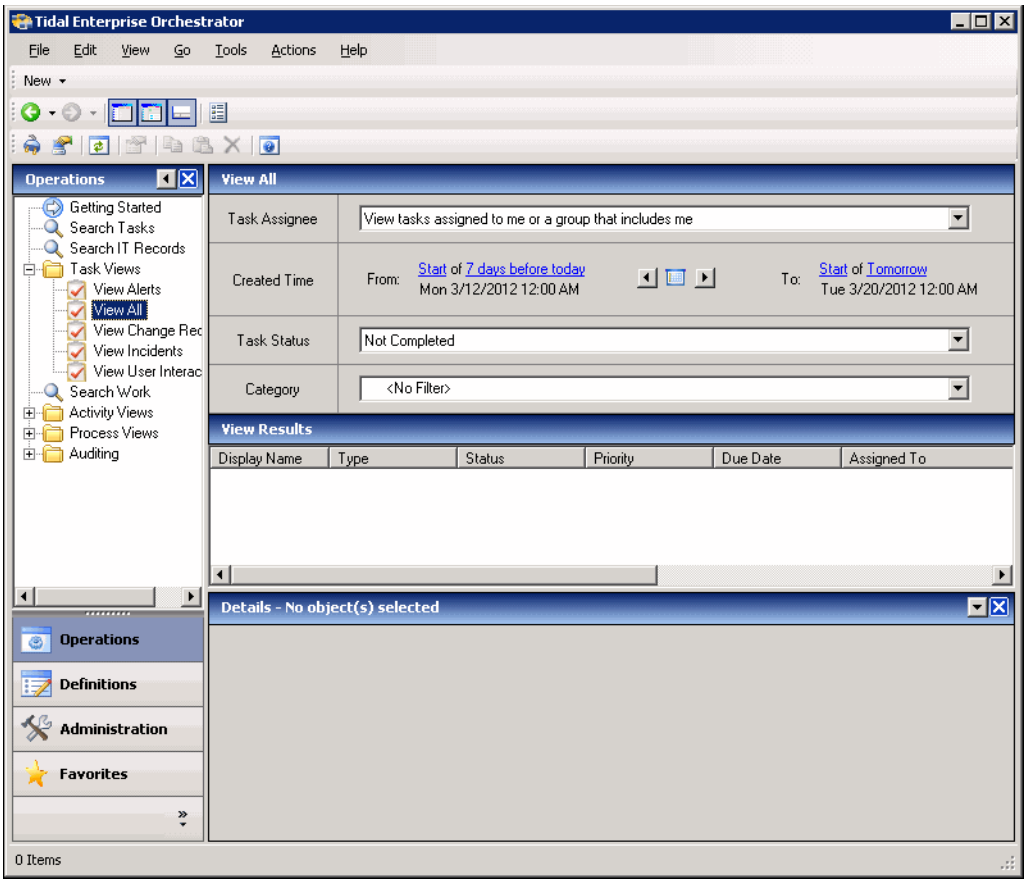
# Monitoring Task Views

Tasks allow a process to create an IT record or perform human interaction, and provide ownership and a lifecycle. Process Orchestrator enables the user to act on tasks within the operator console, which managers can use to set or change assignments to those on their staff. Assignment changes, state changes, and all other changes are audited, so you can keep track of exactly who did what in the system.

The Operations > Tasks view displays all tasks or all tasks that have been assigned to a specific user or group.

The Tasks display can be filtered to display by category, task type, and status. You can also decide which task view to display.

***Figure 3-6 Operations > Tasks > View All***



# Displaying Task Views

From this view, detailed information is provided about all the alerts, incidents, and other tasks that are assigned, in progress, or have been completed. If you have the appropriate security rights, you can create and modify tasks from this view.

**Step 1** Choose **Operations > Task Views**.

**Step 2** To access detailed information about a task view, click the **Task Views** folder and choose one of the following options.

| Task View | Description |
|---|---|
| View Alerts | Displays all alerts |
| View All | Displays all tasks that are assigned, in progress, or have successfully completed |
| View Change Requests | Displays all change requests |
| View Incidents | Displays all incidents |
| View User Interactions | Displays all tasks that require user action |

**Step 3** On the Task Views Header, choose the appropriate filtering options to display.

**Note** For additional information about the property fields, see the online help.

# Acting on Tasks

The following table summarizes some of the types of changes you can make to a task.

*Table 3-1 Summary of Actions*

| Action | Purpose |
|---|---|
| Changing a task state | See Resolving a Task, page 3-14. |
| Adding a related task | Related tasks could be duplicates of one another or the resolution of the task that is dependant on the solution of the existing task.<br>Choose **Operations > Task Views**, highlight the appropriate task, right-click and choose **Properties > Related > Add.** |
| Adding a option to the input request select question | Create options for the input request select question. |
| Assigning a category to a task | Choose **Operations > Task Views**, highlight the appropriate task, right-click and choose **Properties** > **Categories**.<br>For information about managing categories, see Chapter 10, "Authoring Categories." |

*Table 3-1 Summary of Actions (continued)*

| Action | Purpose |
|---|---|
| Assign a duplicate alert | Indicate that the task is a duplicate of an existing task.<br>Choose **Operations > Task Views**, highlight the appropriate task, right-click and choose **Properties** > **Related.** |
| Modifying the assignment properties for a task | Change the assignee or other assignment properties for the task. This includes taking ownership of the task, if necessary.<br>Choose **Operations > Task Views**, highlight the appropriate task, right-click and choose **Properties > Assignment > Assigned to** |
| Adding parameters for a task | Adding parameters for a task. A single task can include up to ten parameters.<br>Choose **Operations > Task Views**, highlight the appropriate task, right-click and choose **Properties** > **Parameters > Add**. |

# Viewing Task Properties

To view the task properties for a specified tasks:

**Step 1** Choose **Operations > Task View**, highlight the appropriate task, right-click and choose **Properties**.

**Step 2** Review the properties, then click **OK**.

**Note** For additional information about the property fields, see the online help.

# Resolving a Task

Most users cannot resolve a task from the Process Orchestrator console because they do not have the rights to modify the task. In general, the user will launch the web console (see Resolving Tasks in the Web Console, page 13-3) when attempting to resolve a task.

To resolve a task:

**Step 1** Choose **Operations > Task Views**, highlight the appropriate task, right-click and choose one of the following:

- Resolve
- Select a choice

**Step 2** Review the details of the task.

**Note** For additional information about the property fields, see the online help.

**Step 3** From the Status drop-down list, choose the appropriate status for the task resolution.

**Step 4** In the Add Notes text box, enter any notes related to the task resolution or status update.

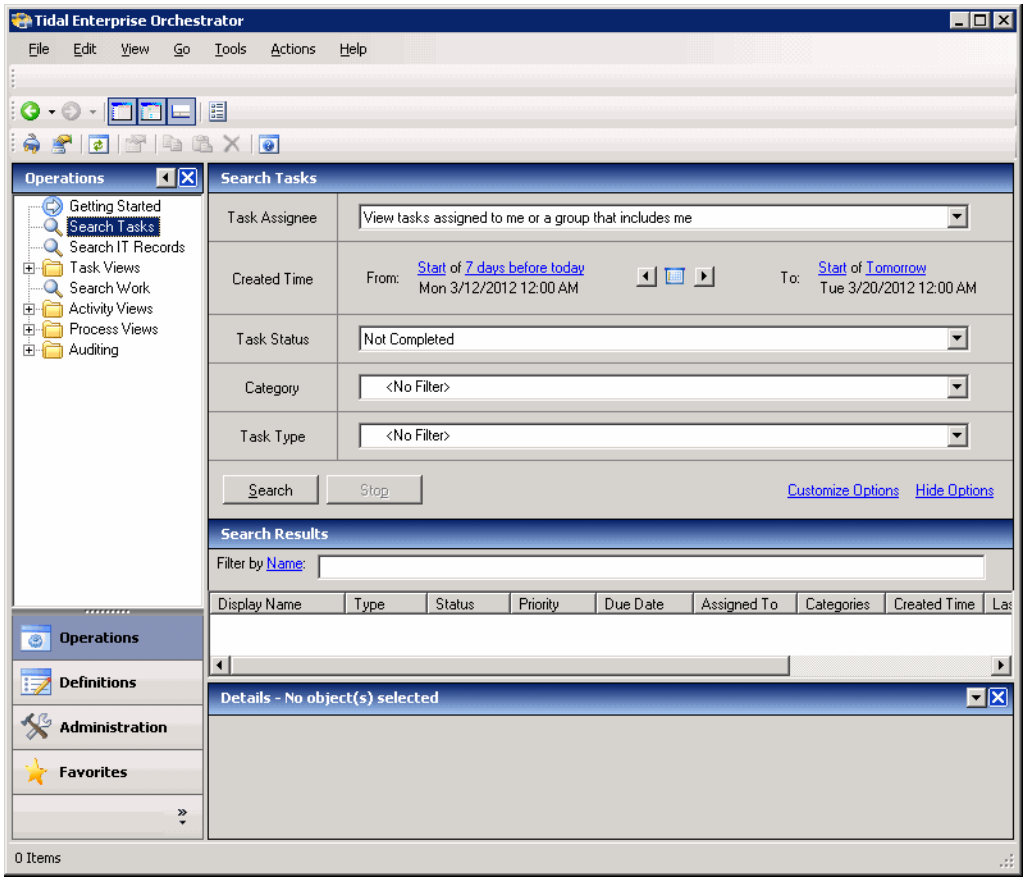
**Step 5** Click **Submit** to resolve the task.

The Task details page is updated and displays the user name of who resolved or updated the task.

# Searching Tasks

The Operations > Search Tasks view displays all tasks created or defined in Process Orchestrator. The results of the query based on the specified filter options. You can search by the assignee, category, task type, or status.

***Figure 3-7 Operations Workspace > Search Tasks***




For information about the fields in this workspace, see the online help.

## Customizing the Search Tasks Header

Use the Select Search Filters dialog box to customize the query header options. If the check box to the left of the option remains unchecked, then the filter option will not display on the designated Search header.

To customize the search filter options:

**Step 1** Choose **Operations—Search Tasks > Customize Options**.

**Step 2** In the Select Search Filters dialog box, check the check box to the right of the appropriate option, then click **OK**.

**Note** For information about the filter options, see the online help.

## Filtering Search Results

Use the Filterpane in the Task Views header to display selected tasks in the results pane.

To filter task search results:

**Step 1** Choose **Operations > Search Tasks.**

**Step 2** The **Filter by** drop-down list in the Task Views header contains several options that can be used to filter the results. Enter the search query information, then click **Search** to query the tasks.

# Searching IT Records

The Operations—Search IT Records view displays only the alerts, incidents, and change requests created in Process Orchestrator. The results of the query are based on the specified filter options. You can search by the assignee, category, task type, and status.

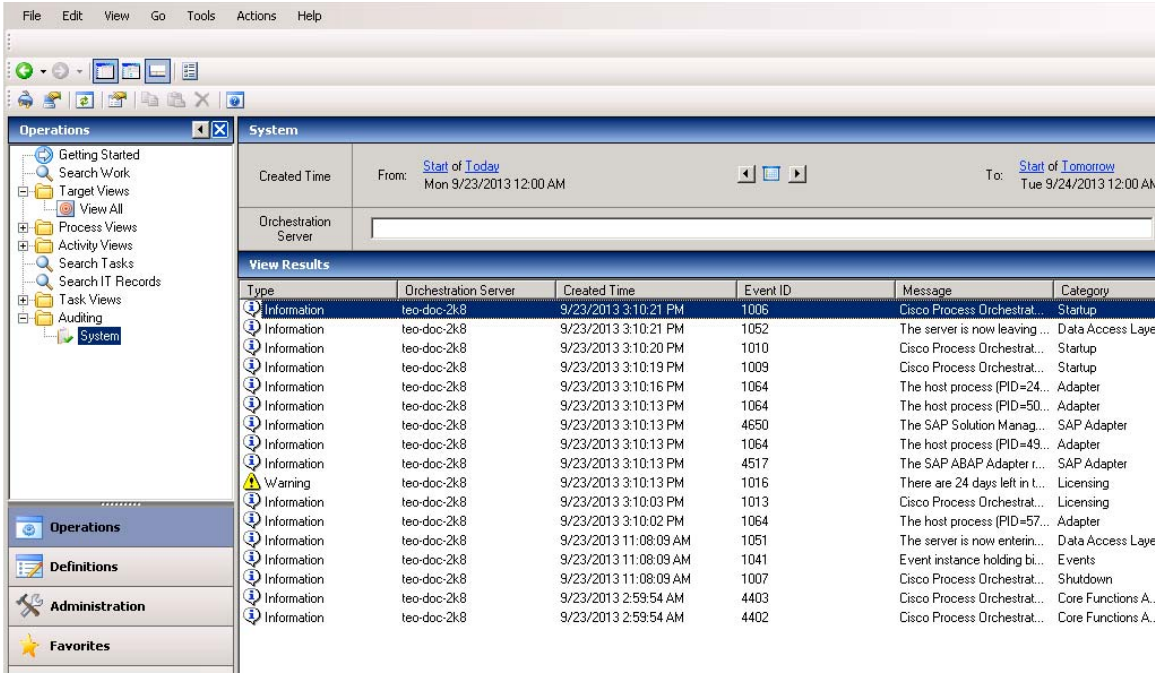
For information about the fields in this workspace, see the online help.

# Monitoring Auditing Information

The Auditing View displays the system auditing logs for system events that have occurred within Cisco Process Orchestrator for each Process Orchestrator server in the environment.Use this view to:

- Display detailed information regarding system events that have occurred within Process Orchestrator, such as start and shut down time and error occurrences.
- Filter the results by Orchestration Server name.

*Figure 3-8 Operations > Auditing View*




To view the details in an audit log:

**Step 1** Choose **Operations > Auditing > System**.

**Step 2** To view the audit logs for a specific server in your environment, enter the server name in the **Orchestration Server** filter text box.

**Step 3** From the Auditing Log Results pane, highlight the appropriate log entry, right-click and choose **Properties**.

**Note** For details about the log entry information, see the online help.

**Step 4** To add, remove, or sort column headings on the display, see Configuring Columns, page B-4.

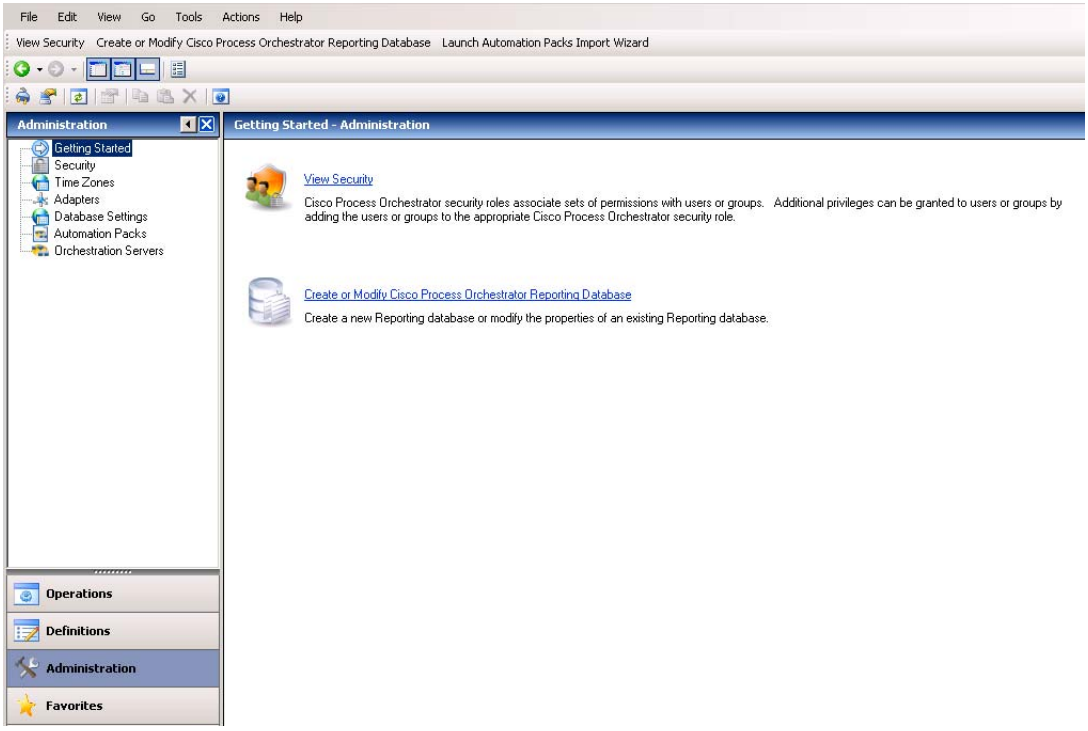

# Administration

Use the Administration workspace to perform administrative actions within the product, including:

- Managing Process Orchestrator servers in a high availability environment
- Determining what Process Orchestrator objects users are allowed to view, create, or modify
- Enabling time zones to be used when configuring processes
- Configuring adapters
- Creating and managing the report database, and running reports

The Administration > Getting Started view provides the initial access to the Administration pane.

***Figure 4-1 Administration > Getting Started***



In this view, the following administrative actions are available:

| Options | Description |
|---|---|
| View Security | Launches the Administration > Security view to display the rules that ship with the product |
| Create or Modify Cisco Process Orchestrator Reporting Database | Launches the Administration > Database Settings view where you can view reports, create or modify the report database |

The Administration workspace contains navigation items that perform the various actions that are available for the user. The Administration Results pane contains the following column headings.

| Navigation Item | Function |
|---|---|
| Security | The Administration > Security view displays the security rules that are shipped with the product and/or defined by the user. |
| Time Zones | The Administration > Time Zones view to displays the time zones that ship with the product. |
| Adapters | The Administration > Adapters view displays the adapters that are installed with the product and their associated objects. |
| Database Settings | The Administration > Database Settings view displays the reporting and processes databases associated with Process Orchestrator. |
| Automation Packs | The Administration > Automation Pack view displays the list of automation packs that are currently in Process Orchestrator. |
| Orchestration Servers | The Orchestration Servers view displays the Process Orchestrator servers that are currently configured in the environment. |

The following topics guide you through accessing the Administration workspace and using the features in this view.



# Viewing Process Orchestrator Server Ports

To display the ports on which a Process Orchestrator server is listening:

**Step 1** Choose **Administration > Orchestration Servers,** and double-click the server name.

**Step 2** In the Properties dialog, click the **Ports** tab.

**Note** For information about the field parameters, see the online help.

# Updating the Product License

Cisco Process Orchestrator is packaged with a trial license that is valid for a specified number of days. The trial license can be used until the license expires or is updated with a permanent license.

After the trial license expires, you must enter the appropriate license information (the license must be updated to a permanent license) to continue using the product.

To update your license

**Step 1** Choose **File > Update License**.

**Step 2** In the License Information panel, enter the appropriate information, then click **Next**.

**Note** For information about the field parameters, see the online help.

**Step 3** Select the **I agree** radio button then click **Next**.

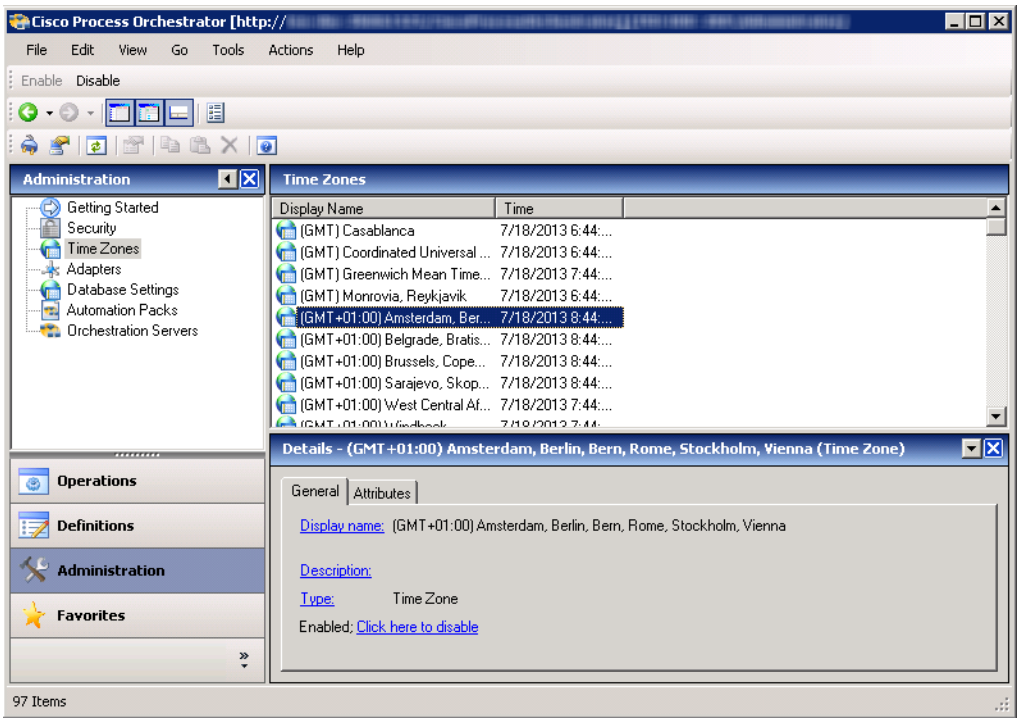
**Step 4** When the Completing the Update License Wizard panel displays, click **Finish**.

**Step 5** When the License was successfully updated message dialog box displays, click **OK**.

# Managing Time Zone Properties

Use the Administration > Time Zones view to display the time zones that ship with the product. The time zones are used when specifying a schedule trigger for a process. The time zones cannot be modified. However, you can add a description to the time zone properties and enable or disable the time zones. If a time zone is disabled, it will not be available for selection when defining schedule properties.

***Figure 4-2 Administration > Time Zones***




You can update the properties of a specific time zone using the Time Zone Properties property sheet. From this property sheet, you can enter a description for the time zone and enable or disable the time zone. Click the **Used By** tab to display the processes or schedule definitions that reference the time zone.

To update the properties of each security rule:

**Step 1** Choose **Administration > Time Zones**, right-click a time zone, then choose **Properties**.

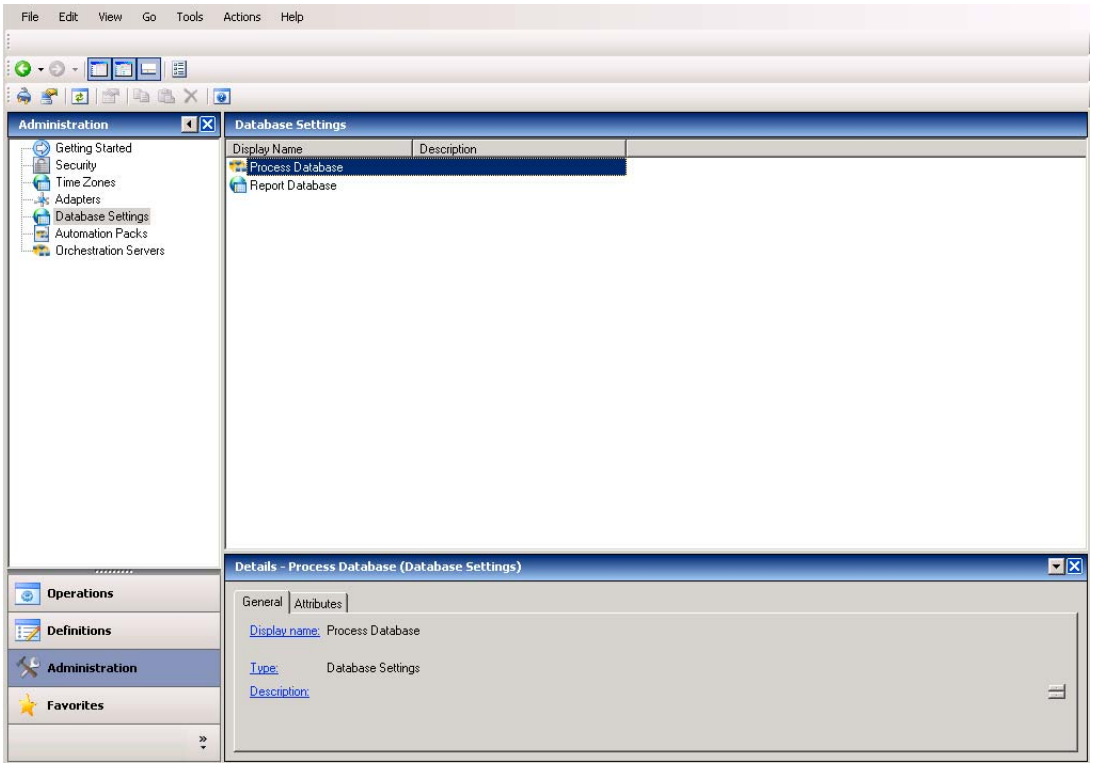
**Step 2** In the [time zone] Properties dialog, enter the appropriate information, then click **Next**.

**Note** For information about the properties fields, see the online help.

# Managing Database Settings

Use the Administration > Database Settings view to create and manage the database connections included in the product.

***Figure 4-3 Administration Database Settings View***




For example:

- Database Settings > Process Database contains the database grooming settings properties for the processes database. This dialog also contains the Groom Now option that allows you to manually start the database grooming process.
- Select the Reporting database to import reports, view database properties, and delete the database from the product.
- To view database properties, highlight the appropriate database, right-click and choose **Properties.**

To manage the database connections:

**Step 1** Choose **Administration > Database Settings**.

**Step 2** Choose one of these databases:

| Database | Description |
|---|---|
| Process Database | The process database displays the properties for the Process Orchestrator performance database. For information about configuring thegrooming settings of the process database, see Managing the Process Database, page 4-6. |
| Report Database | The report database generates reports for viewing process execution history and to audit process changes. For information about managing reports, see Managing the Report Database, page 4-6 |

# Managing the Process Database

The Process Database is created during the installation process, but only the default size of the data files is determined. Use the Process Database Properties property sheet to determine the amount of data that is archived in the database files before being deleted from the system.

To configure the database settings:

**Step 1** Choose **Administration > Database Settings**.

**Step 2** Highlight **Process Database**, right-click and choose **Properties**.

**Step 3** In the Process Database Properties property sheet, click the **Configuration** tab.

**Step 4** To modify the database grooming settings for the automation database server, specify a value for the grooming fields, then click **OK**. Or click **Groom Now** to start the grooming process immediately.

**Note** For more information about the database parameters, see the online help.

# Managing the Report Database

Process Orchestrator supports two reporting technologies:

- SQL Server Reporting Services. Customers who install Microsoft SQL Server as their chosen database platform can add the Reporting Services technology for free by simply specifying the component during SQL Server installation. SQL Server Reporting Services is only supported for Process Orchestrator databases hosted on an SQL Server.
- SAP BusinessObjects. BusinessObjects is the only solution that supports reporting against an Oracle Process Orchestrator database.

The reporting technology is installed separately from the product. Typically a reporting installation spans Process Orchestrator servers.

# Creating a Reporting Database Connection

The Process Orchestrator Reporting database is initially created during the installation process. There are occasions when the report database connection may need to be removed and then re-connected.

**Before You Begin**

- You must be logged in with an account that has Administrator rights on the machine where the report database is being created, whether it is the local machine or a remote machine.
- You must also have Reporting Services permission to create or modify the reports as create or modify permission over the Report Database object within Cisco Process Orchestrator.

To create the report database connection:

**Step 1** Choose **Administration > Database Settings**, right-click **Report Database** and choose **Create Process Orchestrator Reporting Database**.

**Step 2** The Welcome to the Report Database Configuration Wizard panel displays. Click **Next**.

**Step 3** The fields on the New Process Orchestrator Reporting Database panel (excluding the Password field) are automatically populated with the name of the local server and the credentials of the currently logged in user. Use *one* of the following methods:

- To use the credentials that are autopopulated for the default user, enter the password in the Password field.
- Specify the credentials for a different user, then click **Next**.

**Step 4** The fields on the Database Settings panel are automatically populated with the default locations and data grooming settings for the database. Use *one* of the following methods:

- To use the information that is auto-populated, click **Next**.
- To modify the data and transaction log file settings for the new database, specify the appropriate information, then click **Next**.

**Step 5** When the dialog box indicates that the database has been successfully created, click **Next**.

**Step 6** Verify the information and click **Finish** to complete the procedure and close the wizard.

# Importing Reports into SQL Reporting Services

During the Core Automation Pack import process, SQL Server Reporting Services report definition files are copied to the file path determined on the Core Report panel. To view the reports in SQL Server Reporting Services Report Manager, the report definition files must be imported into the Report Manager using the Report Database Import Report wizard.

Use the Import Reports process to import new reports into the report database.

### Before You Begin

- To import reports, you must be logged in with an account that has Administrator rights on the machine where the report database is being created, whether it is the local machine or a remote machine.
- Your account must also have Reporting Services permission to create or modify the reports as create or modify permission over the Report Database object within Process Orchestrator.

To import reports:

**Step 1** Choose **Administration > Database Settings**, right-click **Report Database** and choose **Import Reports**.

**Step 2** In the Report Database Import Report wizard, enter the location of the report files to be imported into the Report Manager. For example:

```
Cisco/Cisco Process Orchestrator/ExtractedData/SQL Server Reporting Services Reports
```

**Step 3** After the reports have been imported, click **Next**.

**Step 4** Verify the URL of the report server, then click **Finish**.

# Importing Reports into BusinessObjects

During the Core Automation Pack import process, BIAR files are copied to the file path determined on the Core Report panel. To view the reports in BusinessObjects InfoView, the BIAR file must be imported onto the BusinessObjects server using the BusinessObjects Import Wizard.

### Before You Begin

To import BIAR files, you must be logged in with an account that has the appropriate permissions on the Business Object server, whether it is the local machine or a remote machine.

Use the following steps to import reports into BusinessObjects.

**Step 1** Choose **Start > All Programs > BusinessObjects XI.3.1 > BusinessObjects Enterprise > Import Wizard**.

**Step 2** From the Welcome to the Import Wizard panel language drop-down list, choose the appropriate language to use to import the reports (English is the default), then click **Next**.

**Note** For information about the field parameters, see the online help.

**Step 3** On the Source environment panel, specify the environment to where the wizard will import the appropriate user and object information, then click **Next**.

The fields in the lower pane changes based on the selection.

| Field | Description |
|---|---|
| Source | From the drop-down list, choose **Business Intelligence Archive Resource (BIAR) File**. |
| BIAR file | Enter the appropriate file path or click **"..."** to locate the BIAR file on the computer. |

**Step 4** On the Destination environment panel, specify the appropriate information for Business Object Enterprise XI 3.1, then click **Next.**

**Step 5** On the Select objects to import panel, choose the appropriate items to import into your Business Object Enterprise XI 3.1 server, then click **Next.**

**Step 6** On the Import Scenarios panel, select the appropriate radio button for handling scenarios where the objects already exist in the destination environment, then click **Next**.

**Step 7** On the Incremental import panel, uncheck the check boxes next to options that perform the following imports as appropriate, then click **Next.**

- To import report objects without overwriting dependent objects that already exist in the destination environment.
- To import report objects without importing a universe or connection that would overwrite a universe or connection in the destination environment.

**Step 8** Review the summary information on the Import Wizard panel, then click **Next**.

**Step 9** On the Users and groups panel, check the check boxes to import the appropriate specific users and groups, then click **Next**.

**Step 10** On the Custom Access Levels panel, check the appropriate check boxes to import custom access level, then click **Next**.

**Note** If you import a user and an object, the Import Wizard imports the rights of that imported and does not exist in the destination environment, the Import Wizard gives the individual rights specified in the level on the object for the user.

**Step 11** On the Categories panel, choose the appropriate categories to be imported, then click **Next**.

- For large document domains, you can import incrementally and import documents one category at a time.
- To import all the objects associated with the category, check the **Import all objects that belong to the selected categories** check box.

**Step 12** On the Folders and objects panel, select the folders and objects to import, then click **Next**.

- Click **Select All** to check all check boxes of the folders and objects displayed.
- To import all instances of each selected object, check the **Import all instances of each selected object** check box.

**Step 13** On the Select application folders and objects panel, select the application folders and objects to import, then click **Next**.

**Note** If the selected folders and objects exist on the destination system, they will be updated using the source system as a reference.

**Step 14** In the Import options for universes and connections panel:

a. Select *one* of the following import radio buttons for universe objects:
   - Import all universes and connection objects (Default)
   - Import all universes and only connection objects used by these universes
   - Import the universes and connections that the selected Web Intelligence and Desktop Intelligence documents use directly. In the next dialog box, you can select additional universes that are not used by any imported document.

b. To import universe overloads, check the **Keep universe overloads for imported users and groups** check box.

   If you are importing universe overloads, you must also choose the users, groups, and universes during previous steps in the Import Wizard. You must also choose to overwrite existing objects in the Incremental import panel. If you did not choose these options, click **Back** until you get to the appropriate panel.

c. If you choose to import users and groups, universe overloads, and other objects and do not want to overwrite all of the object rights for the users and groups and objects that you've selected, use *one* of the following methods:
   - Ensure that you choose only the universes that you want to import and then import the other objects in another import process.
   - Delete the universe overload in the destination environment and then uncheck the **Overwrite existing objects** and **Overwrite object rights** check boxes in the Incremental import panel. In this case, the Import Wizard imports only the universe overloads that do not already exist in the destination environment.

d. Click **Next**.

e. If you selected either of the first two options, the Import options for publications panel displays. Skip to Step 15 to continue.

f. If you selected the third option, the Universe folder and universes panel displays. Choose the appropriate universes and universe folders.
   - If the selected source environment is other than BusinessObjects 5.x or 6.x, the universes that are linked to specific documents cannot be cleared from the list. You can choose additional universes that are not used by any imported document.
   - If no universe is found, the associated documents will not be imported and a warning message appears. If this occurs, link the documents to a universe, republish the documents to the repository, and retry the import.

g. Click **Next**.

**Step 15** On the Import Options for Universes panel, choose the appropriate profile and publication recipient radio buttons, then click **Next**.

| Option | Description |
|---|---|
| Select import option for importing profiles | • Import all profiles<br>• Import profiles used by selected publications |
| Select import option for importing publication recipients | • Import recipients used by selected publications<br>• Do not import recipients |

**Step 16** On the Remote Connections and Replication Jobs panel, choose the appropriate remote connections and replication jobs to import, then click **Next**.

**Step 17** On the Ready to import panel, click **Finish**.

# Configuring the BusinessObjects Connection for the Process Orchestrator Report Database

BusinessObjects can support a Process Orchestrator Report Database running on either an SQL server or Oracle. Use the following steps to configure the BusinessObjects reporting connection to the Process Orchestrator reporting database.

### Before You Begin

- If you are running on an SQL server:
  - Verify that the Microsoft SQL native client or the Oracle client is installed.
  - On the reporting database, verify that the named pipes are enabled. To enable the named pipes, use MS SQL Configuration Manager.
  - For additional information on the prerequisites, refer to the SQL server information on the Microsoft Developer Network.

To configure the BO reports connection:

**Step 1** Choose **Start > All Programs > BusinessObjects XI.3.1 > BusinessObjects Enterprise > Designer**.

**Step 2** On the User ID dialog box, enter the appropriate credentials, then click **OK**.

**Step 3** On the Universe Designer dialog box, choose **Tools > Connections**.

**Step 4** From the Connections List, highlight **Process OrchestratorReporting**, then click **Edit**.

**Step 5** On the Edit Process Orchestrator Reporting Connection dialog box, in the Authentication Mode list, verify that Use specified username and password is selected.

**Step 6** Specify the reporting database information, then click **Next**.

**Step 7** Verify the information on the Configuration Parameters and the Custom Parameters panels, then click **Finish**.

# Accessing SQL Server Reporting Service Reports

Process Orchestrator provides SQL Server Reporting Services reports required by managers and auditors as well as reports required by developers of process automation. In general, the data that is provided for these diverse audiences is the same. What differs is the granularity and level of aggregation of this data and the specific data that is the focus of a given report.

**Before You Begin**

- Access to any report is controlled and available only to authorized users.

To access reports from your web browser:

**Step 1** In the Address bar of your web browser, enter the following URL:

```
http://<ReportServer>/Reports
```

where <ReportServer> is the name of the server hosting the reports.

If you use SQL server named instances, the URL and Report folder include the instance name. For example, if you enter <SQLserver>\NamedInstance:

- The URL will be `http://<ReportServer>/Reports_NamedInstance`.
- The report folder will be located at `Cisco Process Orchestrator Reporting > <ReportServer> > NamedInstance > <ReportDBName>`.

**Step 2** Click **Cisco Process Orchestrator Reporting - <Report Server> - <Report DB Name>**.

The Process Orchestrator Reporting Database Report Manager home page displays.

**Step 3** On the Process Orchestrator Reporting Database Report Manager home page, click **Core** to view the Core Process Orchestrator report folders.

The available reports and a brief description of the information that is generated by each report is displayed.

**Step 4** Click the report name to enter the search criteria and generate the report.

# Accessing Reports in BusinessObjects InfoView

Process Orchestrator provides reports required by managers and auditors as well as reports required by developers of process automation. In general, the data that is provided for these diverse audiences is the same. What differs is the granularity and level of aggregation of this data and the specific data that is the focus of a given report.

**Before You Begin**

- Access to any report is controlled and available only to authorized users.

To access Process Orchestrator Core reports in Java Infoview:

**Step 1** On the BusinessObjects client machine, choose **Start > All Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise > BusinessObjects Enterprise Java InfoView**.

**Step 2** In the SAP BusinessObjects application, enter your user credentials to log on to InfoView.

**Step 3** Choose **Documents List > Public Folders > Cisco Process Orchestrator > Core > Auditing** (or) **Operations**.

**Step 4** To view a report, double-click on the report name.

## Modifying the Report Database

In some cases, you might need to move the SQL Server Reporting Services server to another location. You must remove the current Report Database connection and connect to the new server using the Report Database Import Report wizard.

To modify the report database:

**Step 1** Choose **Administration > Database Settings**.

**Step 2** Highlight **Report Database**, right-click and choose **Remove Cisco Process Orchestrator Reporting Database Connection**.

**Step 3** To connect to the new server, see Creating a Reporting Database Connection, page 4-7.

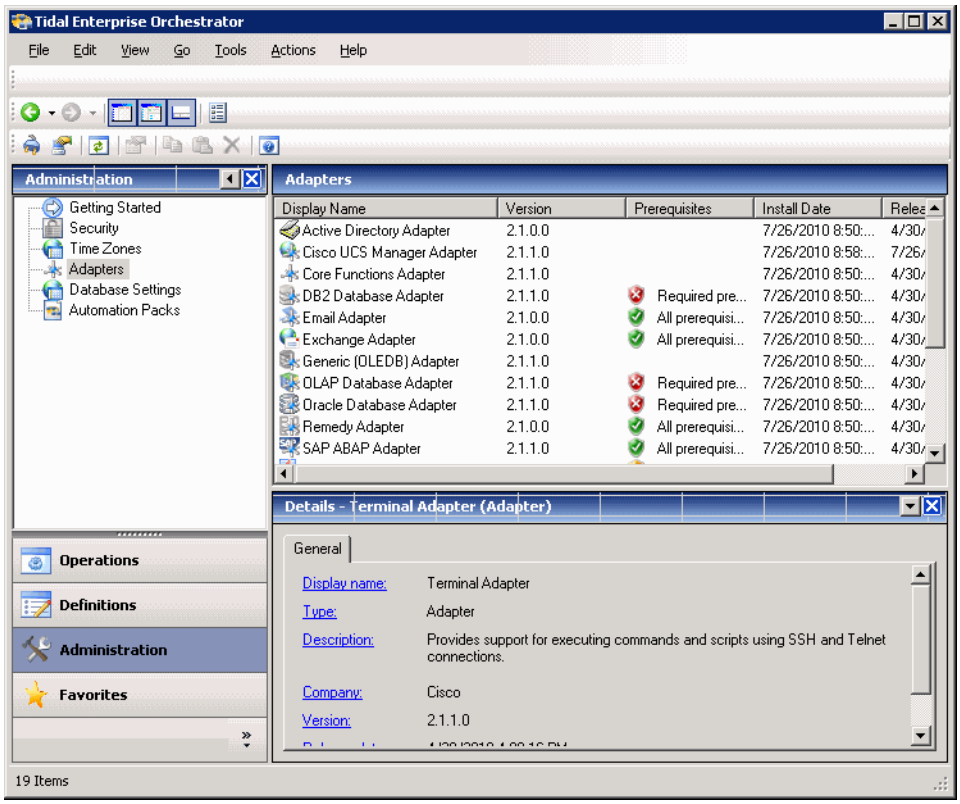
# Configuring Adapters

Adapters are one of the extensibility mechanisms used to extend Process Orchestrator functionality to integrate with devices, environments, applications, or tools without undergoing core modification. Examples of adapters include:

- The primary adapter for Cisco Process Orchestrator is the Core Functions Adapter. This adapter provides the core features and objects to be used to manage IT processes.
- Microsoft Windows Adapter provides Windows objects, such as the Windows computer target, Windows runtime user, and Windows-related activities.

Use the Administration > Adapters view to display the adapters that are installed with the product and their associated objects.

***Figure 4-4 Administration > Adapters***




# Viewing Adapter Prerequisites

Each adapter can have prerequisites that must be satisfied before the adapter can be fully functional. For example, the Core Functions Adapter has no prerequisites, but the SAP ABAP Adapter requires the SAP .NET 3.0 connector; without this connector, the SAP ABAP adapter will not function correctly.

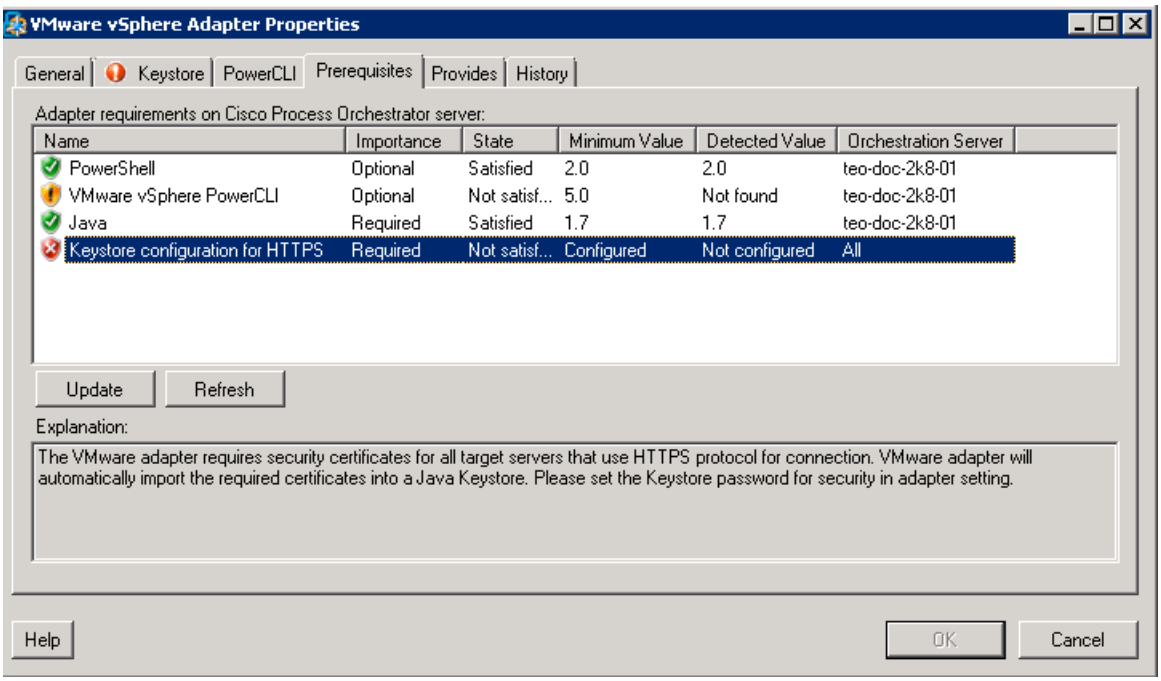
Prerequisites apply to particular Process Orchestrator servers, so prerequisites can be satisfied on one server but not on another. This means that sometimes an adapter might be functional on one server and not on another if the prerequisites are not met on that other server. If you are running in a high availability environment, it is important that all required adapter's prerequisites are satisfied on all Process Orchestrator servers.

To check the status of the prerequisites for each adapter:

**Step 1** Choose **Administration > Adapters**, then double-click an adapter name.

**Step 2** In the [adapter name] Adapter Properties dialog, click the **Prerequisites** tab.

*Figure 4-5 Administration > Adapters > Prerequisites*




**Step 3** If your environment contains more than one server, you can view the prerequisites for each server. Choose **Administration > Orchestration Servers > PO Server properties > Prerequisites**.

### Related Topics

For information about using the features supplied by the Core Functions adapter, see Using Core Activities, page 12-5.

# Viewing Adapter Properties

To view general information related to the adapter, the specific functions that the adapter provides, and a history of changes that have been made to the adapter, choose **Administration > Adapters > Adapter Properties**. The information on the property dialog for each adapter varies.

# Viewing Objects Provided by the Adapter

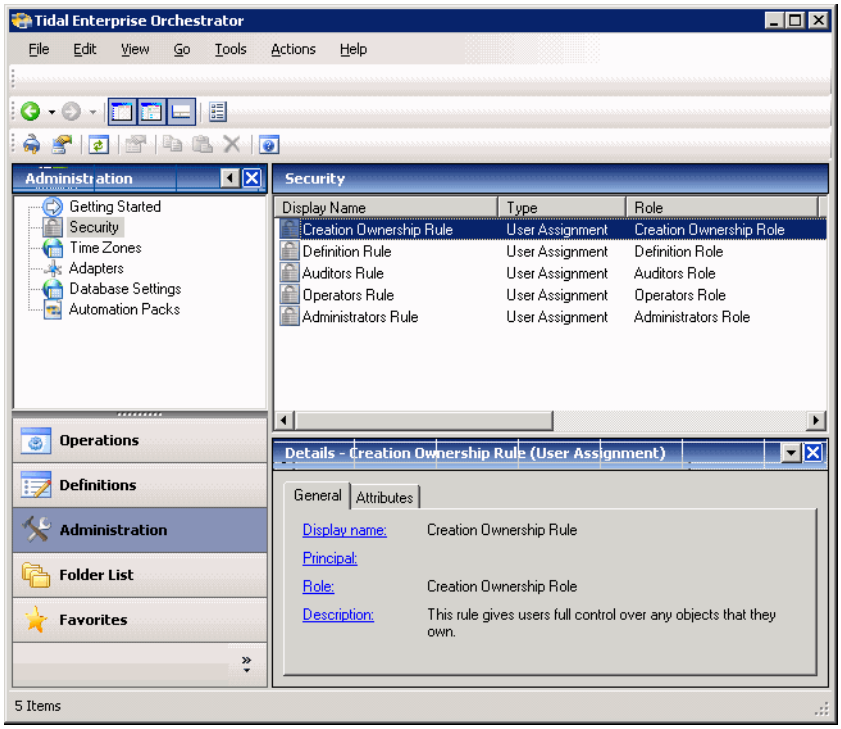
Each adapter provides specific functionality within Process Orchestrator. Use the **Provides** tab in the Administration > Adapters > Adapter Properties dialog to view the functionality that is provided by an adapter.

# Configuring Security

As a Process Orchestrator administrator, you can limit users access to only a subset of Process Orchestrator processes or limit a whole group to read-only access for specific objects. For example, the Process Orchestrator administrator can configure members of the Accounting Process Authors group with the ability to only create, delete, modify or schedule processes in the Accounting category and can only act on targets that represent the accounting services. However, those members of that group will not be allowed to view or modify any other processes or targets in Process Orchestrator.

Using the Security view, you can determine what Process Orchestrator objects users are allowed to view, create, or modify.

***Figure 4-6 Administration > Security View***




If the user has permissions to view (or edit) only a subset of Process Orchestrator objects, when the Process Orchestrator user interfaces (Console, CLI, or Web Service) display Process Orchestrator objects to the user, the display will only show those objects that the user has permissions to view or modify.

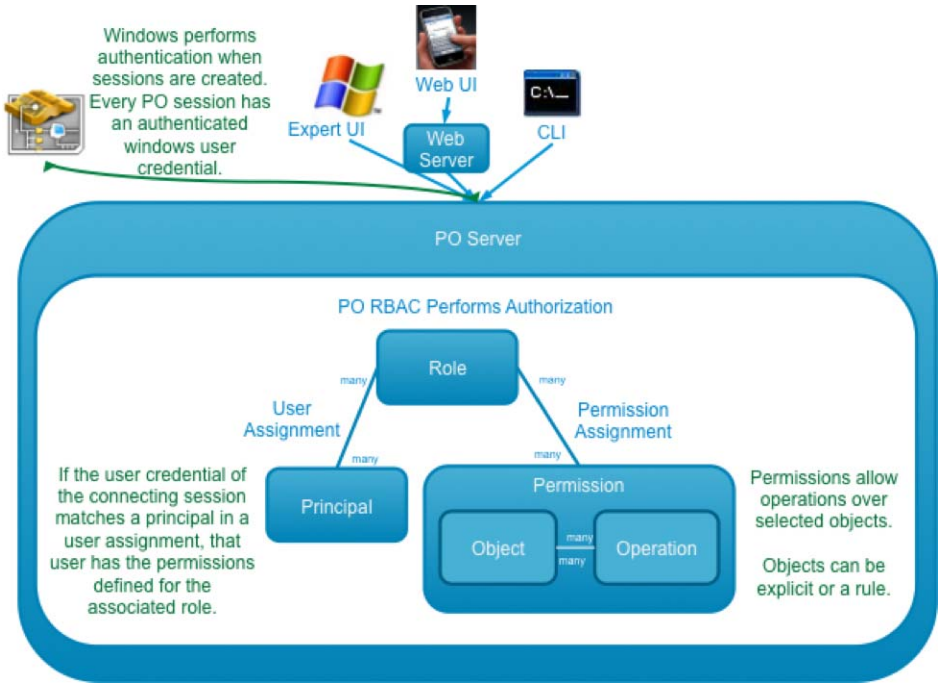
When the user does not have sufficient permissions in Process Orchestrator to perform an operation and attempts to perform that operation, Process Orchestrator will generate an error, as well as log an audit failure event to the event log.

# Role-Based Access Control (RBAC)

In Process Orchestrator, authorization is performed using a Role-Based Access Control System. Roles are a collection of permissions. Each permission pairs a set of operations that can be performed over some set of objects. A user assignment gives end users the ability to perform the role.

Typically, roles are defined according to a standardized job function within IT. Examples might include "Level 1 Helpdesk," "Level 2 Helpdesk," "Human Resources," "Network Configuration," "SAP Basis Expert," and so on. Security groups already in the directory for the users in these job functions are then typically assigned to the roles.

***Figure 4-7 Process Orchestrator Role-Based Access Control***




Permissions define what operations can be performed over what objects. This defines the rights and associates them with set of Process Orchestrator objects. This is similar to file permissions (such as read or update).

- Operations are things such as Cancel, Change Owner, Create, Delete, Read, Start, Update, and Use. Most other Operations automatically allow Read. Change Owner automatically allows Update and Read.
- Objects specify rules that match elements from the functional model, such as Processes, Targets, and Runtime Users. Specifically, several types of object rules are supported:
  - Object List—Allows rights only to specified objects in the list.
  - Object Type—Allows rights to all objects of that type (for example, all targets or all processes).
  - Owner Security—Allows rights to all objects of a specified type that are owned by a specific principal (user or group in Active Directory).

A User Assignment is a link to a security principal. The user assignment is the thing that is defined in Process Orchestrator, since it defines the rule for who is in the Role. The Principal is actually in an external directory, so the user assignment is a reference to the directory.

## Predefined Security Roles

The following roles are defined by default, but custom user roles can be created using the Administration view:

| Role | Description |
|---|---|
| Administrators Role | These users have access to all functionality in the product. Users can view or modify any definition, process or setting. Only a small number of users are assigned this role. These users have permissions to modify adapter settings, import automation packs, and configure databases. |
| Auditors Role | The user can view any definition or instances, such as running processes, but are not allowed to modify any of the objects. |
| Create Ownership Role | All users are assigned to this group. It provides full control over any objects the user owns. |
| Definition Role | This is a user who can define processes. The user can:<br>• View all Operations information (Activity Views, Process Views and Auditing)<br>• Start adhoc processes<br>• Cancel running processes<br>• View/update definitions<br>The user *cannot* update administration settings. |
| Operators Role | This is a classic role for a level 1 Service Desk employee, executing workflows. The user can:<br>• View all Operations information (Activity Views, Process Views and Auditing)<br>• Start ad-hoc processes<br>• Cancel running processes<br>The user *cannot*:<br>• Update definitions<br>• Update administration settings |

## Predefined Security Permissions

In the Security Role Properties property sheet, the Operations tab displays the list of permissions included and/or available for inclusion into the security role. Check the check boxes for the appropriate powers for the security role.

For a list of the predefined security permissions, see the online help.

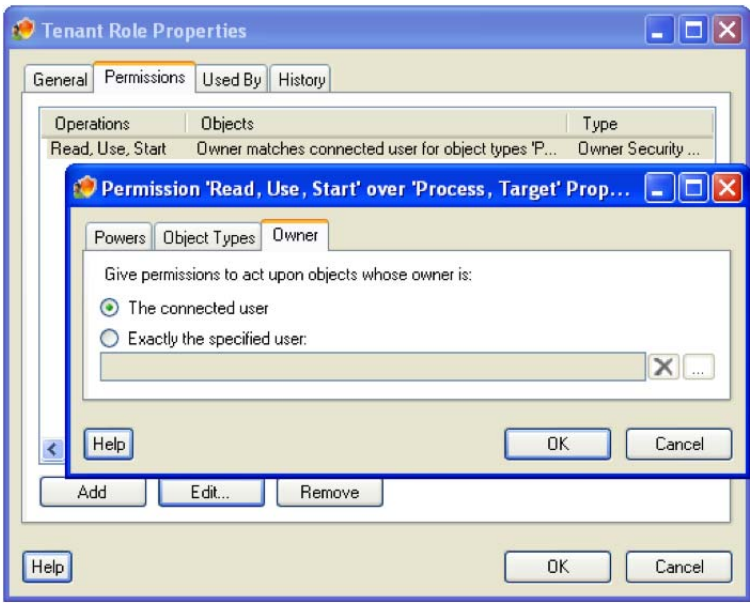
## Object Owners

Process Orchestrator objects have an owner. For example, Process Orchestrator targets, processes, calendar, target group, global variables, etc. all have an owner field. Owners are security principals; by default, they reflect the user who created an object. However, they can be set to any Active Directory security principal. In addition, in the preferences section of the UI, users can specify a security principal such as a group name to use for all objects they create. This concept is useful in team development.

One type of RBAC object definition within an RBAC permission is an Owner security permission. This has two possible settings:

- The *connected user* matches objects the user owns, either directly or indirectly. For example, if the user running the Process Orchestrator console is a member of a group specified in an owner field, the permission would match that object.
- *Exactly the specified user* matches objects where the user is an explicit match.

***Figure 4-8 RBAC Object Specification Using an Object's Owner***




This feature can be used to give a simple yet powerful way to say users can act on objects they create. It can also be used to specify access to objects according to the security principal of the connecting user. If groups are used for owner fields rather than the ID of the user who created the object, users of that group will have access. This is very useful in team development scenarios.

Object views have an optional column that can be used to manage object ownership.

# Creating a New Security Role

A security role is a list of permissions that enhances product use through logical groupings of powers and objects for reuse in multiple places. A security permission is a pairing of a scope that defines the objects and powers over those objects.

After a security role has been defined, it is available in a list of available security roles on the Administration > Security view.

To create a new security role:

**Step 1** Choose **Administration > Security**, right-click and choose **New > Role.**

**Step 2** On the **New Role** property pages, define the properties.

**Note** For information about the field parameters, see the online help.

**Step 3** Click the **Permissions** tab and add permissions as necessary.

**Step 4** Click the **User Assignments** tab and add user assignments as necessary.

**Step 5** Click **OK**. The new security role is displayed on the Administration > Security view.

# Changing the Owner of a Security Role

With the appropriate rights, ownership of a security role can be changed using the Process Orchestrator security feature.

To modify the owner or principal of a security role:

**Step 1** Choose **Administration > Security**, highlight an existing security role, right-click and choose **Properties**.

**Step 2** On the Security Role Properties property sheet, click the **General** tab, scroll to the Owner field, then click **"..."** to the right of each field.

**Step 3** To change the default object type in the Select this object type field:

a. Click **Object Types** on the Select User or Group dialog box.

b. Check the check box to the left of the appropriate object types, then click **OK**.

**Step 4** To change the server location for querying a user or group:

a. Click **Locations** on the Select User or Group dialog box.

b. Expand the appropriate server name, choose the user or group, then click **OK**.

**Step 5** In the Enter the object name to select text field, enter the user name (domain\user name) or group, then click **Check Names**.

**Note** If there is a matching user name or group, it is displayed in the field. If there is not a match, an error message displays.

**Step 6** To enter additional query information:

a. Click **Advanced**.

b. In the Name field, enter the appropriate terms, then click **Find Now**.

**Note** To generate more search results, leave the Name field blank and then click **Find Now**.

**Step 7** In the Search Results pane, choose the appropriate user or group, then click **OK**.

**Step 8** Click **OK**.

# Assigning Users to Security Roles

Use the following steps to quickly assign a user or group to a security role.

To assign a principal to a security role:

**Step 1** Choose **Administration > Security**, highlight the appropriate security role, right-click and choose **Assign**.

**Step 2** On the Select User or Group dialog box, in the Enter the object name to select text field, enter the user name (domain\user name) or group, then click **Check Names**.

**Note** If there is a matching user name or group, it is displayed in the field. If there is not a match, an error message displays.

**Step 3** On the Search Results pane, choose the appropriate user or group, then click **OK**.

# Removing Users from Security Roles

To remove user assignments from a security role:

**Step 1** Choose **Administration > Security**, highlight the appropriate security role, right-click and choose **Properties**.

**Step 2** Click the **User Assignment** tab.

**Step 3** Highlight the principal to be removed and click **Remove**, then click **OK**.

# Modifying Security Permissions

With the appropriate rights, operations, objects, and owners can be added or removed from a security permission. There are three types of security permissions:

- Operations—The Operations tab displays all the operations in Cisco Process Orchestrator, such as Break Lock, Delete, Change Owner, and so on. Use this tab to add or remove an operation from the security permission.
- Object Types—The Object Types tab displays the top-level object types in Cisco Process Orchestrator, such as adapters, processes, targets, and global variables. Use this tab to add or remove an object type from the security permission.
- Owner—Use the Owner tab to modify the Windows security principal (either a user or a group) associated with the object and the permission. The permission is enforced upon the specified owner of the object.

To edit a security permission:

**Step 1** Choose **Administrations > Security.**

**Step 2** Highlight the appropriate security role, right-click and choose **Properties**.

**Step 3** On the [Security Role] Properties property sheet, click the **Permissions** tab, highlight the appropriate security permission, then click **Edit**.

**Step 4** Click the **Operations**, **Object Types**, or **Owner** tab, make the necessary changes, then click **OK.**

**Note** Removing an operation or object from the list is permanent. Click **Add** to add a new operation or object to the security permission.

# Deleting a Security Role

Deleting the role deletes all associated permissions.

**Note** Removing the security role from the list is permanent. To add a new security role, see Creating a New Security Role, page 4-19.

To delete a security role:

**Step 1** Choose **Administrations > Security.**

**Step 2** Highlight the appropriate security role, right-click and choose **Delete**.

**Note** The default security role cannot be deleted.

**Step 3** On the Confirm Security Role Delete dialog box, click **Yes** to delete the security role.


# Managing High Availability and Resiliency

This topic documents the basic approach used to support operational continuity, the ability to maintain availability during routine maintenance, and disaster recovery requirements for a Cisco Process Orchestrator. Using the production-level planning and operations described in the following topics, implementing these features should result in as much as 99.95% up time for your site:



**Related Topics**



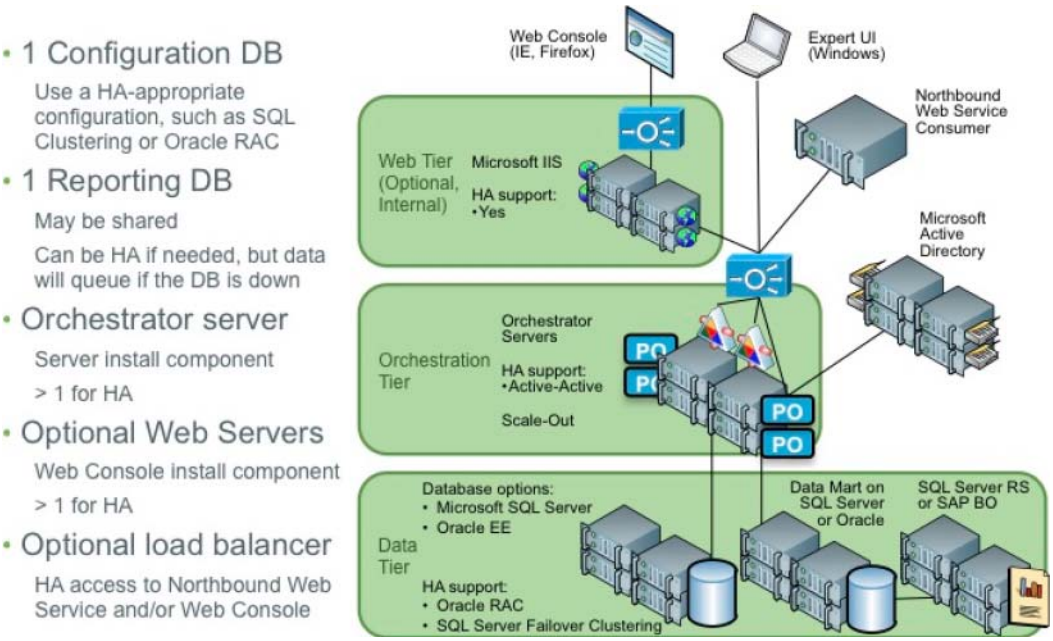
## Understanding High Availability and Resiliency

The Cisco Process Orchestrator active-active server provides a highly available and scalable solution, the goal of which is to virtually eliminate down time due to hardware application failures. This solution, which protects critical pieces of the system from failure and excessive loads, includes these features:

- Rather than a single server performing all of the work (running processes, monitoring triggers, and so on), a Process Orchestrator environment can contain many servers. If one server fails, as long as another is available to take its place, useful work can continue.
- In a Process Orchestrator environment, all servers connected to the system share the workload, which helps increase scalability and performance. The more servers that are available in the system, the less work each of them must perform.
- A Process Orchestrator environment is available during routine maintenance, such as rebooting the operating system, applying OS security patches, and performing minor upgrades (even to the Process Orchestrator itself).

The following figure illustrates the recommended Process Orchestrator high availability environment:

***Figure 1 High Availability Environment***




The Process Orchestrator environment consists of:

- A reporting database. Although the environment requires a reporting database to enable reporting features, the reporting database can span environments and therefore can be thought of as external to the environment.
- A configuration database. The configuration database is specific to *one environment*, and therefore requires a high availability-appropriate solution.
- Multiple Process Orchestrator servers
- Optional: Multiple web servers
- A load balancer in front of the northbound web service and/or web servers (optional). For information about installing a load balancer, see the *Cisco Process Orchestrator Installation Guide*.

Because the Process Orchestrator environment supports high availability, the installation process:

- Requires an environment name (defaults to PRODUCTION).
- Differs for "subsequent" servers that are added to an existing high availability environment. During the installation of a secondary server, the installation process does *not* create the databases, configure security, import automation packs, or configure Windows accounts.

  However, the secondary server *does* require connectivity to an existing server in the environment. In addition, a security check is performed to ensure that the person installing the new server has the permissions to:

  - Add the server to an existing environment
  - Access the processes database
- Configures a default Process Orchestrator Windows Runtime User for Orchestration Server targets.

- Recommended: Assign security principals to default roles (typically Active Directory groups). The install process prompts you to specify the groups to be assigned to default security roles, but allows this step to be skipped. However, you will need to do this eventually or the environment will only work for the account that installed it.

**Note** Do not use local security groups and users in a high availability environment. By their nature, local security groups and users are accessible to one Process Orchestrator server, but are not accessible to another server. This means that they will be very confusing to configure and work with because server A will have a different "view" of permissions than server B.

In addition, because the Process Orchestrator environment supports high availability:

- The license key is tied to a computer name. This association is checked only when the license is applied. Therefore, if you have a high availability environment with three servers, for example, you can obtain a license key tied to one of the servers and then apply that license using the UI connected to that server. After the license has been applied, the license loses its association with the first server's name and becomes available to all servers in the environment.
- Process execution and event monitoring are automatically "equally" divided between all Process Orchestrator servers, with the following exceptions:
  - Only the *current* load is taken into account.
  - CPU, disk space, and other performance factors are *not* taken into account.
  - Due to the technical constraints of adapter connections, some work must be performed on a specific node.
- There are several constructs in Process Orchestrator that relate to files that exist on the servers. These constructs must now store files on some network share, which is available to all servers, so that if one server writes the file and fails, or if work is shifted for load balancing, other servers can access it to continue the automation. Ideally the network share would not be a directory on some specific Windows server, but a highly available location with redundancy and fault tolerance, such as an HA NAS system. Examples of such elements that should now use a share in a multi-server Process Orchestrator environment are:
  - Automation summaries, which are stored as XML files. The server must log in to write automation summaries from multiple servers.
  - Files that are written by some process and later read by the same or another process such as email attachments or FTP files. Although Process Orchestrator provides backward compatibility so that processes that rely on the well-known Windows Computer target can run after the upgrade, processes that rely on file locations *local* to a Process Orchestrator server will not be high availability-ready.
- Windows Computer targets are created for each Process Orchestrator server. However, the singleton (well-known) Windows Computer target from prior Process Orchestrator versions still exists and represents the first server in the environment. Processes that rely on the well-known Windows computer target will not be high-availability ready.
- The Core Automation Pack contains a well-known target group that includes all Process Orchestration server targets.

**Note** When writing high-availability processes, use this group instead of the well-known Windows computer target ID.

# Advance Planning

Ensuring both operational continuity and disaster recovery preparedness requires:

1. Creating a support team (see Creating the Cisco Process Orchestrator Support Team, page 5-4).
2. Planning for high availability by installing multiple Process Orchestrators on virtual machines (see Planning for High Availability, page 5-5).
3. Using one of the recommended databases (see Storing the Information, page 5-6).
4. Considering the dependencies Process Orchestrator has on other services (see Considering Dependencies on Other Services, page 5-6).
5. Develop a strategy for performing regular database and system backups.

## Creating the Cisco Process Orchestrator Support Team

To maintain operational continuity and ensure that the people on your site are prepared for any emergency:

**Step 1** Create your Cisco Process Orchestrator Support team *well in advance of any disaster.*

When a disaster occurs, the first step of any recovery process should be to contact this team. The members of this team should know the prescribed (and latest) list of procedures, and each member should have a pre-defined role in the recovery process.

This team should consist of:

- The team assigned to day-to-day administration and operations.

**Note** Although Process Orchestrator can be useful across a broad span of IT users, it needs an administrator/operator. This person will use the Operations Workspace (see Monitoring Processes, page 5-9) to monitor the health of processes and receive operational alerts, and will typically serve as the primary contact for customer support issues.

- The database administrator (DBA) responsible for the database if the disaster is related to the Process Orchestrator database.
- The manager of the VM farm on which the Process Orchestrator servers run.
- Personnel who can run the software that restores the Process Orchestrator server. This can be the VM farm operator.

**Step 2** Because outages in individual upstream systems can affect processes that span technologies and services, maintain the contact information for the owners of any upstream systems (see Considering Dependencies on Other Services, page 5-6). This information should include current names and phone numbers.

# Planning for High Availability

A single-server environment does not provide high availability, so you should plan to implement multiple Process Orchestrator active-active servers. When you are planning your Process Orchestrator environment, consider these recommendations:

- Be sure that all Process Orchestrator servers have similar hardware and software configurations; that is, the same (or similar) amounts of memory, number of CPUs, OS levels, and so on.

  Cisco Process Orchestrator's high availability solution does not take hardware or operating system differences into account when distributing load. When new work arrives to be executed by Cisco Process Orchestrator, only the current load (the number of process instances and monitored triggers) is considered. Over time, more work will be assigned to less loaded or more capable Process Orchestrator servers, but the amount of memory, CPUs, or storage is not directly taken into account when assigning work.
- Install the Process Orchestrator servers on virtual machines backed by networked storage. Using this approach, if a host fails, the VM can be migrated to a new host using a tool such as vCenter. Many customers find this method of fault tolerance preferable because the host can change to another site in disaster recovery situations.
- VM farms tend to be sets of machines connected to the same storage, and might be in the same data center. Consider replicating your snapshots to a secondary data center.

### Moving to High Availability

Customers running multiple database and Process Orchestrator 2.x server installations running the *same content* against many targets should consider going to a single database and a multiple high availability server installation. To migrate to this environment:

**Step 1** Upgrade one of the servers directly.

**Step 2** Add additional high availability servers (on separate hardware, unrelated to the old 2.x servers).

**Step 3** Create automation packs to include your Targets from the non-upgraded 2.x environments.

**Step 4** One by one, shutdown your old 2.x servers and at the same time, import the automation packs with targets from those servers into the new high availability environment.

Customers running multiple database and Process Orchestrator 2.x server installations running *different* content against many targets might not want to use high availability, but it can be accomplished. In this case, to migrate to this environment:

**Step 1** Perform same steps listed above.

**Step 2** Export whatever content you need from each 2.x server (in addition to the targets), then import the content back into the new high availability environment.

# Storing the Information

Virtually all Process Orchestrator state information is stored in the database. Although there are two databases, only the configuration database is a hard upstream dependency (see Considering Dependencies on Other Services, page 5-6) and must be made highly available to prevent a database-based Process Orchestrator outage. The reporting database is used for auditing and reporting purposes, but Process Orchestrator does not depend on its presence or on it being up and running.

Process Orchestrator supports the following high availability databases:

- SQL Server Failover Cluster
- Oracle Real Application Clusters (see the Oracle RAC configuration steps in the *Cisco Process Orchestrator Installation Guide*)

Although the bulk of the Process Orchestrator server's state is stored in the database, the following important pieces of data are stored in the Process Orchestrator server system:

***Table 5-1 Information Held in the Process Orchestrator Server System***

| Type of Information | Description |
|---|---|
| The keys used to encrypt security credentials | This key is stored in a controlled space in the Windows security subsystem of the Windows server hosting the Process Orchestrator server.<br><br>Process Orchestrator stores runtime user names, passwords, and other credentials in the database so that it can connect using those credentials when running an activity against a target. Credentials are encrypted with a key specific to the Process Orchestrator environment before being stored in the database.<br><br>This environment-specific key, which is stored in the Windows security system of each Process Orchestrator server, can be exported as a file, retained for disaster recovery purposes, and moved to a standby/backup server for recovery (see Saving the Process Orchestrator Server Data in Preparation for a Disaster, page 5-8). In a disaster recovery situation, the Process Orchestrator database is useless without the corresponding and separately stored encryption keys. |
| The Process Orchestrator server installation | This is core software that can be recreated from the installation media. |
| The server configuration file | This file configures certain properties of the Process Orchestrator server. For example, it tells the Process Orchestrator server which ports to open to talk to the client. It also tells the server where the database is and which credentials to use to access it. |
| Persistent queue files for data to be written to the reporting database | These files store reporting data so that it is not lost across server restarts (see Handling Restarts and Failures, page 5-16). |

# Considering Dependencies on Other Services

Process Orchestrator interacts with many different IT technologies and processes (see Considering Dependencies on Other Services, page 5-6). It is critically dependent on some technologies, such as its backend database on an Oracle or SQL Server (a hard upstream dependency), and on other technologies such as a DB2 database or VMware vCenter connection (soft upstream dependencies), only to the extent that processes interact with those technologies.

Similarly, other IT services might rely on Process Orchestrator. For example, an IaaS private cloud service built on Cisco Intelligent Automation for Cloud (that includes Process Orchestrator) will be non-operational or will have to revert to all-manual workarounds without a functioning Process Orchestrator.

Other IT services could operate in various degraded states without Process Orchestrator. For example, Incident Management in Remedy will still function as IT's Incident Management without a functional Process Orchestrator, but capabilities such as automated troubleshooting (a hard downstream dependency) and ticket enrichment (a soft downstream dependency) will not be available.

To summarize, when you are planning your Process Orchestrator environment, take into consideration all of these types of dependencies:

***Table 5-2 Dependency Types***

| Dependency | Description |
|---|---|
| Hard | A component or service is required by the dependent service to be fully operational. |
| Soft | A component or service is not required by the dependent service to be fully operational, but functionality directly involving the component might be degraded or broken. |
| Upstream | The IT services that Process Orchestrator relies on. |
| Downstream | The IT services that rely on Process Orchestrator. |

For more information about the dependencies on Process Orchestrator and Process Orchestrator's dependencies on other services, see the *Cisco Process Orchestrator Compatibility Matrix*.

# Ensuring Operational Continuity

Ensuring operational continuity of your Process Orchestrator environment includes these actions:

- Backing Up Your Data, page 5-7
- Monitoring Processes and Events, page 5-8
- Handling Errors, Exceptions, and Diagnostics, page 5-9

## Backing Up Your Data

In a disaster recovery situation, you will need copies of the following information, so be sure to regularly:

- Back up your database. Follow the standard database backup procedures for data recovery and the best practice database resiliency strategies of your database vendors, which should be provided in the documentation for the applicable database platform.
- Save the Windows security credentials encryption key for the Process Orchestrator environment. This only needs to be done once; this key is shared by all servers in a single Process Orchestrator environment. For information about how to save the key, see Saving the Process Orchestrator Server Data in Preparation for a Disaster, page 5-8.
- Save the Process Orchestrator server configuration file. For more information about this file, see Storing the Information, page 5-6.

- Take snapshots of the Process Orchestrator server VMs and save them at an alternate site (see Planning for High Availability, page 5-5).
- Save any customizations to the Web.config file in the Web Console server.

## Saving the Process Orchestrator Server Data in Preparation for a Disaster

When making preparations for a disaster, the most important piece of information to save is the security key that a Process Orchestrator server uses to encrypt sensitive data before storing it to the database. This environment-specific key is stored in the Windows security system of each Process Orchestrator server.

**Note** In a disaster recovery situation, the Process Orchestrator database is useless without the corresponding and separately stored encryption key for the Process Orchestrator environment.

Another important piece of data is the Process Orchestrator server configuration file. For more information about this file, see Storing the Information, page 5-6.

To save this information:

**Step 1** Once for the Process Orchestrator environment:

**a.** Export the security credentials encryption key to a file:

```
aspnet_regiis -px "Tidal Intelligent Automation Server" Keys.xml -pri
```

Aspnet_regiis.exe is a Microsoft.NET framework utility that resides in the .Net Framework folder. This folder is typically located at this location:

```
C:\Windows\Microsoft.Net\Framework64\{Version}\aspnet_regiis.exe
```

You might need to specify the full path to this utility if you have not defined the path in your Path environment variable

**b.** Save the Keys.xml file, then move it to a secure location (typically a standby or backup server) separate from the Process Orchestrator server.

**Step 2** For each Process Orchestrator server, copy the server configuration file to a standby or backup server. You can find this file in the installation folder.

```
%INSTALLDIR%\Tidal.Automation.Server.exe.config
```

# Monitoring Processes and Events

Each server in a Process Orchestrator environment publishes performance and events. A Process Orchestrator server can also provide data about its own health to operational support systems, and can interact with operational support systems with regard to the health of services for which it provides automation.

If you want to monitor the entire high availability environment, however, you must monitor *all* of the servers. Most application management systems include the ability to monitor the data that Process Orchestrator publishes, such as Windows event logs and performance counters.

Process Orchestrator can integrate with IT management or network management tools such as the Cisco Prime products. Process Orchestrator also provides direct out-of-the box integrations with other management tools, including:

- Microsoft System Center Operations Manager
- SAP CCMS and Solution Manager
- Remedy Service Desk

Process Orchestrator also provides numerous features to integrate in a generic manner with other management systems. For example, the triggered Process Orchestrator process can invoke a web service or run a script and pass in the incident, so the information can be published to another service assurance system.

## Monitoring Processes

Process Orchestrator provides full visibility into the processes running on the system, including the status of all running processes, which activities have executed, succeeded, failed, and so on.

Use the Operations view to monitor the processes that are scheduled to execute, view processes that are currently running, and verify that processes have successfully completed. You can also start processes or interact with human steps in processes called Tasks. For more information about monitoring processes, see Chapter 3, "Monitoring Operations."

## Managing Events

Process Orchestrator provides IT process records such as alerts and incidents, and uses these records to:

- Supply incidents and events for external tools it might be monitoring or managing.
- Manage the Process Orchestrator itself. For example, the Process Orchestrator Core automation pack performs some self-monitoring of the Process Orchestrator, and can raise an alert or incident if there is something that requires the Process Orchestrator administrator's attention.

Use the **Operations > Auditing > System** view to review the list of internal Process Orchestrator events, including notifications of warnings and errors, and information events about the general functioning of the system.

**Note** Events from all Process Orchestrator servers in a high availability environment appear in this view.

For more information about managing events, see Monitoring Auditing Information, page 3-16.

# Handling Errors, Exceptions, and Diagnostics

Use Windows event logs and Windows performance counters to monitor the Process Orchestrator platform, either of which can be monitored through Microsoft Windows directly or fed into the service assurance platform of choice. Because event logs and performance counters are the standard for monitoring Windows-based applications, most service assurance tools include the ability to monitor these elements.

- System event logs

The Process Orchestrator server logs errors that can be useful when you need to diagnose failures to the Windows Event Logs. For example, if there is an error connecting to a database, this will be logged as an event.

These logs can also be picked up by most enterprise systems management tools if you want a view of Process Orchestrator health within your systems or application management console.

For more information about viewing system events, see Monitoring Auditing Information, page 3-16.

- Windows performance counters

  The Process Orchestrator server publishes a number of Windows performance counters concerning its operations. For a list of Windows performance counters, see Windows Performance Counters, page 5-19.

**Note** Be sure to monitor the system event logs and Windows performance counters from *all* Process Orchestrator servers in an environment. Monitoring one server is not sufficient to monitor the health of all servers.

# Maintaining Availability During Routine Maintenance

This section discusses how to maintain the availability of the servers in the Process Orchestrator environment during routine maintenance:



## Handling Routine Maintenance Outages

To maintain availability of the overall high availability environment during routine outages, such as Microsoft security patches and minor Process Orchestrator upgrades that do not affect the database schema:

**Step 1** Shut down one Process Orchestrator service (while keeping the others running). Wait for the Process Orchestrator service to completely stop via a friendly shutdown.

**Step 2** Install the Microsoft patches or Process Orchestrator patches on that server.

**Step 3** Reboot as necessary, then restart the Process Orchestrator service.

**Step 4** Repeat steps 1-3 on all Process Orchestrator servers, one at a time, to maintain availability of the overall high availability environment.

For more information about shutting down and restarting Process Orchestrator, see the *Cisco Process Orchestrator Installation Guide*.

# Adding More Capacity

To add more capacity to your Process Orchestrator environment, just add another Process Orchestrator server. Before you can do that, you must:

- Be able to connect to an existing Process Orchestrator server
- Have permission to add a new server
- Acquire the credentials for connecting to the Process Orchestrator process database

For information about how to add a new server, see the *Cisco Process Orchestrator Installation Guide*.

# Relocating a Process Orchestrator Server

To relocate a Process Orchestrator server:

**Step 1** Add a new server to the environment (see Adding More Capacity, page 5-11).

**Step 2** Uninstall Process Orchestrator from the old server (see the *Cisco Process Orchestrator Installation Guide)*.

# Maintaining the Database

The following sections describe how to maintain the Process Orchestrator database:



## Grooming the Database

Process Orchestrator provides settings to control grooming of the following types of objects:

- Various sections of its Process and Reporting databases.
  - The Process database instances are primarily useful for viewing prior instances in the main expert UI, understanding the specific activities which were executed, querying the activity instance inputs, outputs, and other execution details typically useful in troubleshooting and development scenarios.

    Grooming the Process database can help optimize performance. Reducing the database size can improve database insert speeds, but at a cost of being able to view older process instances in the UI. If you perform complex views including large numbers of historic processes, the database and UI must deal with the larger data volumes. Larger views might also mean larger data payloads coming through the server to the UI. This factor therefore not only affects the database layer, but also the server and UI performance.

    Grooming the Process database as tightly as business scenarios allow optimizes performance of both the Process Orchestrator Server and UIs. For example, if no business requirement exists to monitor and troubleshoot failed processes past the end of a shift, setting grooming for one day might be appropriate.

  - Data about completed process instances is available on a long-term basis in the Process Orchestrator Reporting database. The Reporting database provides information on which processes have run, when they started and ended, whether they were successful or failed, how they were started, and who started the processes in long term storage. The Reporting database has information about processes only, not activities within the process.
- Task instances, such as alerts, incidents, and change requests. These objects are groomed *only* upon their completion, which means that an open or active task can remain in the database forever. By default, the tasks can stay in the database for a long time.
- Audit data, which is groomed less aggressively because it is more important to keep around longer.

Using the default settings, grooming is automatic, but you can tune these settings to your needs. For example, there are options to:

- Control how much data is retained for completed instances.
- Mark expired tasks as completed.
- Configure a process to archive on failure. This allows low data and performance load during normal/successful executions of the process, while archiving failures so that they can be debugged and diagnosed.
- Start grooming immediately (rather than to wait for the scheduled time).

To change the grooming settings for the Process database, see Managing the Process Database, page 4-6.

## Database Performance Best Practices

Proper database server hardware and routine database maintenance can have substantial effects on performance. Although the Process Orchestrator ships with performance-optimized schemas, including the relevant indices, you must install and operate these databases.

Ideally, to optimize performance, a database administrator (DBA) familiar with best practices should prescribe and configure the server hosting the database platform containing the Process Orchestrator databases. The DBA should also be involved in the installation of the database and should perform routine maintenance.

In high performance scenarios, the following best practices can dramatically affect performance:

- Run regular backups of the Process Orchestrator database. Without backups, the transactional logs of the database will grow fairly substantially, resulting in large files and poor database performance, and ultimately affecting Process Orchestrator's performance.
- Always run the database server on physical hardware, not on a VM.
- Use a separate host server for the database instead of running the database server on a Process Orchestrator server.
- Disk throughput (I/O) is also an important metric for large scale deployments. Use a separate high speed disk for the database, operating system, program files, and swap files.
- Provide sufficient memory to avoid paging. In very large installations, Process Orchestrator has benefited from databases running 16 or even 32GB of RAM.
- Use a high-speed network connection. Typically this means the database is "close" to the Process Orchestrator server, and certainly in same data center.

For best practices, refer to the documentation associated with your chosen database platform.

# Performing Disaster Recovery

This section discusses the following topics:



## Recovering from a Server Failure

In a multi-server high availability environment, there is very little you need to do to recover from a single server failure. When a Process Orchestrator server fails (due to, for example, a network outage, disk failure, or software failure), the remaining Process Orchestrator servers within the environment will:

1. Recognize the failure (within a few seconds).
2. Report the failure in event logs.
3. Redistribute the work that was performed by the failed Process Orchestrator server among the remaining healthy servers.

If the server failure is recoverable, simply bringing the server or service back up will put it back into the high availability environment and operations will resume.

If the server failure is not recoverable (for example, a hardware failure), you can add a new Process Orchestrator server to the environment. There is very little need to recover the failed server; it will not be considered healthy, and no work will be assigned to it. To delete the old server from the environment, choose **Administration > Orchestration Servers > Remove**.

## Recovering from a Reporting Database Failure

If only your Reporting database has failed, there are several options:

- If possible, recover the database server. Operations will resume as before, and no changes are needed.
- If the database server is not recoverable but the database is (because there is a backup), restore the database to a new database server (see Restoring the Database to a New Database Server, page 5-13).
- If both the database server and the database are not recoverable, recreate the database on a new database server (see Recreating a New Database on a New Database Server, page 5-14).

### Restoring the Database to a New Database Server

To restore the database to a new database server:

**Step 1** Open a console connecting to *any* Process Orchestrator server in the environment.

**Step 2** Remove the existing Reporting Database connection and connect to the new database server (see Modifying the Report Database, page 4-13).

The changes will propagate throughout the entire high availability environment.

## Recreating a New Database on a New Database Server

To create a new database on a new database server:

**Step 1** Open a console connecting to *any* Process Orchestrator server in the environment.

**Step 2** Choose **Administration > Database Settings**, right-click **Report Database** and choose **Remove Cisco Process Orchestrator Reporting Database Connection**.

**Step 3** Create a new Reporting database on the new database server (see Creating a Reporting Database Connection, page 4-7).

The changes will propagate throughout the entire high availability environment.

# Recovering from a Total Database Failure

If your entire database has failed:

**Step 1** Contact your Cisco Process Orchestrator Support team (see Creating the Cisco Process Orchestrator Support Team, page 5-4). To ensure that the latest procedures are followed and to minimize downtime, the support team should be contacted as early as possible and involved throughout the process. Do not wait to call them until the disaster recovery process goes astray.

**Step 2** Stop the Process Orchestrator service on all of the server machines.

**Step 3** Restore the database from the latest backup to a new database server (or new database schema). The restore will depend on the database version; follow the manufacturer's instructions for your database server.

**Note** If you do *not* have a backup, you *cannot* recover the database. In that case, you must reinstall the entire environment.

**Step 4** Log in to the first Process Orchestrator server, then use the Database User utility (Tidal.Automation.Server.DatabaseUserConfigurationUtility.exe) to point the server to the new database and/or new credentials.

**Step 5** Start the Process Orchestrator service on this server.

**Step 6** Launch the Process Orchestrator console, then verify and correct Reporting database configuration if needed. This step is required if the Reporting database connection or the reporting credentials have changed.

**Step 7** Log in, reconfigure, and restart (Step 4 and Step 5) all other Process Orchestrator servers in the environment.

# Recovering from an Entire Environment Failure

If your environment fails but you have snapshots of the Process Orchestrator server VMs saved at an alternate site, use the procedure described in Recovering from a Total Database Failure, page 5-14 instead.

Follow these steps when your environment fails and you *do not* have snapshots of the Process Orchestrator server VMs saved at an alternate site:

**Step 1** Contact your Cisco Process Orchestrator Support team (see Creating the Cisco Process Orchestrator Support Team, page 5-4). To ensure that the latest procedures are followed and to minimize downtime, contact the support team as early as possible and involve them throughout the process. Do not wait to call them until the disaster recovery process goes astray.

**Step 2** Restore the database from the latest backup to a new database server (or new database schema). The restore will depend on the database version; follow the manufacturer's instructions for your database server.

**Note** If you do not have a backup, you *cannot* recover the database. In that case, you must reinstall the entire environment.

**Step 3** To import the Windows security credentials encryption key from the Keys.xml file created in Saving the Process Orchestrator Server Data in Preparation for a Disaster, page 5-8 to the new (first) Process Orchestrator server in the environment, enter the following command (*do not cut and paste*):

```
aspnet_regiis -pi "Tidal Intelligent Automation Server" Keys.xml -exp
```

Aspnet_regiis.exe is a Microsoft.NET framework utility that resides in the .Net Framework folder. For example, this folder is typically located at this location:

```
C:\Windows\Microsoft.Net\Framework64\{Version}\aspnet_regiis.exe
```

You might need to specify the full path to this utility if you have not defined the path in your Path environment variable.

**Note** It is not enough to just bring the Process Orchestrator environment up; you also need the security credentials encryption key to apply to the server at the new site to make this database copy usable. If you do not have a backup of this key, you *cannot* recover the database. In that case, you must reinstall the entire environment.

**Step 4** Reinstall the core Process Orchestrator software from the installation media. Be sure to reinstall the *same version* and all of the updates/hotfixes.

**Step 5** Replace the server configuration file on the new machine with the file copied in Saving the Process Orchestrator Server Data in Preparation for a Disaster, page 5-8.

**Note** If the file contents contain any path (such as the log file path), make sure that the path is still valid in the new server machine.

**Step 6** Use the Database User utility (Tidal.Automation.Server.DatabaseUserConfigurationUtility.exe) to point the server to the restored database and/or credentials.

**Step 7** Start the Process Orchestrator service on this server.

**Step 8** Launch the Process Orchestrator console, then verify and correct the Reporting database configuration if needed. This step is required if the Reporting database connection or the Reporting credentials have changed.

**Step 9** Install any additional Process Orchestrator servers as *new* High Availability servers (see the *Cisco Process Orchestrator Installation Guide*). The additional servers will receive the database settings and encryption keys from the first server.

**Step 10** Because the Process Orchestrator server is now installed on a different server:

a. Review the list of Windows targets in the Process Orchestrator, remove the Windows computer targets that point to "old" Process Orchestrator servers that are no longer available, and disable the old server target in the target definition.

b. If the Web Server is on a different computer or virtual machine from the Process Orchestrator server computer, and if the Web Server survived the outage but the Process Orchestrator servers did not, choose **File > Environment Properties** to verify and update the Web Console configuration file location.

c. Use the Core Functions Adapter properties to review and update the Automation Summary configuration.

**Step 11** Choose **Administration > Orchestration Servers** and remove the old servers.

# Handling Restarts and Failures

This section discusses the following topics:



## Handling Running Processes During Restarts and Failures

The rules described in this section apply to moving a process from one Process Orchestrator server to another server when a single server fails. If server A goes down, then if the process hasn't stopped on a non-restartable activity, it will move to another active server.

By default, Process Orchestrator persists all data about running processes so that state is preserved across service restarts; whether the restarts are planned or unexpected, process and activity states are persisted to the Process Orchestrator database. When the Process Orchestrator server shuts down in a friendly manner, it allows in-flight activities time to complete, but blocks new activities from running. This typically allows activities that might not be marked as restartable to complete so that processes do not fail during friendly shutdowns.

Generally, processes will pick up and start running again after a restart or failure; in some activities, this is configurable by the user. Many activities, if they were running when the server shut down, can be restarted without any side effects. Some examples:

- If an activity is passive (such a configuration query), Process Orchestrator can simply run the query again.

- Other activities can have side effects and cannot be rerun without consequence. For example, if a CLI adds some entry for a device to a file, running the CLI again will cause two entries for the device. In these cases, Process Orchestrator shows these processes as failed after the restart; the operator must examine the failures, then rerun the failed processes.
- The Web HTTP Request has a “Restart when interrupted” checkbox. Click this option if the HTTP Request is performing a read operation, in which case it should be restartable, but do not click it when performing a write operation because you could accidentally create multiple entries.

Process Orchestrator offers process Start Points that process authors can use to give operators restart points when a process needs to be restarted. If a volatile activity is run in Process Orchestrator and has not returned a status and Process Orchestrator shuts down, when Process Orchestrator restarts, this activity will not resume and the results of what the activity was supposed to do are lost.

Adapters determine whether to declare activities as restartable. The Process Orchestrator server also allows activities to save their state as they run, which allows long-running activities to be restarted. For example, a Create Approval activity might save a state that the approval request task has been created but the activity is blocked waiting on an approval.

Process Orchestrator's writing of state is transactional. Process Orchestrator will not proceed to the next step until the state change is committed to the database. There should be no time gap where process state can be lost. This persistence has a cost. With every step in every process, Process Orchestrator writes to the database to update states such as when the activity starts, stops, etc. Where this performance hit is inappropriate, Process Orchestrator provides a setting to turn restartability off for a process. Many processes do not need to restart after a shutdown or failure; it might often be acceptable to rerun the process at the next scheduled interval or on the next instance of the trigger. For example, many SAP diagnostic processes fall into this category. In the case of SAP diagnostics, Process Orchestrator's SAP automation pack process definitions have restartability turned off on analysis processes. If the server fails, the state of the SAP server with respect to alerts might not be the same after restart anyway. It is often best to not restart these processes but to instead to relearn the state of the SAP system. Also, by disabling restartability, the performance impact of maintaining state is reduced.

# Handling Reporting Data Queues During Restarts and Failures

To improve performance, reporting data is held in queues and is occasionally written to the database in bulk rather than an item at a time. During a friendly shutdown, the queue is flushed so that the reporting data is fully synchronized to the reporting database. In the event of a failure, the data in the queue will be present on restart and will be sent whenever the Process Orchestrator Server starts up; no data will be lost. However, if an unexpected failure occurs for which there is a need to reinstall the Process Orchestrator server on a different computer or restore the computer hosting the Process Orchestrator server from a backup, reporting data from these queues is lost.

# Handling Scheduled Triggers During Restarts and Failures

In a Process Orchestrator high availability environment, schedules that are monitored on server A will move to Server B when server A fails. There is a small window (of a few seconds) during which a schedule could be missed while server B is detecting server A’s failure.

# Handling Events During Restarts and Failures

Process Orchestrator adapters resume sending events after the Process Orchestrator server restarts. While the adapter is still responsible for the implementation in this area, existing Process Orchestrator adapter implementations attempt to be consistent with this guaranteed delivery design. For example, when the SAP adapter reads CCMS alerts or the Remedy adapter polls for Incident state, the adapter stores the last-read record so that it can resume reading records starting with the next entry when the Process Orchestrator server resumes. For non guaranteed-delivery, network-initiated event technologies such as SNMP traps, the Process Orchestrator server cannot know about events that occurred while it was down. If required, many of these technologies can use highly-available intermediaries to persist the transient events. For example, there are tools to listen for SNMP traps and convert them to persistent stores such as log files or Windows events.

There are two types of event systems in the Process Orchestrator server:

- Event-based triggers. Because of the adapter implementations (see Handling Running Processes During Restarts and Failures, page 5-16), event-based triggers are not lost across server restarts or failures. Like state management in other areas of the product, trigger submissions are transactional. When Process Orchestrator adapters send a trigger to the Process Orchestrator server, processes depending on that trigger are initiated as a part of the submission. As with any other processes, these process instances are persisted to the database so that after a restart, the triggered processes are running. With the exception of transient, non-persisted events, such as SNMP traps or performance thresholds, there should be no time gap where events should be lost such that triggers fail to launch.
- Correlation. The Correlation feature allows a server to tie together a related series of events. This is achieved through a caching mechanism to ensure that the event data is available to the Correlate activities in processes when needed. This cache of events is not retained across server restarts, which can cause issues with some Correlate activities. Correlate activities can be used to make a process wait for a certain condition, or branch based on how many events with particular properties have been received in a particular time frame. Because the cache is not persisted across server restarts, events received before the restart will not be matched by Correlate activities. These effects can affect accuracy of a process “decision” made based on the data from a Correlate activity.

  In practice this has been found to not be an issue for the following reasons:

  - First, correlation is a fairly advanced feature in the product and is very rarely used in a majority of scenarios.
  - Second, best practices also dictate that a correlation time frame is configured to be fairly small, which will minimize both the likelihood and impact of this compound condition. The time required to restart a server will take up some or all of this time frame if the correlation time frame coincides with a server restart.
  - Third, these correlations are typically used in diagnostic situations, and often the events repeat if the problem recurs. Where a problem is still occurring, Process Orchestrator will typically pick up the condition the next time the diagnostic process launches.
  - Finally, Process Orchestrator automation packs tend to set these processes as non-persistent anyway so they are not restarted. If the Process Orchestrator server is down, it is usually best to get a fresh view of the health of the component being monitored rather than depend on the status before the Process Orchestrator server restart. In many cases, the failure is resolved and it just creates noise to record the old diagnosis.

# Windows Performance Counters

A Cisco Process Orchestrator server publishes performance counters that can be used to monitor its health and diagnose performance issues. These counters are grouped into several performance objects, which are discussed in the following sections.

The following standard Windows performance counters are also of interest:

- Memory—Pages/sec
- Processor—% Processor Time
- Process—Private Bytes
- Process—Handle Counts
- Physical Disk

## Cisco Process Orchestrator Object

*Table 5-3 Windows Performance Counters - Cisco Process Orchestrator*

| Counter | Description |
|---|---|
| Running Process Instances | The number of running process instances |
| Running Activity Instances | The number of running activity instances |
| In-Memory Activity Instances | The number of in-memory activity instances.<br>An in-memory activity instance might be running now or have run within a process that is running now. For example, if you have a single 10-step sequential process, there will always be only one active activity instance. But as the workflow continues to run, the number of "in-memory" activities will grow as the execution progresses. When the process stops running, the number drops to zero. |
| In-Memory Process Instances | The number of in-memory process instances.<br>In-memory activity instance might be running now, or have run within a process which is running now. For example, if you have a single 10-step sequential process, there will always be only one active activity instance. But as workflow continues to run the number of "in-memory" activities will grow as the execution progresses. When the process stops running, the number drops to zero. |
| Schedules Monitored | The number of schedules being monitored in Process Orchestrator. |
| Schedules Met | The number of monitored schedules met in Process Orchestrator.<br>If there is only one process definition that has an "Every day, every 5 minutes" schedule trigger, then "Schedules Monitored" will be 1 for as long as the server runs. "Schedules Met" will grow by one every five minutes as the schedule fires process instances. |
| Events Monitored | The number of events being monitored in Process Orchestrator. |

*Table 5-3 Windows Performance Counters - Cisco Process Orchestrator*

| Counter | Description |
|---|---|
| Events Fired | The number of monitored events fired in Process Orchestrator.<br>If there is only one process definition that has an "Every day, every 5 minutes" schedule trigger, then "Events Monitored" will be 1 for as long as the server runs. "Events Fired" will grow by one every five minutes as the schedule fires process instances. |
| Web Service Calls | The number of calls to the Process Orchestrator Northbound Web Service. |
| Completed Activity Instances /sec | The rate (per second) at which activities are being completed (with success or failure). |
| Completed Process Instances /sec | The rate (per second) at which running process instances are being completed (with success or failure). |
| Started Activity Instances /sec | The rate (per second) at which activity instances are being started. |
| Started Process Instances /sec | The rate (per second) at which process instances are being started. |

# Cisco Process Orchestrator Adapter Object

There are separate instances for each adapter.

*Table 5-4 Windows Performance Counters - Cisco Process Orchestrator Adapters Object*

| Counter | Description |
|---|---|
| Running Activities | The number of activities running under this adapter. |
| Activities Run | The number of activities this adapter has run |
| Disappearances | The number of times this adapter has disappeared unexpectedly. |
| Events Monitored | The number of events being monitored by this adapter. |
| Excessive Memory Restarts | The number of times this adapter has been restarted due to excessive memory usage. |
| Host Process ID | The ID of the process that hosts this adapter. |
| Idleness Shutdowns | The number of times this adapter has shut down due to idleness. |
| Requested Restarts | The number of times this adapter has requested to be restarted. |
| Starts | The number of times this adapter has started |
| Targets Managed | The number of targets this adapter is managing. |

# Cisco Process Orchestrator Persistent Queue Object

There are separate instances for each queue.

*Table 5-5 Windows Performance Counters - Cisco Process Orchestrator Persistent Queue Object*

| Counter | Description |
|---|---|
| %Used Space | The maximum space for the queue is preallocated; this counter measures how much of the queue is used. |
| Queue Item Submits/Sec | Number of items submitted to the persistent queue per second. |
| Queue Items | This is the current number of items in the queue. This number will vary as new items are added by the process execution and removed as the engine records them into the reporting database. |

# Cisco Process Orchestrator Data Access Layer Operations Object

There are separate instances for each type of DAL (Data Access Layer) operation.

*Table 5-6 Windows Performance Counters - Cisco Process Orchestrator Data Access Layer Operations Object*

| Counter | Description |
|---|---|
| # of Data Access Layer Calls | Number of times a particular DAL (Data Access Layer) call was made. |
| # of Errors in Data Access Layer Calls | Number of times a DAL call completed with an error. |
| Average Data Access Layer Call Duration (milliseconds) | Average time it took to complete this type of DAL operation. |
| Cumulative Data Access Layer Call Duration (milliseconds) | Total time spent in all DAL operations of this type. |
| Longest Data Access Layer Call Duration (milliseconds) | The time it took for the longest of DAL operations of this type to complete. |

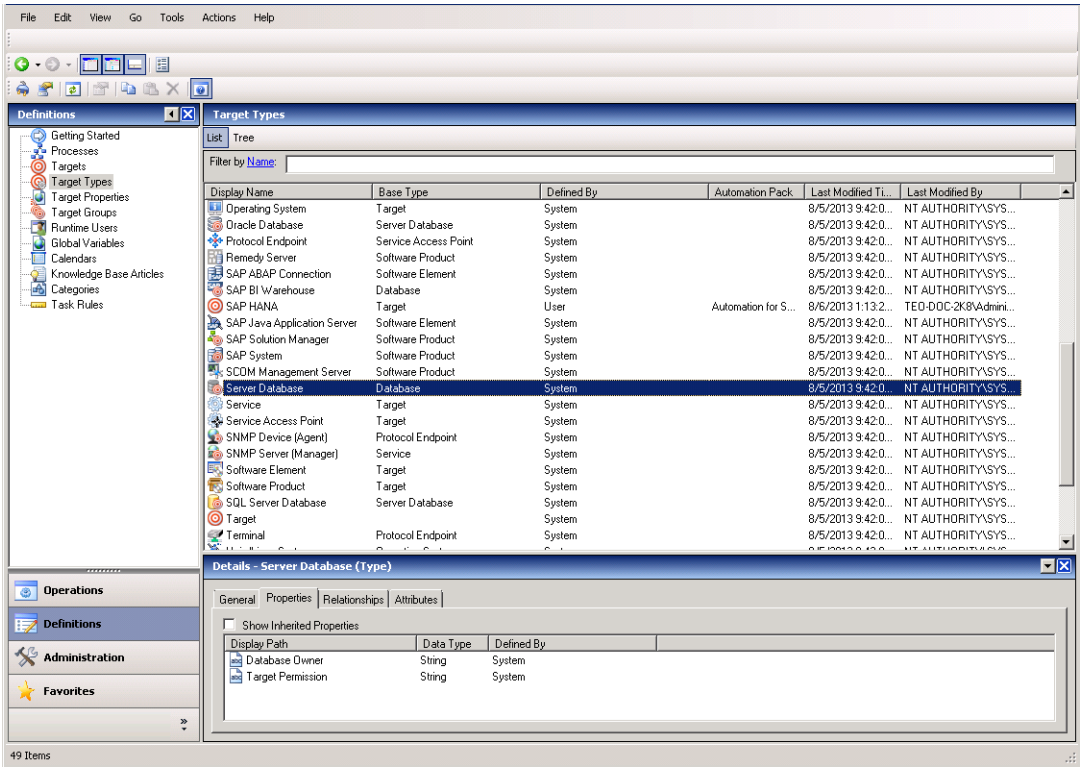

# Authoring Target Types

Target types provide a way to define a service or other IT element that is not represented by any target type provided by an adapter. All new targets are created based upon a target type.

Target types define inheritance, properties, and relationships. Similar to other aspects of automation configuration, target types can be placed in an automation pack that is version controlled, enabling the partner community to define new types of services or augment those delivered by others.

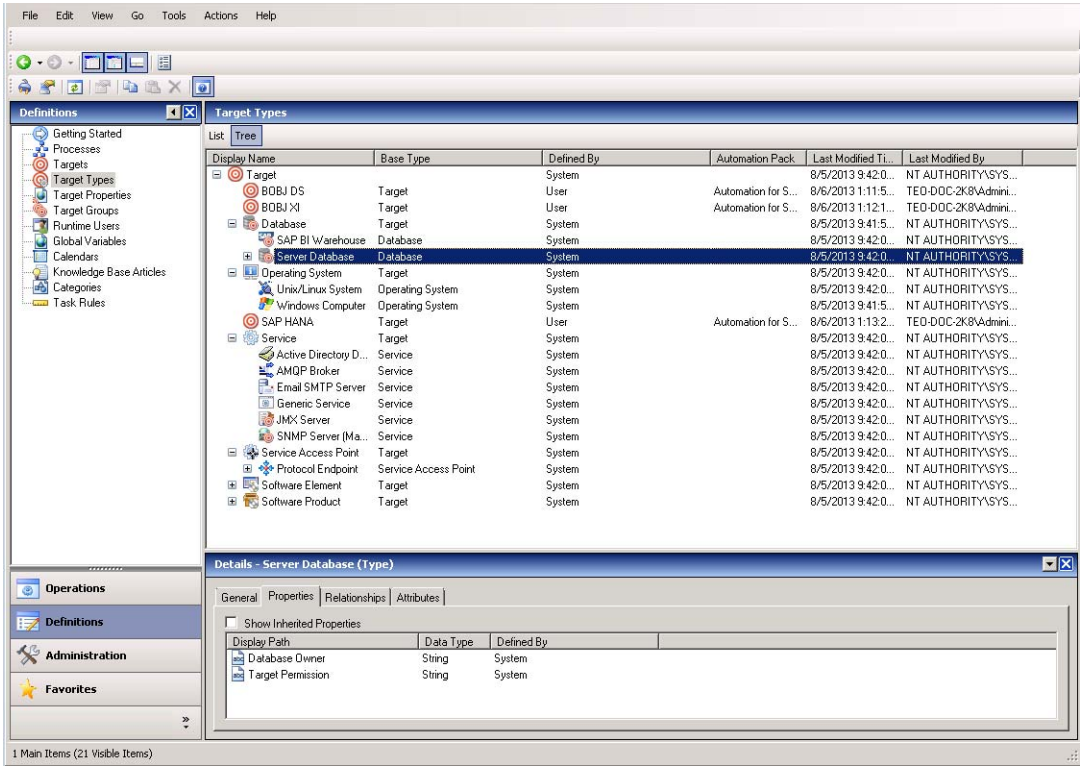
Use the Definitions > Target Types view to display the list of defined target types.

***Figure 6-1 Definitions > Target Types, List View***



Because target types have an inheritance hierarchy, they can also be shown as a tree list view based upon inheritance. Click the **Tree** and **List** buttons to toggle between these views.

***Figure 6-2 Definitions > Target Types, Tree View***




The following topics provide information about target types and how to manage them.

- Creating a Target Type, page 6-3
- Defining Inheritance, page 6-4
- Defining Target Properties, page 6-4
- Defining Relationships, page 6-6

### Related Topics

- Target Types, page 1-17
- For a generic overview of related object-oriented programming concepts, see this video: https://www.youtube.com/watch?v=lbXsrHGhBAU

# Creating a Target Type

To create a target type:

**Step 1** Choose **Definitions > Target Types**, right-click, and choose **New >Type**.

**Step 2** On the New Type Wizard, **General** pane, specify the basic properties for the new type, then click **Next**.

**Note** For information about the field parameters, see the online help.

When you create a new target type, you can define what target type your new type inherits from by specifying its *base type*:

a. In the General dialog box, click **Base Type**.

b. Select the base target type from the **Select Target Type** list and click **OK**.

   Your new type will inherit from the base type. After the target type has been created, you can change the target properties to use a different base type, but your options are very limited (see Defining Target Properties, page 6-4).

When you create a new target type, you can assign it a custom icon. To assign a custom icon to a target type:

a. In the General dialog box, click **Use custom icon**, then click **Select icon…**

b. Use one of the pre-packaged icons or choose the **Custom** tab.

   The Custom tab allows you to choose images that have a .bmp extension. If you choose to use this option, you must provide the appropriate image.

c. After the target type has been created, you can change the target properties to use a different icon (see Defining Target Properties, page 6-4).

**Step 3** On the Properties pane, specify the new properties for this type, then click **Next**.

- Each target type that you define can have a set of properties that can come from a variety of automation packs.
- Process Orchestrator allows you to add properties to any type, not just the types you ship in your automation pack.
- Properties can inherit from ancestors.
- You can package properties you add in automation packs and distribute them.

**Step 4** Click the **Relationships** tab, specify the new relationships (target references) for this type, then click **Next**.

**Step 5** On the Completing the New Type Wizard panel, click **Finish** to create the new type.

### Related Topics


- Adding Targets and Target Groups, page 7-19
- Creating Triggers, page 8-1
- Defining Inheritance, page 6-4

# Defining Inheritance

The idea of inheritance is that a type can contain all aspects of another type but can be extended to provide a more specialized type; that is, inheritance formalizes the relationship. For example, a cat is a specialized type of a mammal, and a mammal is a specialized type of an animal. A cat "is a" mammal, and hence "is an" animal. If a mammal generically can do something like wake up, the cat can also do it, but the cat can do additional things, such as meow and purr. If you define a property of a mammal, like its weight or body temperature, the cat would also have that attribute, but you could add additional properties like the breed, being Siamese, tabby, and so on.

This concept is very useful in real world modeling of types of IT services and can greatly simplify implementations. The concept can allow common aspects of a collection of types to be done once in a shared type, then inherited, rather than repeating the common implementation in each type in the collection. The mechanism avoids duplicating properties and processes, and therefore makes it easier to maintain the automation.

Conversely, it is often simpler to implement something new by taking something that already exists in automation and extending it for the new specialization. This allows you to focus implementation on only what is new and unique, rather than all of the aspects that are common with the existing automation.

**Note** If a type has a "has a" relationship rather than an "is a" relationship, use Target Relationships (see Defining Relationships, page 6-6), not inheritance.

Inheritance and target relationships are mutually exclusive concepts; inheritance creates a *subtype* from a type, while relationships model the *connection* or association between one target and another target.

#### Related Topics

- To define inheritance, see Creating a Target Type, page 6-3.
- To use inheritance, see Selecting the Target Type on Which a Process Acts, page 7-5.

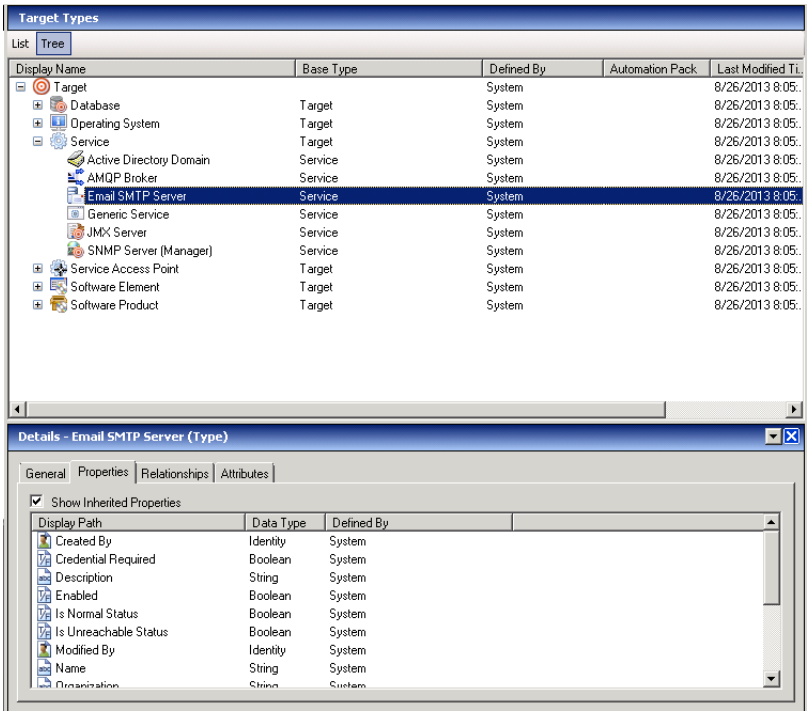
# Defining Target Properties

Target properties allow the storage of information for targets created from a target type. For example, a target property could store network device discovery information such as operating system, version, various optional cards or modules installed in a device, serial numbers, or even configuration. Network automation can then use this information in its workflows. Target properties can be used in target groups that specify collections of targets matched by some criteria.

Target Properties can include a default value. This can be important in allowing a general setting that is only overridden on specific targets. For example, you might define a global threshold, then allow customization for a specific target instance where needed.

The list of properties on a type inherits from ancestor types, but the type can insert additional properties. You can package target property definitions from the type as well as target instance property values in automation packs to transfer them between environments. The author has full control of the evolution of their properties within types, including support for upgrade and other lifecycle management.

*Figure 6-3 Properties on Target Types*




Process Orchestrator provides a unique implementation of target properties that allows properties to be moved between types. Prior to target types, Process Orchestrator provided a Service Target type, which allowed modeling arbitrary services with a single type. This built-in type still exists, and is now called "Generic Service." Upgrades will function with no problem. However, this approach should be used less in the future now that new types are possible.

Using target types, you can:

1. Create a new subtype of a generic service.
2. Move the properties in some namespace from the Generic Service to the new type.
3. Rename the properties to eliminate the namespace and simplify the name.
4. Move the inheritance from generic service to some new place in the type structure.
5. Ship the automation pack for the new type to a customer.

Importing the new pack will upgrade the types and their properties to the new structure, without losing the values stored in the properties.

To define properties for a target type:

**Step 1** Choose **Definitions > Target Types**, highlight the appropriate target type, right-click, and choose **Properties.**

**Step 2** Click the **Properties** tab, update the properties for this target type, then click **OK**.

For information about the different types of variables used in some properties, see Supported Variable Types, page 7-18.

For information about the property field parameters, see the online help.

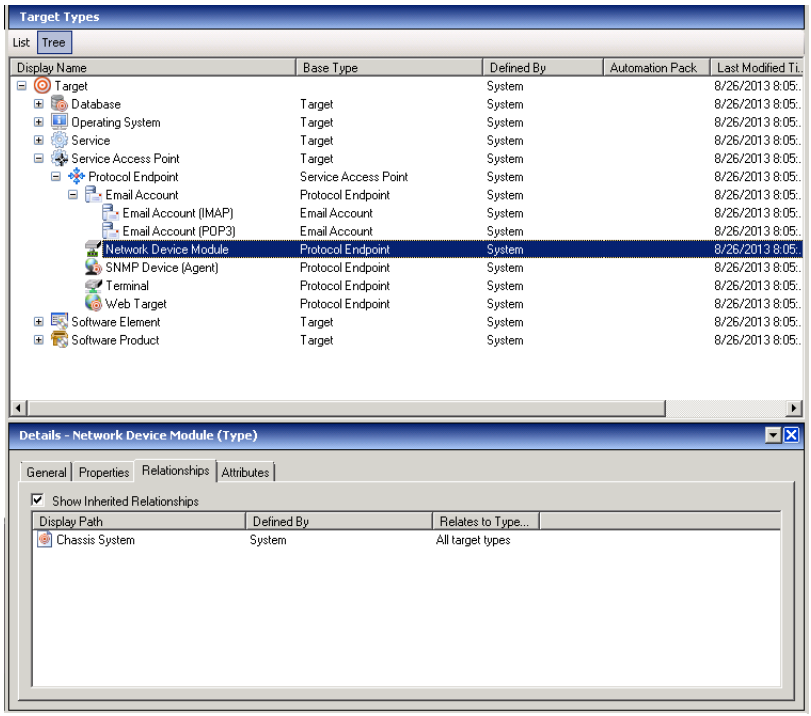
# Defining Relationships

A relationship is basically just a named target reference. It can also be thought of as a "Target Reference" property.

The Relationships tab in the Target Types properties page displays both inherited relationships as well as relationships that are explicitly defined on this target type. The full set of inherited relationships is shown:

- System-defined relationships exposed from the built-in base system target types are visible but read-only.
- User-defined relationships can be modified.

***Figure 6-4 Relationships within a Target***




To define relationships for a target type:

**Step 1** Choose **Definitions > Target Types**, highlight the appropriate target type, right-click, and choose **Properties.**

**Step 2** Click the **Relationships** tab, then choose **New > Target Reference Property**.

**Step 3** Using the New Target Reference Property Wizard, create a target reference for this target type, then click **Finish**.

**Note** For information about the field parameters, see the online help.

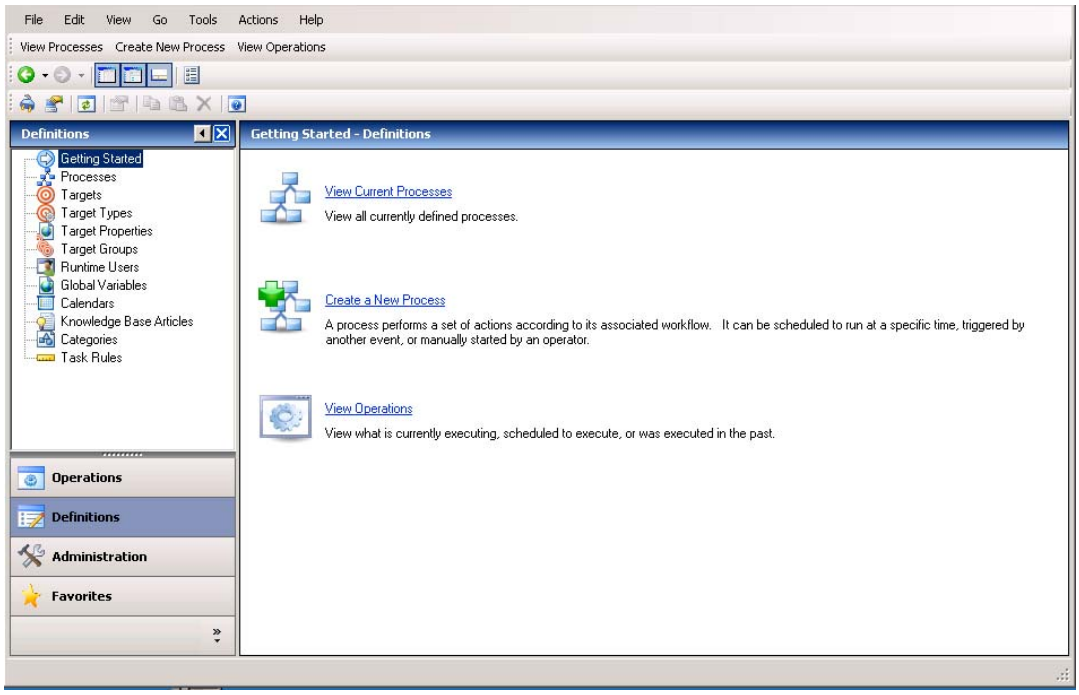

# Authoring Processes

A process is basically a constructed workflow that consists of activities, invocations of child processes, and logic components that can be included to complete the process. Cisco Process Orchestrator allows you to automate IT processes based on your organizational requirements using a workflow format. The Process Orchestrator also includes many pre-defined processes in its packaged automation packs.

Processes can be viewed from both the Operations and Definitions workspaces:

- The Operations workspace displays scheduled and running processes. This workspace displays the Activity Views, Process Views, and Target Views. These views display the execution progress of the process instances.
- The Definitions workspace displays the defined processes and provides options to edit existing processes and define new processes. From the Processes View, you can also execute processes and view the status of a process in progress.

***Figure 7-1 Getting Started—Definitions***



The following sections provide information about creating and managing processes:



# Accessing the Process Editor

Cisco Process Orchestrator simplifies process creation using a drag and drop workflow designer. The Process Editor allows you to drag activities, workflow activities (logic), or processes you want to invoke into a process definition where you want it to occur. The Process Editor dynamically accepts the new addition and shows the resulting logic.

Unlike drag-and-wire approaches common to other orchestrators, the logic of the process is always clear, without having to constantly modify the layout as the process definition evolves. This makes it easier for other operators to view or edit processes that have previously been created. You can also collapse sections of the workflow that are not currently of interest.

Use the Process Editor to:

- View and modify the properties of an existing process
- Define properties
- Construct a workflow for a new process

**Note** The procedure to access the editor depends on the task to be performed.

To access the Process Editor:

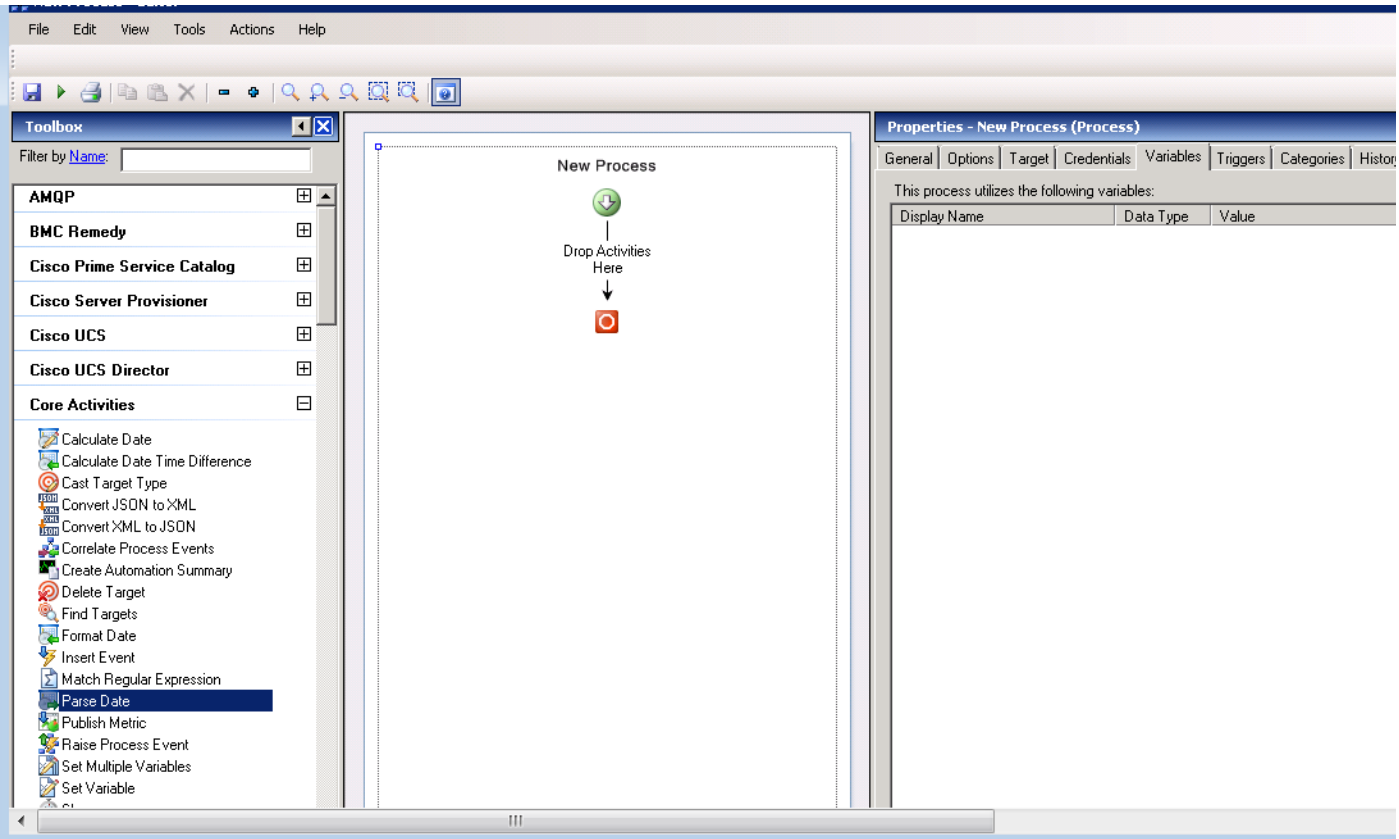
- If you are creating a new process, use *one* of the following options:
  - Choose **Definitions > Processes**, right-click and choose **New > Process**.
  - From the Actions menu, choose **New > Process**.
  - On the Actions toolbar, choose **New > Process**.

**Note** The New button on the Actions toolbar is not available if a process is highlighted in the Processes Results pane.

- If you are modifying an existing process, on the Processes Results pane, right-click the appropriate process, then choose **Edit**.

The Process Editor opens.

***Figure 7-2 [Process Name]—Editor***




### Toolbox Pane

The toolbox pane on the left side of the editor includes these views. The navigation items displayed in the toolbox depend on the view that is selected.

| View or Tab | Description |
|---|---|
| Activities | The Activities view displays the list of activities that are used to construct the process workflow.<br><br>The activities that display depend on the adapters that are installed. |
| Processes | The Processes view displays the list of defined processes that can be included in other processes.<br><br>To add an existing process to a new process, drag and drop the appropriate process onto the workflow pane and define the criteria for execution of the process. |

| View or Tab | Description |
|---|---|
| Logic | The Logic view displays the list of workflow components that support or define the workflow logic and provide control over the execution of the workflow logic.<br><br>For information about configuring the logic components, see Adding Logic Components to a Process, page 7-12 |
| Favorites | The Favorites view allows you to choose commonly used items from all the other tabs. For example, if you often use AMQP activities and some of the core activities, you can add them all to your Favorites tab. Then, instead of having to locate the activities in the activities tab, they can be quickly located in Favorites tab. Similarly, if you have favorite processes or logic items, these can also be added to the Favorites. |

#### Workflow Pane

The Workflow pane is a canvas located in the center portion of the Process Editor (see Figure 7-2). Use this area to create processes by dragging and dropping activities, other processes, and components from the toolbox onto the canvas.

Process Orchestrator ships with automation packs that contain built-in process definitions that cannot be modified. If you select one of the shipped processes, a limited version of the Process Editor displays only the workflow pane and properties pane; the Toolbox pane in the Process Editor is not displayed.

#### Process Pane

The Process pane is located on the right side of the editor and displays the properties for the selected process, as well as selected activity, child process, or workflow logic element properties.

## Toggling the Process Editor View

In the Process Editor view, you can toggle the view between the process property pages and the activity view pages.

To switch to the process properties, use *one* of the following options:

- Choose **File > Process Properties**.
- In the Workflow pane, click anywhere outside of the activities in the workflow.

## Expanding the Workflow View

To expand the workflow, use *one* of the following options:

- Choose **View > Expand**.
- On the toolbar, click the **Expand** tool.

## Collapsing the Workflow View

To collapse the activities in the Workflow pane, use *one* of the following options:

- Choose **View > Collapse**.

- On the toolbar, click the **Collapse** tool.

# Creating a Basic Process

**Before You Begin**

Identify any drag-and-drop objects in the toolbox that will be required by your new process.

To create a basic process:

**Step 1** Choose **Definitions > Processes**, right-click and choose **New > Process**.

**Step 2** On the Process Properties pane, define the process properties.

**Note** Because Process Orchestrator no longer enforces uniqueness of object names, two content authors can create and ship automation packs or processes with identical names. As a result, a customer importing both automation packs will have two seemingly identical processes. To avoid confusion to the customer, use a namespace (a unique prefix) when naming objects in Process Orchestrator. However, there are no technical issues with having duplicate process names.

**Step 3** On the Toolbox pane, drag and drop the appropriate items onto the workflow pane (see Toolbox Pane, page 7-3).

**Step 4** On the Properties pane, define the properties for each object selected on the workflow pane. The available property pages are determined by the selected objects.

**Step 5** When all of the property pages are complete, click **Save** to save the process.

## Reverting Process Changes

This option is used when changes have been made to the process definition, but are no longer necessary. It deletes all changes made since you last saved the process. If you have saved the process and have not made any additional changes, no action will be taken.

To reverse process changes, choose **File > Revert**.

The process reverts back to the last saved version of the process. All new activities, components, and other changes are removed from the process definition.

# Selecting the Target Type on Which a Process Acts

Process definitions control the target types on which they act. This selection supports inheritance.

To select the target types on which a process can run:

**Step 1** Open the process in the Process Editor (see Accessing the Process Editor, page 7-2).

**Step 2** Click the **Target** tab, then click the target types on which this process can run.

*Figure 7-3 Viewing Target Types in a Process Definition*

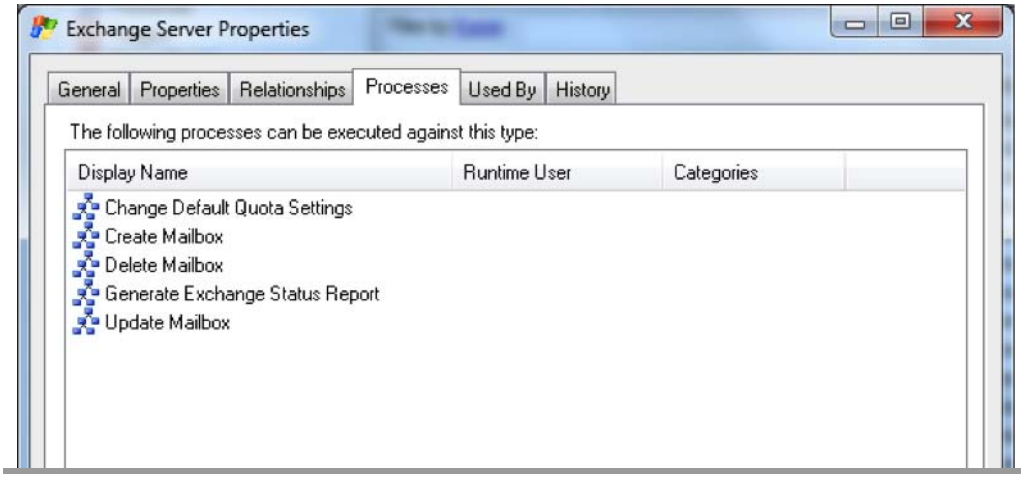

For example, assume that an SQL Server 2008 type provides a more specific implementations of an SQL Server Database through inheritance. All process that can run on an SQL Server Database run against a SQL Server 2008; a SQL Server 2008 "is a" SQL Server Database.

To view the processes that run on a given target type or target instance:

**Step 1** Choose **Definitions > Target Types**, highlight a target type, right click and choose **Properties**.

**Step 2** Click the Processes tab. The list shows the processes that can be executed against this target type.

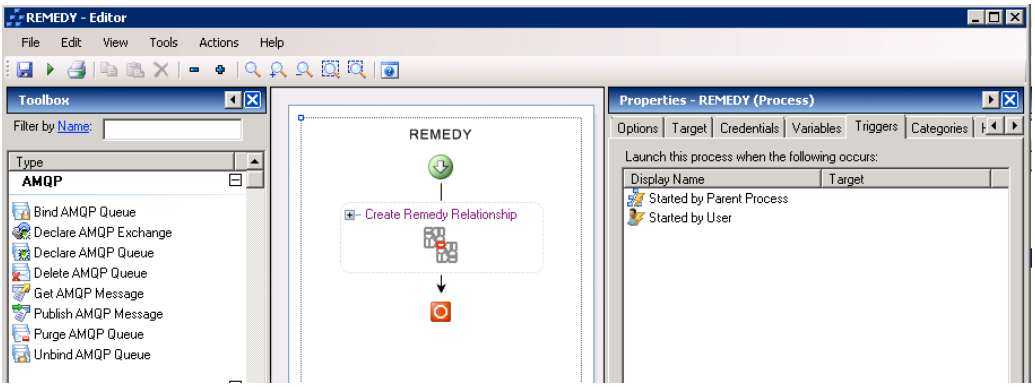
*Figure 7-4 A List of Processes That Act on a Type in a Type Definition*



# Adding an Invocation of an Existing Child Process

Before an existing process is available in the Processes pane of the Process Editor and can be used in another process' workflow, the Started by Parent Process trigger must be defined in that existing process.

***Figure 7-5 Started by Parent Process Trigger Example***




**Before You Begin**

- Confirm that the process you want to invoke includes the trigger, Started by Parent Process.

To invoke an existing process as a child of the primary process:

**Step 1** Choose **Toolbox > Processes,** then highlight and drag the appropriate process to the Workflow pane.

**Step 2** On the [Child Process Name] Properties property sheet, click the **General** tab and modify the general property information.

**Note** For information about the field parameters, see the online help.

**Step 3** Click the **Target** and **Credentials** tabs and modify those properties as needed.

**Step 4** Click the **Inputs** tab in the [Variable Name] field and modify the parameters for executing the child process. If the process does not have any input variables, the page will be blank.

**Step 5** Click the **Save** tool to save the child process definition.

# Authoring Task Activities

Task activities allow the creation, manipulation, and deletion of tasks within process workflows. They allow processes to interact with task records.

There are two primary types of tasks:

- **IT process records** tasks include the alerts, incidents, and change requests. For more information, see IT Process Records, page 1-22.

- **Human interaction** tasks are steps in a workflow where a human needs to take action. For more information, see Human Interactions, page 1-23.

# Authoring Task Activities

A process workflow consists of one or more *activities* that can be executed by the product. An activity definition contains actions and any other information required for the product to execute it.

To add an activity that creates a new task or acts on an existing task within a process workflow:

**Step 1** In the Process Editor, choose **Toolbox > Tasks,** choose the appropriate activity, then drag and drop the activity onto the Workflow pane.

- Use IT Records activities to create workflows that handle basic IT functions. To access these activities, choose **Toolbox > Tasks - IT Records > [Activity Name]**.
- Use user interaction activities to create workflows that handle functions that require human interaction. To access these activities, choose **Toolbox > Tasks - User Interactions > [Activity Name]**.

**Step 2** On the **[Activity Name]** property pages, define the properties.

**Note** For information about the property field parameters, see the online help.

- Click the **Assignment** tab to modify individuals or groups of individuals assigned to the task and other assignment properties.
- Click the **Parameters** tab to define parameters for a specific task.
- Click the **Categories** tab to:
  - Assign a category to the task or change existing category assignments.
  - Check the **Task should inherit process categories** check box to inherit the categories assigned to the process.
- Click the **External** tab to specify the external incident management system information to support the synchronization between Process Orchestrator and the system that the customer is using.
- Click the **Web Form** tab to apply a specific XSL transformation to the task XML. XSLT transformation is used to convert XML text to the HTML web pages used by the Web Console (see Chapter 13, “Using the Web Console.”)
- Click the **Knowledge Base** tab to assign a knowledge base article to the object.
- Click the buttons on the **Result Handlers** tab to manage the Condition Branches on the workflow.

**Step 3** Click **Save** to save the activity definition.

The following table summarizes the available task activities.

*Table 7-1 Summary of Task Activities*

| Task Activity Name | Purpose |
|---|---|
| Assign Task | Modify the list of assignees for a task. On this activity, tasks can be assigned to a user or group of users. Users can also remove current assignees from tasks. |
| Correlate Alert | Watch for alerts raised before or after a process starts and that are related to data from the process, such as its trigger, input parameters, activity outputs, or other elements referenced from the process workflow. |
| Find [Task] | Find tasks that were created in the product and that match the specified criteria. |
| Publish Task to Event Log | Retrieve task information and publish it to the Cisco Process Orchestrator Tasks event log on the Process Orchestrator server. |
| Update [Task] | Update the properties for a specific task. The properties for the tasks can be updated using their respective activities in the Toolbox pane. |
| Wait for Task to Enter State | Wait for a task to match a specific state before the activity continues.<br>If the task reaches a completed state without matching, or if the duration expires without matching a specified status, then the activity succeeds without matching. |

# Task Properties That Task Activities Manipulate

The following table summarizes the types of changes you can make to task properties within a task activity in your process workflow.

For information about the field parameters, see the online help.

*Table 7-2 Summary of Common Actions Against Tasks That Can be Defined in a Process*

| Use Case | Description |
|---|---|
| Modifying automation summary properties | Define the properties used to capture relevant diagnostic and state information about the process execution.<br>On the activity property page, click the **Automation Summary** tab.<br>This tab is available only in these activities:<br>• Create Alert<br>• Update Alert<br>• Create Change Request<br>• Update Change Request<br>• Create Incident<br>• Update Incident |
| Adding an assignee to a task | Change the assignee or other assignment properties for the activity.<br>On the activity property page, click the **Assignment** tab. |

*Table 7-2 Summary of Common Actions Against Tasks That Can be Defined in a Process (continued)*

| Use Case | Description |
|---|---|
| Assigning duplicate criteria for a task | Define the criteria to indicate that the alert is a duplicate. If the alert is a duplicate, the new alert is automatically resolved as a duplicate and linked to the original alert as a related task.<br>On the activity property page, click the **Duplicate** tab.<br>This tab is available only in these activities:<br>• Create Alert<br>• Update Alert |
| Removing an assigned duplicate property | Remove a task property from the duplicate criteria.<br>On the activity property page, click the **Duplicate** tab, then click **Remove**. One field will always remain.<br>The Remove button removes the last property added to the list. For example, you cannot remove *Parameter 4* without removing *Parameter 5* from the list.<br>To keep *Parameter 5*, update *Parameter 4* with the information from *Parameter 5*, then click **Remove** to remove *Parameter 5* from the list. |
| Adding a related alert | Assign alerts that the incident originated from.<br>On the activity property page, click the **Related** tab, click **Add** then, enter the Alert ID or click the **Reference** tool to search for the alert property to add to the list. |

# Debugging Task Activities

The actions in this section demonstrate things you can do when viewing an activity that created a task or that manages tasks. For example, you might do them when you want to view a process workflow that actually ran and look at a specific task-related activity in the workflow.

## Viewing Correlated Alert Results

When the Correlate Alerts activity is launched, the alerts that were found by the activity are displayed from the Operations Workspace activity instance view.

To view the correlated alerts results:

**Step 1** Choose **Operations > Activity Views**.

**Step 2** Highlight the **Correlate Alerts** activity instance, right-click and choose **Properties**.

**Step 3** Click the **Correlated Alerts** display-only tab to view the alerts collected by the Correlate Alerts activity.

## Viewing Find [Activity] Results

Use the following steps to review the results of the query generated by the Find Alerts, Find Change Requests, and Find Incidents activities.

To view the Find [Activity] results:

**Step 1** On the Operations workspace, click the Activity Views folder.

**Step 2** Highlight the **Find [Activity]** activity instance, right-click and choose **Properties**.

**Step 3** Click the **Results** display-only tab to view the list of alerts, change requests, and incidents queried by the Find activity.

**Note** For additional information about the property fields, see the online help.

## Viewing Wait for Task to Enter State Results

When the Wait for Task to Enter State activity is launched, the matching tasks that were found by the activity are displayed from the Operations Workspace activity instance view.

To view the correlated alerts results:

**Step 1** On the Operations workspace, click the Activity Views folder.

**Step 2** Highlight the **Wait for Task to Enter State** activity instance, right-click and choose **Properties**.

**Step 3** Click the **Result** display-only tab to view the status values matched by the Wait for Task to Enter State activity.

The following information is displayed:

| Field | Description |
|---|---|
| Result | Values for the task state<br>• Unknown—Indicates activity is running or has failed to complete<br>• Timed Out—Activity timed out without matching<br>• Task Completed Without Matching—Task has reached a completing state that was not a state that was not included we were attempting to match on.<br>• Matched—Status was matched by the task |
| Matching Status | Indicates the status that was matched |

# Adding Logic Components to a Process

Insert process logic components into a process to support or configure the process logic and provide control over the process execution.

The following table summarizes the types of logic components and when to use them.

***Table 7-3 Summary of Logic Components***

| Activity Type | Purpose |
|---|---|
| Completed | Signal the completion of an activity and terminate the process. The component ends the workflow and sets the state of the workflow to *Succeeded*, *Failed (Completed)*, or *Failed (Not Completed)*. |
| Condition | Execute one of the defined Condition Branches in a process. It checks conditions for each of the branches, in order from left to right, and executes the first Condition Branch whose condition is *true*. |
| Condition Branch | Create a branch in the process and execute the branch only if the specified condition is met. |
| For Each | Add the activities to the process that should be executed one time for each item in the target source. |
| Parallel | Run two or more branches of a process simultaneously. The component consists of two or more sequential block components that execute their activities in parallel. |
| Sequence | Run activities in a block in a sequential order. |
| Start Point | Indicate points within a process workflow (in addition to the top of the process workflow) at which you can start the process.<br>You will always be able to start the process from the beginning. However, if a process contains a Start Point component, then you can start the process from the location of the Start Point within the process. |
| While | Execute a sequence of child activities (contained in the While Block) that repeats as long as the specified condition is *true*. |

To add logic components to a process:

**Step 1** In the Process Editor, choose **Toolbox > Logic**, then highlight and drag the appropriate logic component to the Workflow pane.

**Step 2** On the component property pages, define the properties.

**Note** For information about the field parameters, see the online help.

**Step 3** Click **Save** .

# Adding Process and Global Variables

The variables feature provides a storage area for information that is used on a regular basis to avoid having to specify the same information in several places. Data stored in a variable can be altered to affect process execution behavior.

You can use variables in Process Orchestrator for a variety of purposes. For example, as:

| Usage | Description |
|---|---|
| Name Variables | The most common use of variables is a name that has a changeable value. For example, a global variable can be used to store information used in processes such as:<br>• Locations of files and directories<br>• Email addresses<br>• Order numbers<br>• User names |
| Process or Activity Property Variables | In a process or activity definition, you can refer to the process properties or the properties of a prior activity in the process. In this scenario, the properties of the process or activity can also refer to associated objects.<br>One of the most common uses of variables is to define activity configuration. Any field in an activity with a Reference tool can refer to a variable value rather than an explicit value. |
| Status Tracking Variables | Another common use of variables is to track state. For instance, you can use variables as a loop counters to store the number of times a loop has executed and know the current loop iteration running. |
| Summary Variables | You can also use a variable to build up a 'summary' message. For each event that happens, you can append 'what just happened' to the variable. At the end of a process, the result will be the contents of this variable as an entire summary of the process. |

The following topics describe how to create and apply variables:


- Creating a Global Variable, page 7-14
- Creating a Process Variable, page 7-15
- Inserting Variable References, page 7-17
- Setting a Variable, page 7-18
- Supported Variable Types, page 7-18


Target types and target properties are often a better alternative to using variables. See:

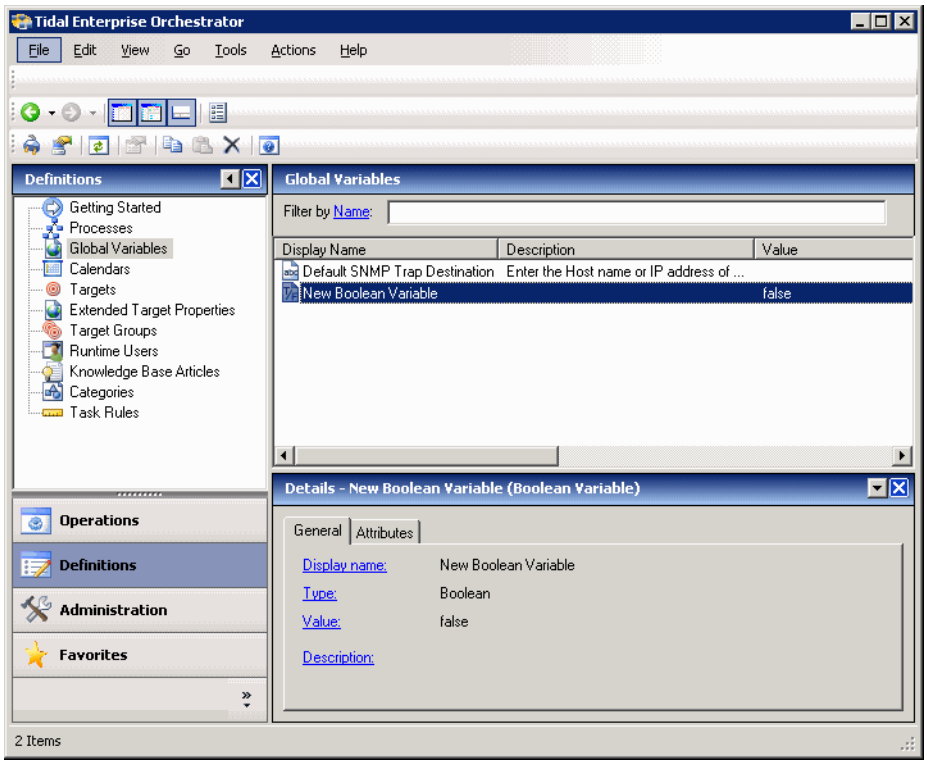

- Creating a Target Type, page 6-3
- Defining Target Properties, page 6-4

# Creating a Global Variable

Global variables span processes. All processes can reference or update them. Process Orchestrator often stores data such as performance thresholds in global variables so that customers can edit them easily without modifying process definitions, and automation packs can then update processes without affecting customer settings.

The Definitions > Global Variables view displays all the defined global variables in the console. Additional features from this view include creating a new global variable, modifying and deleting a global variable.

***Figure 7-6 Definitions—Global Variables***




To create a global variable:

**Step 1** Choose **Definitions > Global Variables**, right-click and choose one of the global variables types.

The New [Name] Variable Wizard launches.

**Step 2** Click the **General** tab and enter the general information about the variable.

**Note** For information about the field parameters, see the online help.

**Step 3** On the Variable Value panel, enter the appropriate value for that variable type, then click **Next**.

**Step 4** Click **Finish** to complete the procedure.

## Creating a Target Reference Global Variable

Use a target reference global variable to define a variable containing a reference to a target of a specific type. For example, perhaps there are several targets representing nodes of some service. One variable might reference the active node, and another variable might reference a standby node.

**Note** If you have a target reference global variable, there is exactly one target reference. If you really want a target reference for every target (with possibly a default target reference), create a target relationship defined on a target type. For more information about how target references are used to define relationships in target types, see Defining Relationships, page 6-6.

To create a target reference global variable:

**Step 1** Choose **Definitions > Global Variables**, right-click and choose **New > Target Reference Variable**.

**Step 2** Enter the general information about the variable, then click **Next**.

**Note** For information about the field parameters, see the online help.

**Step 3** On the Reference Type panel, enter the appropriate target type to use as a reference, then click **Next**.

**Step 4** On the Variable Value panel, choose one or more targets to specify the initial value of the variable, then click **Next**.

**Step 5** Click **Finish** to complete the procedure.

# Creating a Process Variable

Process variables are created from within a process and can be used as a reference value to store or pass a value between executions of a process or between steps within a single process. Process variables can also be used to collect input parameters from the user or parent process. These variables are only available from within the defined process and cannot be accessed or referenced by objects outside of the process.

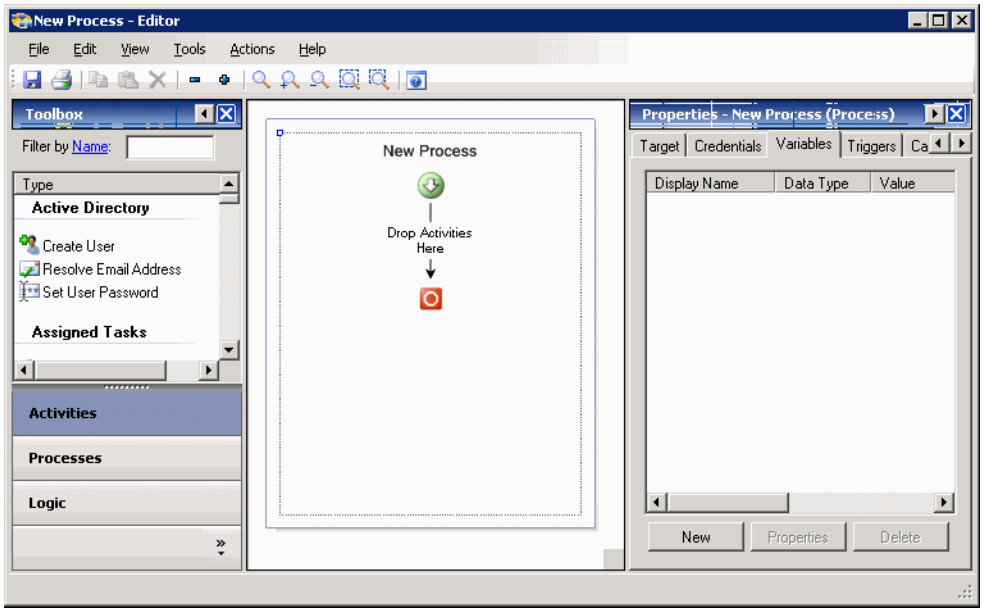
There are several types of process variables:

| Variable Type | Description |
|---|---|
| Definition | Definition variables are created in the process definition and only that process can reference them. The value is persistent and shared across instances of that process. |
| Input | Input variables are created in the process definition. They specify data (parameters) that can be passed into the process. Input variables can optionally be marked as required. |

| Variable Type | Description |
|---|---|
| Output | Output variables are created in the process definition. They specify data that the process can return to its caller. |
| Local | Local variables are created in the process definition and only that process can reference them. However, a new instance of the variable is established for each process instance, so that one instance's interaction with the variable cannot affect another instance of the process. |

The Process > Variables tab displays the defined variables for the open process.

***Figure 7-7 Process Editor Dialog—Variables Tab***




Variables can be defined in the process when something is used multiple times. For example, if there is one person who is must always be the first approver, you can create a variable for that person so you do not have to reenter the email address in the activity properties; you can simply reference the variable.

To add a process variable to a process:

**Step 1** In the Process Editor, choose **Process**, click the **Variables** tab, then click **New > [Variable Type]**.

**Step 2** Enter the information required for the variable type.

**Note** For information about the field parameters, see the online help.

**Step 3** Click **OK** to close the dialog box.

# Inserting Variable References

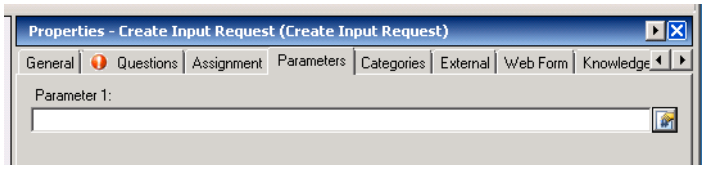
The Reference tool to the right of a text field indicates that the field can be populated by referencing a defined variable or the property of an activity in the process. Use the Insert Variable Reference dialog box to choose a defined variable or object property to populate a field.

Only variables valid for the selected field can be selected in this dialog box. The OK button does not activate until a valid object or variable is selected.

To insert a variable reference:

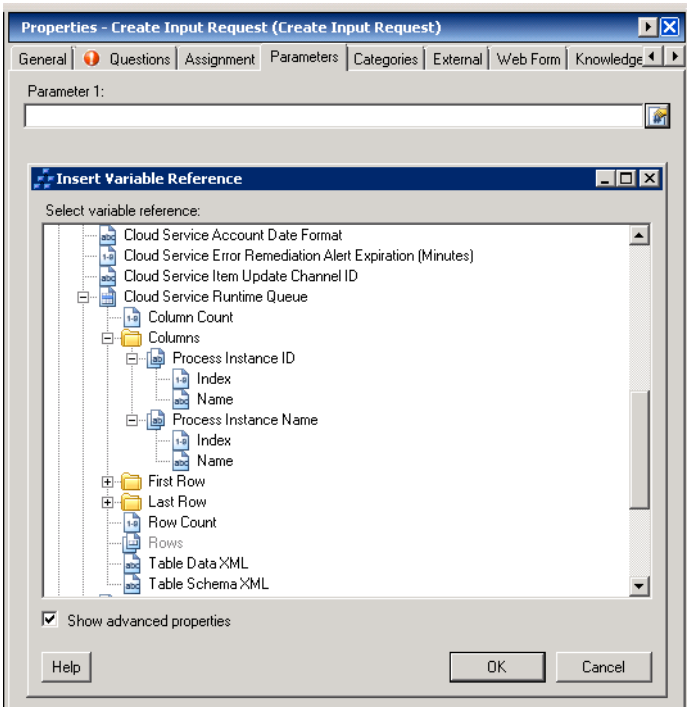
**Step 1** To the right of a field on a property page, click the Reference tool.

***Figure 7-8 Process Editor—Reference Tool Location***




**Step 2** Expand the appropriate top level items to locate the appropriate variable.

***Figure 7-9 Sample Variable References***




**Step 3** From the list of available of items, select the appropriate property or variable and click **OK**.

# Setting a Variable

Use the Core activity, Set Variable, to modify the value of a defined variable. Choose the variable that you want to modify and enter the new value for the variable.

**Note** To set several variable values at the same time, use the Set Multiple Variables activity. It is more efficient than using several Set Variable activities.

You can also include formulas to modify variable values. For example:

```
5+10
-or-
[Activity.Reference1] / [Activity.Reference2] * 100) + [Activity.Reference3]
-or-
[Activity.PropertyName1] [Activity.PropertyName2]
```

The Set Variable activity *cannot* modify:

- Table variables. To manipulate the values of table variables, see Using Table Activities, page 12-3.
- Target properties. To manipulate the values of target properties, use the Update Target activity.

# Supported Variable Types

The following variable types are supported:

| Variable Type | Description |
|---|---|
| Boolean | Indicates whether a set of elements should be interpreted as *true* or *false*. |
| Hidden String | Holds data that must be protected from other Process Orchestrator users and from auditing operations performed by Process Orchestrator.<br>For more information about hidden strings, see Using Hidden String Variables, page 7-19. |
| Identity | Represents the value of a user or group identity. |
| Numeric | Single whole or decimal number (positive and negative). |
| String | Defines a variable containing a string of text. |
| Table | Stores a set of records in a table format.<br>**Note** Table variables are supported by a separate activity called Set Table Variable. They are also supported by the Set Multiple Variables activity, but are *not* supported by the Set Variable activity.<br>For information about modifying the format of the tables, see Using Table Activities, page 12-3. |
| Target Reference | Contains a reference to a target of a specific type (see Creating a Target Reference Global Variable, page 7-15). |

## Using Hidden String Variables

Hidden variables are used to specify protected secrets (such as HR information or security information, such as a password) when they are outside of the use cases of runtime users. For example, you might need to pass a password to a script. Runtime users in Process Orchestrator are fully protected, so no process can retrieve the password from any runtime user. Hidden strings provide an alternative for these use cases. Another example is a password that must be passed to a web service invocation.

When values for hidden variables are entered or displayed, their values are obscured. When they are stored, their values are encrypted so that they cannot be compromised by UI or database access.

A common use case is when a password is required to access some service. Typically there will be a target that is associated with the connection, such as a web service. Using service-oriented orchestration principles, you can either:

- Extend the existing target type to add a "password" property of a hidden text type.
- Create a new specialized target type that inherits from the target type and create a "password" property on the new target type.

For example, say that you use a web target to connect to a Cisco Prime Network Services Controller REST API, and that the API requires a password. You could create a new "Cisco Prime Network Services Controller" target type which inherits from web target, then add a hidden text target property "password" on that type. You can then securely store passwords only for Prime Network Services Controller connections and not for all web services in general.

**Caution** In some cases, hidden strings can be passed outside Process Orchestrator. For example, a hidden text variable might be passed to a Windows script or a web service invocation. The variable value is passed in clear text outside the tool. *Care must be taken in this regard.* For example, if someone put a network sniffer on the web service call and you used HTTP rather than HTTPS for the call, the sniffer would be able to see the value. Process Orchestrator protects the value of the variable within the product itself and its logs, but it is the responsibility of the author and their IT department to protect these secrets outside of the product.

**Related Topics**

Runtime Users, page 1-18

# Adding Targets and Target Groups

When you create a process, you must specify where you want the process to run. You can also specify that the process runs on a specific target or target group.

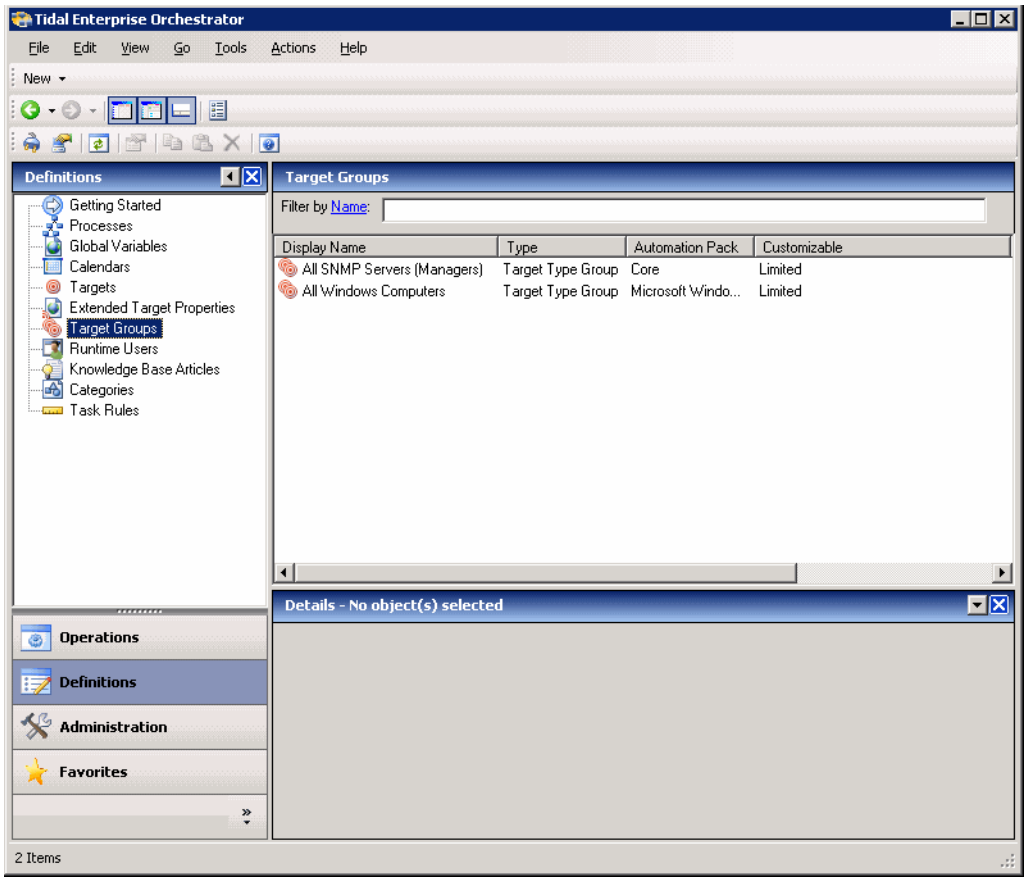
**Note** When you move process definitions from environment to environment using automation packs, the specific targets will be different in the new environment. Therefore, if you plan to export your process in an automation pack, you will want to avoid references to specific targets. Instead, use a target group in your process definition, and place the environment-specific targets in that target group. Use direct target references only in environment-specific processes.

The target group can be defined once and reused in several processes. For example, you might have a database maintenance process that is scheduled to run every month on all database servers. Instead of scheduling the process multiple times to run on each database server, you can create a target group that includes all the database servers and schedule the process to run on all the servers at the same time.

If you choose to execute the process on a target group, you can further specify to run the process on all objects that are included in the target group, or run the process on a specific object within the target group.

Use Definitions > Target Groups to view the defined target groups. From this view, you can create new target groups, modify the properties of a target group, and delete target groups.

***Figure 7-10 Definitions > Target Groups***




# Viewing Members of a Target Group

Viewing members of a target group is important when debugging processes and environments.

To view members of a target group:

**Step 1** Definitions > Target Groups, select the target group, right click and display properties for that group.

**Step 2** Click the **Members** tab.

**Step 3** Examine or view the targets considered to be part of the group by Process Orchestrator.

# Creating a Target Group

You can create several types of target groups:

- A target type group specifies the type of target to be included in a target group as well as additional property-based criteria for a selecting target.
- A virtual target group contains a collection of any type of target or target group that has been defined.
- The Active Directory group types, Active Directory Group and Active Directory Organizational Unit (OU), are supplied by the Active Directory adapter. Membership in these group types is determined dynamically.

To create a new target group:

**Step 1** Choose **Definitions > Target Groups**, right-click and choose **New > [Target Group Type].**

**Step 2** In the New [Target Group Type] Properties dialog, enter the appropriate information for that group type.

**Note** For information about the field parameters, see the online help.

**Step 3** Click **OK** to create the new target group.

**Related Topics**

Deleting an Object, page B-6

# Using a Target Group in Your Process

To use a target group in your process, define the target selection criteria:

**Step 1** In the Process Editor, click the **Target** tab, click **Execute on this target group**, then click **"...**".

**Step 2** From the Select Target Group dialog, select one or more target groups, then click **OK**.

**Step 3** You can choose to run the process on:

- All targets in this group
- All targets that satisfy the specified criteria
- Any target that satisfies the specified criteria
- A target with a specified name

**Step 4** Enter the criteria (if needed).

**Note** For information about the field parameters, see the online help.

# Adding Conditions

The conditions assigned to an object within a process specify when an action is to be taken based on an evaluation of conditions that have been defined. For an explanation of the types of conditions supported by Process Orchestrator, see Conditions, page 1-20.

You can access condition property pages from many places, including the Process Editor and the Task Rules dialog box. The following steps describe how to add a condition using the Process Editor.

To add a new condition:

**Step 1** In the Process Editor, choose **Toolbox > Logic**, then drag a **Condition** or a **Condition Branch** into the workflow panel.

**Step 2** Use the Basic panel to create a simple variable condition that uses a variable to match against a value using a defined criteria operator. When you edit a condition using the Basic panel, you are actually creating a compound condition where every subcondition is ANDed together.

To create a basic condition:

a. Click **Basic**. If there are no conditions, the condition will either be 'TRUE' or 'FALSE' based on the user selection. If there are conditions, the condition is *evaluated* to either TRUE or FALSE.

b. Modify the condition properties.

c. Update the properties for each condition.

**Step 3** Use the Advanced panel to create a more complex condition. You can define the properties of the conditions within the Advanced panel, as well as within the Properties dialog box.

When you edit a condition using the Advanced panel, the compound conditions are explicitly shown. This allows you to create complex conditional logic with the ability to nest any type of condition inside of a compound condition. To add a child compound condition in the advanced editor, choose **New > Compound Condition**.

The conditions specified on the Basic panel can also be configured on the Advanced panel because they transition to simple Process Orchestrator-level variable conditions. All other Process Orchestrator conditions on the Advanced panel cannot transition to the Basic panel; for these conditions, the message “This is an advanced condition and can only be displayed in advanced view” displays.

To add an advanced condition:

a. Click **Advanced**. If there are no conditions, the condition will either be 'TRUE' or 'FALSE' based on the user selection. If there are conditions, the condition is *evaluated* to either TRUE or FALSE.

b. Click the appropriate button to modify the condition properties.

**Note** For information about the field parameters, see the online help.

c. After the first condition is added, choose the appropriate operator (AND or OR). The operator is set to *AND* by default.

**Step 4** Click **OK**.

# Deleting a Condition

A condition can have subconditions, and anywhere there is a condition you can delete a subcondition. Multiple condition equations can be included in a condition. Conditions are deleted only from the object upon which they reside.

## Deleting Conditions from Task Rules

Use the following steps when deleting a condition from a task rule. If you delete the task rule, that automatically deletes the condition.

**Step 1** Choose **Definitions > Task Rule**, highlight an existing task rule, right-click and click **Properties**.

**Step 2** Click the **Conditions** tab, highlight the appropriate condition, then click **Delete**.

**Step 3** Click **OK** to save your changes.

## Deleting Conditions from Triggers

Use the following steps when deleting a condition from a trigger. If you delete the trigger from the process properties, that automatically deletes the condition.

**Step 1** In the Process Editor, choose **Toolbox > Processes**, highlight the process, and choose **Process > Edit**.

**Step 2** In the Process Properties pane, click the **Triggers** tab, highlight the appropriate the trigger, and click **Properties**.

**Step 3** Click the **Conditions** tab, highlight the appropriate condition, then click **Delete**.

**Step 4** Click **OK** to save the changes made to the trigger.

**Step 5** Click **Save** to save your changes.

## Deleting Conditions from a Logic Component

Use the following steps to delete a condition from a logic component such as a Condition Branch or a While Block. If you delete the logic component from the Workflow pane, that automatically deletes the condition.

**Step 1** In the Process Editor, choose **Workflow**, then the relevant logic component.

**Step 2** Choose *one* of the following:

- Right-click and choose **Delete** to remove the logic component and all associated conditions.
- Click the logic component's **General** tab, highlight the appropriate condition, then click **Delete**.

**Step 3** Click **Save** to save your changes.

## Deleting Archive Conditions

Use the following steps when deleting a condition from an archive. If you choose another archive option from the process properties, that automatically deletes the condition.

**Step 1** In the Process Editor, choose **Toolbox > Processes**, highlight the process, and choose **Process > Edit**.

**Step 2** In the Process Properties pane, click the **Options** tab.

**Step 3** In the Archival section, choose one of the following:

- Click **Archive based on condition**, click **"..."**, then use the Archive Condition dialog to delete the appropriate condition or subcondition.
- Check another Archival option and the condition will not be used; you do not need to "delete" it.

**Step 4** Click **Save** to save your changes.

# Managing Processes

## Modifying Process Properties

Process definitions are modified in the Process Editor. With the appropriate rights from the Definitions view, the Process Editor is launched when accessing the process properties. Users with the appropriate rights can also modify processes in the top panel of any of the Process Views in the Operations View.

When user rights are restricted, the Process Viewer is launched with the properties displaying a display-only view.

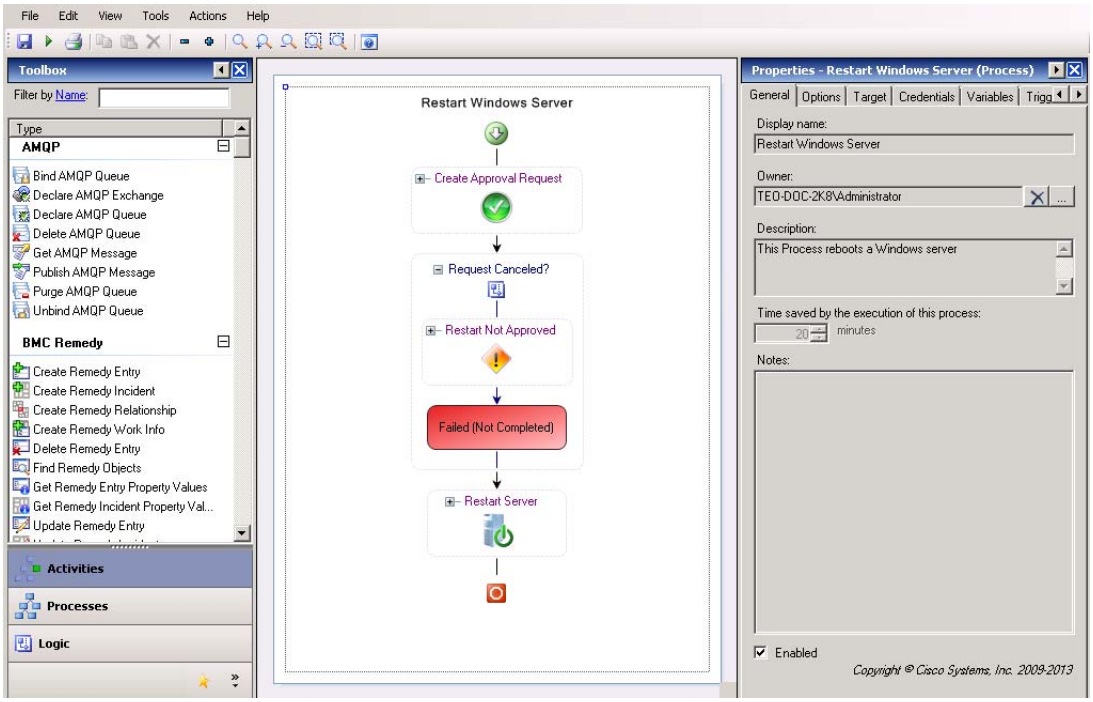
To modify process properties:

**Step 1** Choose **Definitions > Processes**, highlight the appropriate process, right-click and choose **Edit**.

Even when a process comes from an automation pack, some aspects of that process can still be updated. For example, you can:

- Disable/enable existing triggers
- Add additional triggers
- Disable/enable the process
- Change options (such as archive/resume options)

*Figure 7-11 Editing Process Properties*




**Note** If you have been adding activities while editing a process, your focus will often be on an activity, so you will actually be seeing properties of the activity. To return to editing process properties, click in the white space outside of any activity.

**Step 2** Select the appropriate process property tab and modify the fields as necessary.

**Note** For information about the field parameters, see the online help.

**Step 3** Click the **Save** tool to save your changes.

# Modifying Activity Definition Properties

Activities can only be modified in the Process Editor; modifying a process does not automatically modify an activity. Activity definitions are *included* in the workflow of a process definition, so the activity properties must be modified separately from the process properties.

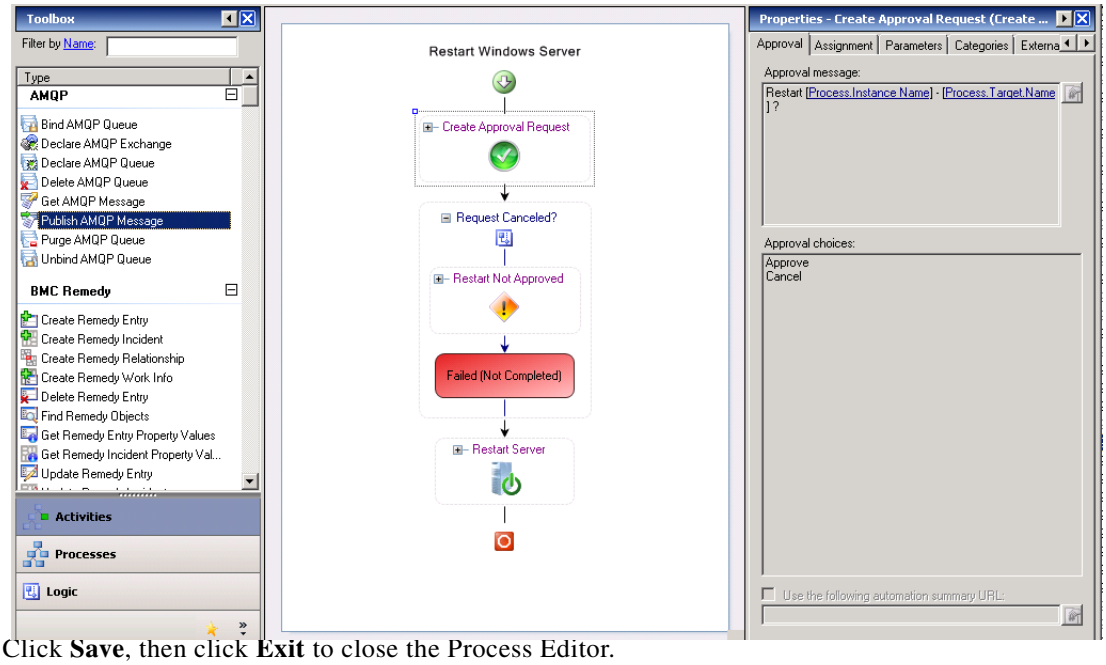
To modify an activity:

**Step 1** Click **Definitions—Process**, highlight the appropriate process, right-click and choose **Edit**.

**Step 2** On the Process Editor Workflow pane, choose the appropriate activity and modify the activity properties as necessary.

**Note** For information about the field parameters, see the online help.

*Figure 7-12 Editing Activity Properties*




**Step 3** Click **Save**, then click **Exit** to close the Process Editor.

# Starting a Process

You can manually start an process displayed on a Process View or the Process Editor under these conditions:

- You have the appropriate privileges.
- The process includes the Started by User trigger.
- The process is enabled (see Enabling and Disabling an Object, page B-5).

Only one process can be manually started at a time.

**Note** When a process is started manually, all conditions and triggers included in the process definition will be overridden.

To start a process:

**Step 1** Choose one of the following:

- From a process view, highlight the appropriate process, right-click and choose **Start Process**.
- From the Process Editor, highlight the appropriate process, then choose **File > Start** or click the **Start** button in the toolbar.

**Step 2** In the Confirm Start Process dialog box:

- If the process has any input variables, verify the variables associated with the process in the Parameters table. To update the variable, highlight the variable and click **Edit** to modify the value.
- To start a process from a specific starting point, check the **Start from start point** check box and then select the appropriate starting point from the drop-down list. The first activity after the specified starting point will run first.
- To specify a target that is different from the default process target, check the **Override target (Target name)** check box and select a target from the drop-down list.
- To create a new target for this process, click **New**. For additional information on targets, see Creating Targets, page 2-4.

**Step 3** Click **OK** to confirm.

**Step 4** The Start Process Results dialog box displays the progress of the process. To view the process workflow, double-click the process instance.

**Step 5** Click **Close** to return to the console.

# Archiving Process Instances

The Process Orchestrator provides settings that control:

- Whether the server saves the process state after the process completes.
- Whether the process is restartable in case of a server restart or failover.

Both the archival and resume options affect performance (how much data is saved into the database and how often). However, only the resume option controls whether the process is restartable. If the process is set to “archive” but “not resume”, it will save the data to the database but will not restart when the server restarts.

To specify the archival and resume settings for an existing process:

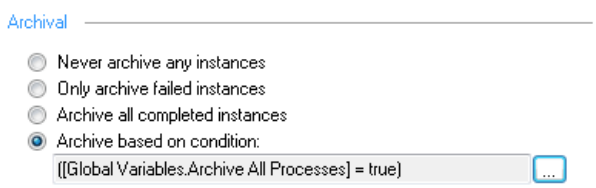
**Step 1** In the Process Editor, choose **Toolbox > Processes**, highlight the process, and choose **Process > Edit**.

**Step 2** In the process properties dialog, click the **Options** tab.

**Step 3** In the Execution section, if you want the server to resume the process that is running when server restarts or fails over, click **Resume execution if interrupted**.

**Step 4** In the Archival section, choose one of the archival options. For example, the following is an example of a archive based on a condition:

***Figure 7-13 Archive Based on a Condition***



**Note** For information about the property fields, see the online help.

**Step 5** Click **Save** to save your changes.

**Related Topics**

- Adding Conditions, page 7-22
- Archiving Process Instances in an Automation Pack, page 11-6
- Chapter 5, "Managing High Availability and Resiliency"

# Creating Processes as a Team

Multiple users might be working together to develop processes that they plan to group into an automation pack. The Process Locking feature is designed to enforce limits on access to a process. You can lock one or more processes from the Process Orchestrator console.

Process Locking also includes these features:

- The UI provides a filter for the process definition view to view processes locked by me, or processes locked by someone else.

  The columns for these views show whether a process is locked, who has the lock, and since what date / time the lock has been held.
- Locked processes:
  - Are read-only to other users.
  - Cannot be enabled or disabled by other users.
  - Can be added or removed from automation packs by other users because this action does not modify the configuration of the process itself.
  - That are locked by another user can be unlocked if you have Break Lock permissions (see Editing Locked Processes, page 7-29).

During the editing process:

- If User A attempts to edit a process that is currently locked by User B, User A receives a warning message and the process displays in Read-only mode.
- If User A is editing a process, which is then locked by User B, User A receives an error message when attempting to save the process.
- User A cannot delete a process that is currently locked by User B.
- A confirmation dialog displays if you close the Process Orchestrator and there are still locked processes remaining.

Team content development *is*:

- A way for users to lock *processes* when they are developing content along with other process authors
- A new environment-wide setting which forces users to lock processes when editing them

Team content development is *not*:

- Locking other object types

- Source control or versioning support
- Being notified in real-time that someone has locked a process that the user is editing

# Enabling Team Process Authoring

The Environment Properties dialog box contains an option to enable/disable team process authoring on a server. If enabled, it:

- Locks and unlocks processes for editing
- Provides a server-wide setting that causes Edit process to become equivalent to Locking a process for editing
- Warns users when they attempt to close the console and still have processes locked for editing

To enable this option:

**Step 1** Choose **File** > **Environment Properties**.

**Step 2** Check the **Enable team development** check box to enable team process authoring on this server.

**Step 3** Click **OK** to save your changes.

# Viewing Process Views by Lock Status

To view locked processes, choose **Definitions > Processes**, choose **Filter by Lock**, then choose one of the options from the drop-down list.

# Editing Locked Processes

Use the Process Locking feature to edit locked processes:

*Table 7-4 Summary of Process Locking Tasks*

| Task | Purpose |
|---|---|
| Lock a process | Enforce limits on access to a process. Locked processes are read-only to other users.<br>Choose **Definitions > Processes**, highlight the appropriate process, right-click and choose **Lock**. |
| Unlock a process | If you have locked a process, you can unlock that process from the Process Orchestrator console.<br>Choose **Definitions > Processes**, highlight the appropriate process, right-click and choose **Unlock**. |
| Break a lock | If a process is locked by another user, and if you have Break Lock permissions, you can unlock that process.<br>Choose **Definitions > Processes**, highlight the appropriate process, right-click and choose **Advanced > Break Lock**. |


# Working with Events and Triggers

Triggers are events and conditions in the system that can fire off processes. The attributes of the trigger that fires off a process instance can be referenced in the process workflow. Processes often use this data to control execution.

Process Orchestrator supports two types of rule-based triggers:

- Events—The Process Orchestrator can monitor for events from the environment, and you can specify triggers that initiate processes when the subscribed event occurs. For example, an event might be an incoming SNMP trap or a fault on a UCS system.events and schedules.
- Schedules—Schedules allow triggering processes at some time by leveraging calendars that define which days something can occur. Calendars can be selected days or sequences of dates such as weekly or monthly, they can represent dates like fiscal quarter end, or they can be combined hierarchically. Schedules then associate a time with a calendar. When the day is in the calendar, the time is evaluated. Times can be explicit or repeating (for example, hourly).

The following sections provide information about working with triggers and events:



**Related Topics**

- Chapter 7, “Authoring Processes”
- Chapter 9, “Scheduling Processes”

# Creating Triggers

The following table summarizes some of the more commonly-used types of triggers and when to use them. For a complete list of trigger types, see the *Adapter Integration and Automation Packs Guide*.

***Table 8-1 Commonly-Used Trigger Types***

| Trigger Type | Purpose |
|---|---|
| AMQP (Advanced Message Queuing Protocol) | Use AMQP to create Process Orchestrator triggers for events that occur out on the IT landscape for which Cisco does not have adapters, but that can generate AMQP messages. |

*Table 8-1 Commonly-Used Trigger Types*

| Trigger Type | Purpose |
|---|---|
| Process Event | Process events allow one process to pass an event to other processes. For example:<br>• A Raise Process Event activity can post a process event, and a trigger can monitor for a specific event.<br>• A Correlate Process Events activity allows monitoring for an event within a process workflow.<br>Process events include elements of the Process Orchestrator functional model so that they are aware of internal schema elements. For example, you can include a target type in your subscription criteria.<br>Process events are exposed in the northbound web service, so an external system can programmatically submit an event to Process Orchestrator. However, the use cases for process events generally center more around Process Orchestrator-internal use cases; Advanced Message Queuing Protocol (AMQP) is preferred for external message passing.<br>For internal use cases, process events have the advantage because they are native and lightweight within Process Orchestrator. It is easy to create message driven architectures within Process Orchestrator using process events. Since these events are not persisted to the database, they are very lightweight. Process events have the advantage that they do not require external setup and installation, as would be the case with AMQP. |
| Started by Parent Process | Indicates that a process can be used by a parent process. If this trigger is not set in the child process, the parent process will fail.<br>The information used to start the child process can also be included in the automation summary. |
| Started by User | Indicates that the process can be started manually by a user. This trigger is added by default to a process, but can be removed.<br>If a user attempts to manually start a process that does not have the ad-hoc trigger (or the user is restricted from starting the process in the ad-hoc trigger), an error message will be displayed and the process will not be launched.<br>This trigger will expose all the typical trigger properties, as well as some additional properties that allow you to determine exactly how the process was manually started. |
| Started by Web Service | Indicates that the process can be started using the Northbound Web Services. For more information, see the *Cisco Process Orchestrator Northbound Web Services Guide.* |
| Task Event | All task events are monitored by the Process Orchestrator environment. A user can create a trigger that can watch for any task event (task created, task deleted, task updated) that matches the user-specified criteria.<br>For more information about tasks, see Authoring Task Activities, page 7-7 |

*Table 8-1 Commonly-Used Trigger Types*

| Trigger Type | Purpose |
|---|---|
| Target Event | Target events complement target types to allow event-driven automation for each type of service. When used as triggers, target events enable policy, which allows you to invoke automation as data is created or changed, according to patterns in the data. Available target events include:<br>• Target Created Event—Target properties are available in reference control.<br>• Target Changed Event—Current values for target properties are available in reference control. Names of "changed" properties are also available in reference control.<br>• Target Deleted Event—Deleted target properties available in reference control.<br>For more information about targets, see Creating Targets, page 2-4. |

# Creating a New Trigger

To create a new trigger for an existing process:

**Step 1** On the Process Properties pane, click the **Triggers** tab, then choose **New > [Trigger Type]** from the drop-down list.

**Step 2** On the **[Trigger Type]** property pages, define the properties.

**Note** For information about the field parameters, see the online help.

**Step 3** Optional: To specify that the event should be monitored on targets that match specific criteria:

a. On the Trigger Target property page, click **Monitor on this target group** and the appropriate target group.

b. Under Monitor for Events on, click **Choose a target using this algorithm**, then click **Choose the target that satisfies the specified criteria**.

c. Click **"..."** to launch the Target Selection Criteria dialog box.

d. On the Properties pane, specify the required information, then click **OK** to return to the Trigger > Target property page.

**Step 4** Optional: Complete the following tabs, then click **OK**.

- Condition—Specify when an action is to be taken based on an evaluation of the defined conditions. See Adding Conditions, page 7-22.
- Knowledge Base—Choose the appropriate knowledge base article to associate with the trigger. See Authoring Knowledge Base Articles, page 12-8.

**Step 5** Click **OK** to add the trigger to the process.

**Step 6** Click **Save** to save the process with the new trigger.

## Adding Trigger Logic to Process Workflows

You can check the attributes of a trigger in a process, so that if one process has multiple triggers (such as started by user, started by parent process, and a trigger from an AMQP message), you can check which of these events occurred and branch the execution accordingly. The properties of the event that triggered the process are available in the process workflow using reference control.

# Correlating Events

Process Orchestrator contains a technology called Correlex that allows events that match data in the workflow (such as a trigger) to be correlated. Process Orchestrator's event correlation capabilities are real time, in memory, geared towards typical IT staff, not programmers, and can be combined with active probing and action activities to leverage correlated data.

Process Orchestrator's correlation activities can be placed anywhere in a workflow and reference any data from prior workflow steps. It is common for workflows to capture diagnostic data to perform decision tree branching before they correlate other events. A workflow might:

- Probe the system to see if a service is up.
- Test system performance.
- Look up configuration to find a relationship that is not evident in events themselves and then use that new information to capture events.

Process Orchestrator's process workflows can include activities other than correlations to capture supplemental non-event data in real time that might be gone before the end of a classic event correlation time window.

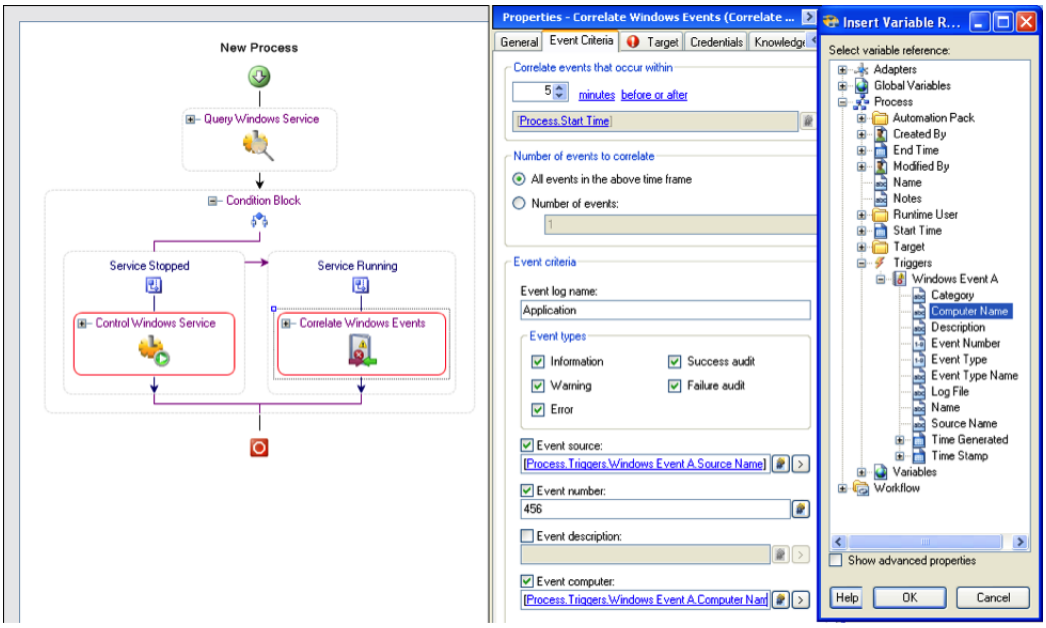
For example, assume there are two events, A and B, and there is a need to handle things one way when only A occurs, and another when A and B happen together. In correlation, a time window is required; if A occurs, it will not be clear whether A and B both occur until the end of the time window. So the results of the correlation might not be able to publish the results until sometime after A occurs. But with complex enterprise systems, often issues can be fleeting. There is a need to capture data right when a condition occurs or it might be gone; in other words, the tool might miss event A while looking for event B. Too often for difficult issues, troubleshooting data is not available at the time the operator is notified or some correlation response script occurs.

In Process Orchestrator, when event A triggers the workflow, the workflow can first focus on data capture, then move on to correlation. In this way Process Orchestrator has at its disposal data that is more real-time, and more complete, than correlation has previously allowed, which means that analysis in Process Orchestrator workflows can be more sophisticated. The workflow can solve a superset of cases compared to traditional correlation, and the data presented to operators is more complete and accurate.

This capability is exposed through Correlate activities. For example, the Correlate Windows Events activity allows the correlation of related events from the Windows Event Logs.

Process Orchestrator provides Correlate activities that correspond to many types of supported triggers. Typically, if Process Orchestrator provides a trigger, it will also provide a correlation activity, though this is not explicitly required.

***Figure 8-1 A Correlate Activity***




Correlate activities have several key pieces of data that specify the events to find:

- The time window in which the event can occur
- The number of events to match. If a number is specified, and when the process workflow execution gets to the correlate activity that number of events has already occurred within the time window, the Correlate activity will immediately match. If not, the activity will block waiting for those events to occur. If *all* is specified, the activity will wait until the end of the window and then return the results.
- The event criteria. This presents fields for filtering that type of event and is specific to the event type and adapter.

Events that match a Correlate activity are available to the later steps in the workflow and are published in the automation summary.

From a technology perspective, the server implementation:

- Allows activities to complete in real time. If you wait for one event and a matching event arrives, the activity will complete immediately rather than waiting until the end of the time window.
- Handles cases where event arrival might precede process execution of the correlation node. A process might not even be executing when an event arrives; it could be triggered later and look retrospectively for matching events. The server must cache data given its knowledge of Correlate activities in process definitions so it is ready to provide matched events should a Correlate activity execute later.
- Allows that correlation activities may never run due to untriggered workflows or workflow logic taking a different flow.

- Performs intelligent event caching and garbage collection based on process definitions.

## Creating Correlate Event Activities

Use a Correlate activity to check whether an event occurred.

To add a Correlate activity:

**Step 1** In the Process Editor, choose **Toolbox > Activities**, enter `correlate` in the Filter by Name text box, then select the correlate activity for the specific event you want to collect.

**Step 2** On the Correlate activity property page, click the **General** tab and enter the appropriate information.

**Note** For information about the field parameters, see the online help.

**Step 3** Click the **Event Criteria** tab and specify the required event properties for the activity.

**Step 4** Click **Additional properties to match** and specify the appropriate information.

For information about the displayed operators, see the online help.

The Expression arrow displayed to the right of the Reference tool indicates that a wildcard expression is available. For a list of the common wildcard expressions, see the online help.

**Step 5** Click **Save** to complete the activity definition.

**Step 6** To view the correlated event results:

a. Choose **Operations > Activity Views**, highlight the correlate activity instance, right-click and choose **Properties**.

b. Click the **Correlated Events** display-only tab to view the alerts collected by the correlate activity.

# Integrating Using the Advanced Message Queuing Protocol (AMQP)

Use AMQP to create Process Orchestrator triggers for events that occur out on the IT landscape for which Cisco does not have adapters. For example, to trigger a Process Orchestrator process in response to a syslog event:

**Step 1** Set up an AMQP queue/exchange with a feed from an off-the-shelf component such as this: https://github.com/bkjones/bevis.

**Step 2** Set up the IT landscape elements (application) to post messages to the newly created queue.

**Step 3** Set up a Process Orchestrator trigger from the queue/exchange.

**Related Topics**

Advanced Message Queuing Protocol (AMQP), page 1-15

# Debugging Process Trigger Instances

Use the Process Editor to view the process triggers. The Triggers tab in the Properties panel displays the triggers associated with the process and each trigger's target.

To debug process trigger instances:

- Review the product's auditing log (choose **Operations > Auditing > System**) for any errors that the server or adapters might have experienced while monitoring for an event.
- If a process contains a trigger, use the Process Editor to view the trigger criteria that the adapter will be monitoring. In the Process Instance view, you can see the specific occurrence of the trigger that caused the specific process instance to execute.

One debugging technique is to create a new separate process that triggers from the same event as the process you are trying to debug, and set the trigger criteria to include all events with no filters. You can then monitor the operations views to see instances of the process which launch, and look at their trigger attributes to see all events. Be sure to delete these debugging processes when you are done diagnosing your problem.


# Scheduling Processes

When defining a process, you can specify when the process will execute. You can execute a process based on a schedule. A schedule specifies one or more times of day, and is combined with a calendar that specifies which days the schedule should initiate the process.

The following topics include how to define a calendar, how to use calendars to define a schedule for a process, and how to use calendars in a condition:



## Authoring a Calendar

Calendars are reusable for schedules within many processes. For example, you can define a calendar for Saturdays. When defining a process that you want to run on Saturdays, you reference the *Saturday* calendar. Other examples include a calendar that includes weekends and company holidays when IT might perform scheduled maintenance, or the last week of a fiscal quarter when IT might exclude non-essential automation or deny change requests.

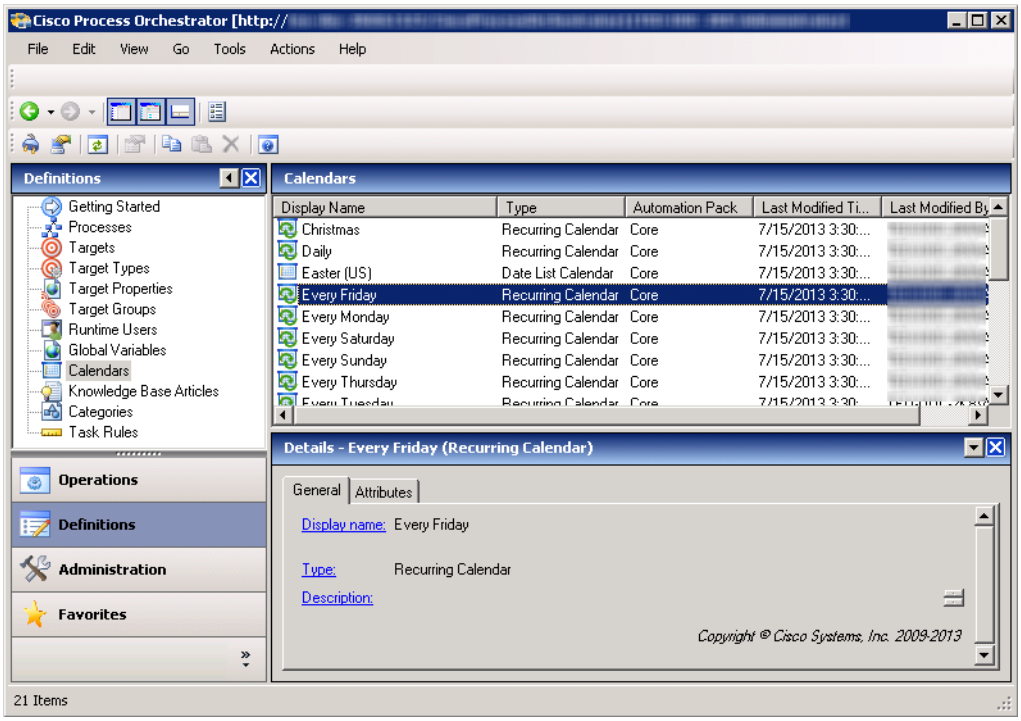
The calendars feature defines the calendar to be associated with a schedule, time, or condition. This feature simplifies:

- Reusing calendar definitions across processes
- Building complex calendars from other calendars
- Viewing the processes that run based on a specific calendar

Cisco Process Orchestrator ships with some pre-defined calendars for the most commonly-used scenarios. However, you can create your own calendar or copy and modify the pre-defined calendars based on your organizational requirements.

Use the Definitions > Calendars view to display the defined calendars.

***Figure 9-1 Definitions > Calendars***




The following table summarizes the types of calendars and when to use them.

***Table 9-1 Calendar Types Summary***

| Calendar Type | Purpose |
|---|---|
| Date List | Specify an explicit list of dates. The processes to which this calendar definition is assigned will execute on the specified dates in the calendar. You might want to use this type of calendar for processes that run on specific days of a specific month. |
| Group | Specify a collection of other defined calendar types, such as inclusion of date list or a recurring calendar within the group calendar definition. The group calendar can contain defined recurring calendars, other group calendars, and date list calendars. You select the calendars to include or exclude in the group calendar definition.<br><br>You can also add dates to or exclude dates from a group calendar. |
| Recurring | Specify a starting date for the subsequent dates and and recurrance frequency. You can specify the calendar to repeat on a daily, weekly, monthly, or yearly basis. |

To create a new calendar:

**Step 1** Choose **Definitions > Calendars**, right-click and choose **New > [Calendar Type]** (see Table 9-1).

**Step 2** On the **[Calendar Type]** property pages, define the properties.

For example, if you want to define a recurring process that runs on the fifth day of each month, enter these values:

- Recurrence Pattern: Monthly
- Start date: (select a day of the month)
- Monthly Recurrence: Every 1 months
- On the following day(s): 5

**Step 3** Click the **Preview** tab to display the list of dates that are included in the calendar. The calendar highlighting the dates displays on the right side of the page.

**Note** For more information about the property fields, see the online help.

**Step 4** Click **OK** to close the dialog box.

# Scheduling a Process Based on a Calendar

To schedule an existing process based on a calendar:

**Step 1** In the Process Editor, choose **Toolbox > Processes**, highlight the name of the process you want to schedule, then choose **Process > Edit**.

**Step 2** In the Process Properties pane, click the **Triggers** tab, then choose **New > Schedule** from the drop-down list.

**Step 3** In the **Schedule Properties** dialog, choose an existing calendar from the **Calendar** drop-down list, or click **New** to create a new calendar (see Authoring a Calendar, page 9-1).

**Step 4** Define any additional required properties. For example:

- To define a recurring process that runs every 5 minutes, enter the **Start Time** (when the process should run on the days included in the selected calendar), the **Number of times to run the process**, and the **Time Interval**.
- To schedule a process to run based on an equation evaluated at run time, see Scheduling a Process to Run Only Under Certain Conditions, page 9-4.

**Note** For more information about the property fields, see the online help.

**Step 5** Click **OK**, then click **Save** to save your changes.

# Scheduling a Process to Run Only Under Certain Conditions

Using the Conditions tab in the Properties dialog, you can schedule a process to run based on an equation evaluated at run time. The process will be run only if the specified conditions are met.

To schedule an existing process to run only under certain conditions:

**Step 1** In the Process Editor, choose **Toolbox > Processes**, highlight the name of the process you want to schedule, then choose **Process > Edit**.

**Step 2** In the Process Properties pane, click the **Triggers** tab, then choose **New > Schedule** from the drop-down list.

**Step 3** Click the **Conditions** tab.

- Use the Basic panel to create a simple variable condition using a variable to match to operator criteria. For example, you can insert a variable reference based on a schedule value such as the schedule name or the local time. Or click **Show advanced properties** to see addition variable references.
- Use the Advanced panel to create a more complex condition. You can define the properties of the conditions within the Advanced panel, as well as within the Properties dialog box.

For more information about using conditions, see Adding Conditions, page 7-22.

**Step 4** Define any additional required properties, then click **OK**.

**Note** For information about the property fields, see the online help.

**Step 5** Click **Save** to save your changes.

# Using a Calendar in a Condition

Calendars are “time” conditions that evaluate to *true* on the days in the specified calendar and between the specified start and end time. For more information about conditions, see Adding Conditions, page 7-22.

To use a calendar in a condition:

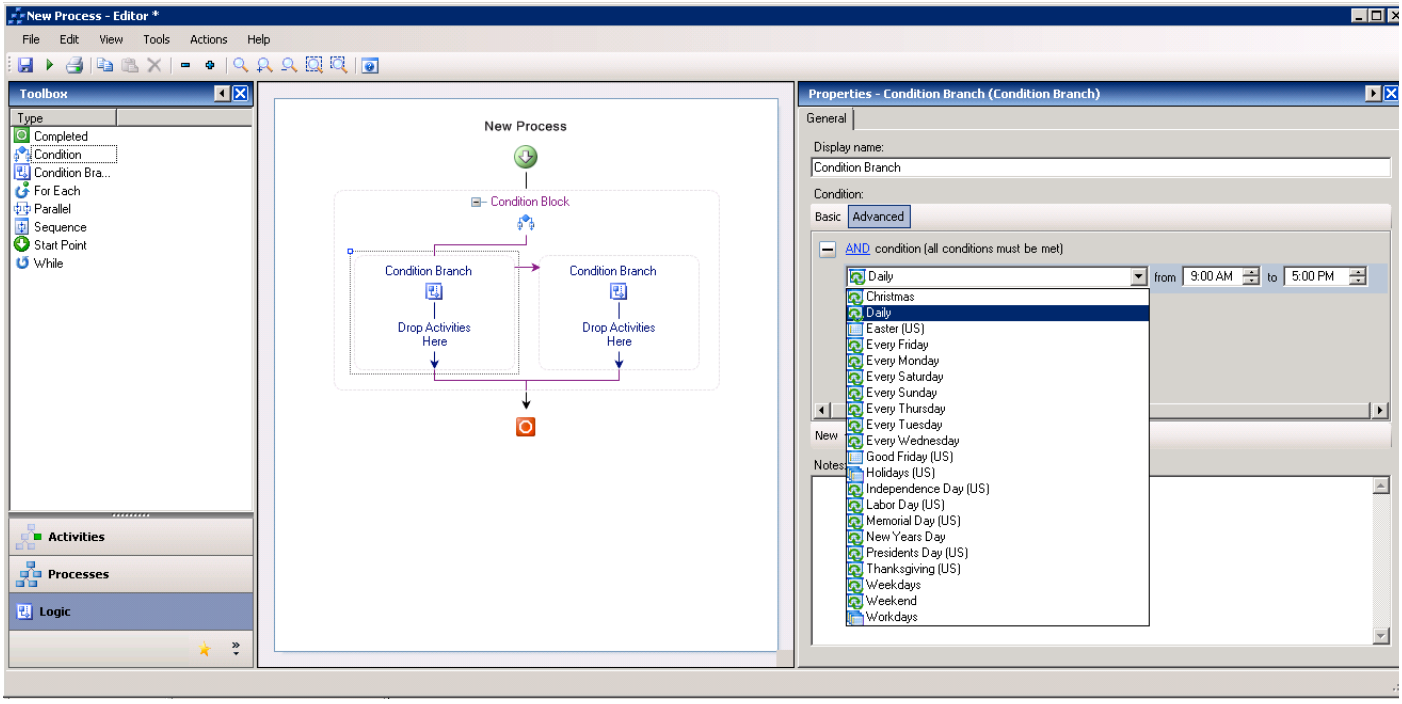
**Step 1** In the Process Editor, you can:

- Add a condition to a process by choosing **Toolbox > Logic**, then drag a **Condition** or a **Condition Branch** into the workflow panel.
- Edit an existing process by choosing **Toolbox > Processes**, highlight the name of the process, then choose **Process > Edit**.

**Step 2** In the Properties panel, click **Advanced > New > Time Condition**.

**Step 3** Choose one of the calendars from the dropdown list and adjust the start and end times as needed.

*Figure 9-2 Using a Calendar in a Condition*



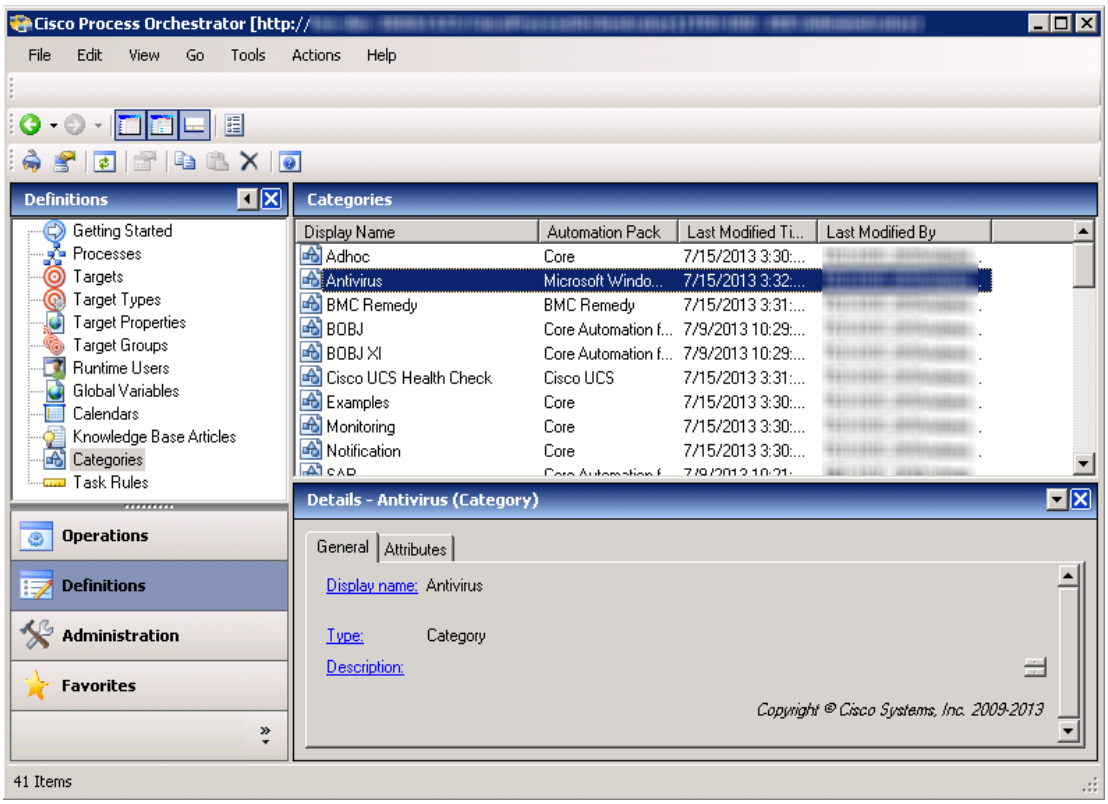

**Step 4** Click **Save** to save your changes.


# Authoring Categories

The Categories feature provides a way to organize your processes based on your organizational or functional requirements. Cisco Process Orchestrator ships with pre-defined categories but provides the functionality for you to create your own business-specific categories. When creating a process, you can assign the process to a category. You can also add other categories to a category to create a hierarchy.

Use the Definitions > Categories view to display the defined categories.

***Figure 10-1 Definitions > Categories View***




The following sections provide information about working with categories:

- Creating Categories, page 10-2
- Adding Objects to a Category, page 10-2

- Viewing Process Properties from the Members Tab, page 10-3

# Creating Categories

Categories can be used to organize your processes based on your business-specific requirements. Use the **Definitions > Categories** view to create a new category and add members to the category. You can also add members to a category when defining a process.

For additional information about defining a process, see Chapter 7, "Authoring Processes."

To create a category:

**Step 1** Choose Definitions > Categories, right-click and choose **New > Category**.

**Step 2** In the **New Category Properties** property sheet, update the properties for this category, then click **OK**.

**Note** For information about the field parameters, see the online help.

# Adding Objects to a Category

You can choose the objects (or create new objects) to be included as members of a category. Both processes and categories can be included as members of a category.

**Note** The category membership of a locked process cannot be edited by other users. For more information about locked processes, see Creating Processes as a Team, page 7-28.

To add members to a category:

**Step 1** Choose **Definitions > Categories**, select a category, right-click and choose **Properties**.

**Step 2** Click the **Members** tab, then click **Add**.

**Step 3** In the Objects available for category membership dialog box, use one of the following methods to select objects:

- If the appropriate objects are listed under Select Category Members, choose the appropriate categories or processes to be included in the category, and click **OK**.
- If the appropriate objects are not listed, click one of the following to create a new object, and click **OK**:
  - **New > Categories**—Define the properties of a new category
  - **New > Processes**—Define the properties of a new process.

  The new object displays in the Select Category Members list. Highlight the new object and any other objects, as necessary. The selected items display on the Members tab.

**Step 4** Click **OK** to close the dialog box and complete the procedure.

# Viewing Process Properties from the Members Tab

To view the processes and categories used in a category:

**Step 1** Choose **Definitions > Categories**, select a category, right-click and choose **Properties**.

**Step 2** Click the **Members** tab. The processes and categories in the selected category are displayed.

**Step 3** To view the properties for a process or category in that list, select the object, right-click, and choose **Properties**.


# Authoring Custom Automation Packs

Automation packs are collections of configuration definitions of process definitions, target types, variables, categories, target groups, and other system elements needed to define a set of automated IT processes (see Process Orchestrator System Elements, page 1-16). The out-of-the-box automation definitions in automation packs allow customers to get productive quickly (see Automation Packs, page 1-15).

Automation packs allow users with the appropriate rights to export their own processes and share them with other customers in the community. Automation pack authors have control over how end users can override their automation definitions. The purpose of this control is to improve the ease of:

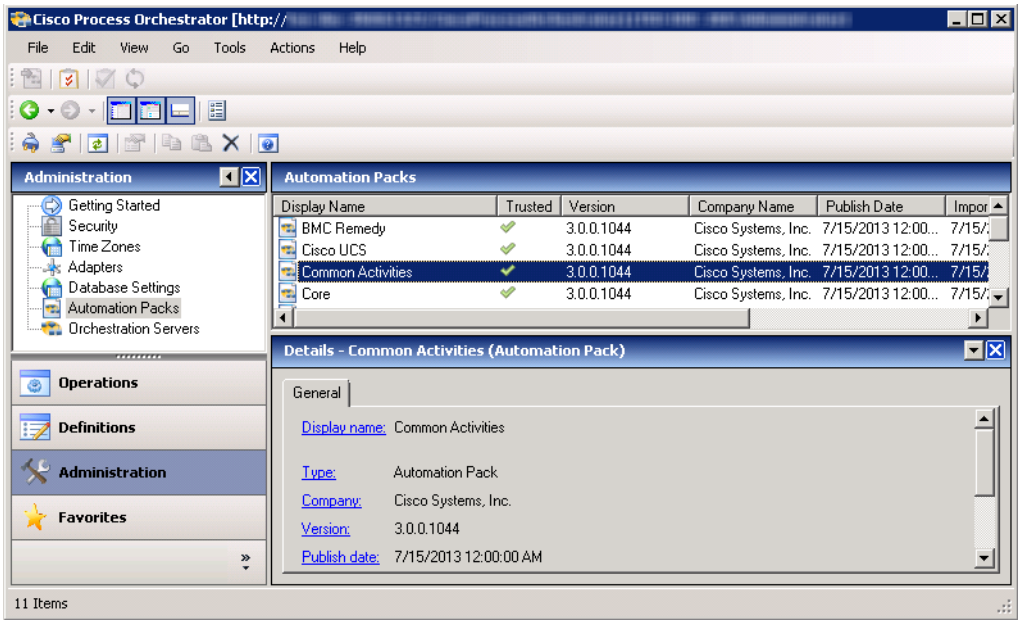
- Consuming future upgrades. Customers naturally expect that some amount of work is involved in upgrades. During an upgrade, all properties customized by users will be preserved. If a workflow is customized, it will be preserved. Process Orchestrator does not merge workflows. If any part of the workflow has been customized, all of the workflow will be left intact (no updates will be made to a customized workflow).
- Consuming patches and hotfixes for content. Customers do not expect these to be difficult or dangerous to consume.

An automation pack is a single file (package) that contains:

- Configuration (processes, categories, knowledge base articles, and so on)
- An optional custom configuration user interface that presents a wizard used to configure essential elements of the automation pack. For example, a configuration UI for SAP performance management might ask the user importing the automation pack to define the SAP systems that Process Orchestrator will manage and provide credentials to reach those systems.
- Report Definition files. These files are extracted into a directory during automation pack import, and can then be imported by the customer into their chosen reporting system
- External management system configuration files such as those to define alerts in a health monitoring application such as Microsoft System Center Operations Manager or HP OpenView SPI Modules.
- Binaries that the customer installs in an external system that the automation pack might require. For example, in Cisco Intelligent Automation for Cloud, Process Orchestrator provides the elements that can be imported into Cloud Portal to order cloud services.

The Administration workspace includes an Automation Pack view that displays the list of automation packs that have been imported. From this view, automation pack authors can create and export their own processes and share them with others in the community.

***Figure 11-1 Administration > Automation Packs View***




This is the typical process you will want to follow when you create an automation pack:

1. Create the basic automation pack—See Creating an Automation Pack, page 11-3
2. Add objects to it—See:
   - Adding Automation Pack Objects, page 11-5
   - Removing Automation Pack Objects, page 11-5
   - Refreshing References in an Automation Pack, page 11-6
   - Viewing Automation Pack Properties, page 11-6
   - Archiving Process Instances in an Automation Pack, page 11-6
3. Verify it—See Validating Automation Packs, page 11-7
4. Modify its settings as necessary—See Controlling the Customization of Automation Pack Elements, page 11-8
5. Review its dependencies—See:
   - Executing a Circular Dependency Check, page 11-8
   - Reviewing Automation Pack Validation Rules, page 11-9
   - Selecting Automation Pack Dependencies, page 11-9
6. Export it—See Exporting an Automation Pack, page 11-9

# Creating an Automation Pack

Process Orchestrator Automation Package files include processes and other objects for Process Orchestrator. The defined automation packs included with the product are located in the install directory on the machine where the product was installed. Individual-created automation packs can be exported and reimported on an as-needed basis.

Each exportable object in Process Orchestrator supports at least two customization settings; you can tune these settings to control what customers can edit (see Controlling the Customization of Automation Pack Elements, page 11-8).

- No—The object is not customizable.
- Limited—Some object properties can be customized. The extent to which properties are customizable varies by object type; the controls for any non-customizable (immutable) property will be read-only. Most objects default to Limited.

Processes have three customization settings:

- No—Users have no process editing capabilities.
- Limited—This is the default setting, under which users can make the following changes:
  - Change the execution and archival option.
  - Change the default target and default runtime user.
  - Change the process input variable value. Users cannot add or create process variables (input or output), only the value of the input variables.
  - Add new triggers to the process.
  - Enable/disable existing triggers. Users cannot remove triggers from the process.
  - Add category (assuming the category customization setting is set to limited).
  - Remove category that is added by users (assuming the category customization setting is set to limited).
- Workflow—Users have near full control to edit processes with this setting. Under the Workflow setting, in addition to the above, users can make the following changes:
  - Edit the process workflow. If the process shipped with some workflow, defined users can delete the existing workflow (activities).
  - Create process local or definition variables.
  - Delete process variables they have created.

  Users cannot:
  - Delete the process
  - Change the input or output variables. Input and output variables are the API or the contract between parent and child process.
  - Change the name, description, target type, or runtime user that the process can run against.

The best practice for authors, then, is to plan extension points. Most processes that an author ships in an automation pack would be locked down. Authors design specific points of extension, and embed calls to processes with workflow customization enabled for these extension points. The workflow for these extension point processes would be empty.

**Note** You cannot export a process from an automation pack you do not own. Therefore, instead of creating your extension in the extension point itself, call a child process where you code the logic. In other words, the contents of an extension point should simply be an invocation of child processes, not activities within the extension point workflow. In this way, you can ship an automation pack with your extension. However, the call to this process must be coded into the extension point in each development, test, and production system.

#### Recommendations

- The functionality authored to extend the solution should be defined in terms of child processes, not activities within the workflow of the extension point.
- It should be exported separately from solution-packaged automation packs
- The functionality authored to extend the solution should be imported separately from solution-packaged content and stitched into the solution extension points.
- By passing target objects rather than specific parameters in to extension points, your automation becomes more extensible from a data perspective as well. For example, someone can come along and add more properties to some target type, and if the target is passed through to the extension point, this data is available.
- If XML is passed into a process request, it is a good idea to make this XML available to extension points as well, so that they are open to extension. Moreover, the child processes in the extension point can modify the target properties.

#### Before You Begin

You must have Process Orchestrator Administrator privileges to create an automation pack.

To create an automation pack:

**Step 1** Choose **Administration > Automation Pack**, right-click and choose **New > Automation Pack**.

**Step 2** In the New Automation Pack Properties property sheet, click the **General** tab, then enter the required information.

**Note** For information about the field parameters for this dialog, see the online help.

**Step 3** If the automation pack file is a community automation pack, click the **About** tab, then check **Community Automation Pack**. This lets anyone modify and export this automation pack.

**Note** The Community Automation Pack checkbox is available only during automation pack creation, and cannot be turned on or off later. Use this feature with caution; it is mostly suitable if the author has no plans for maintaining and releasing new versions, and wants to post this automation pack as an “open source”.

**Step 4** Click the **Objects** tab to add objects to the automation pack.

**Step 5** Click the **Dependencies** tab to review the list of automation packs and adapter information referenced and required by the objects in the automation pack.

**Note** The dependent automation pack and required adapter lists will be updated when the automation pack properties is first opened or when there is a change to the **Objects** page, such as adding or removing an object.

**Step 6** Click the **Style Checker Selection** tab to select the rules you want to validate against an automation pack.

**Step 7** Click **OK** to save your changes.

# Adding Automation Pack Objects

Use this option to add more objects to an automation pack.

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Properties**.

**Step 2** Click the **Objects** tab, then click **Add** to add objects to the automation pack.

**Step 3** In the Select Object dialog box, click the **Filter by Object Type** drop-down list and choose the appropriate object type to filter the list of objects.

**Note** Objects that already belong to an automation pack will be filtered from this dialog box.

**Step 4** Under Select Object, highlight the appropriate objects and click **OK**. Or click **New** to create an object to add to the automation pack.

**Step 5** Click **OK**.

# Removing Automation Pack Objects

Use this option to remove objects from an automation pack. Removing objects from the list of objects in the automation pack does not delete the object from the system.

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Properties**.

**Step 2** Click the **Objects** tab, highlight the appropriate object and click **Remove**.

**Step 3** Click **OK**.

# Refreshing References in an Automation Pack

Use the following instructions to update list of objects in the Objects tab.

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Properties**.

**Step 2** Click the **Objects** tab, then click **Refresh References** to update the list of references associated with the object, then click **OK**.

# Viewing Automation Pack Properties

Use the Automation Packs Properties property sheet to view or modify the processes and objects in an automation pack.

To view automation pack properties:

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Properties**.

**Step 2** On the Automation Pack Properties property sheet, click the appropriate tab to review the properties.

# Archiving Process Instances in an Automation Pack

Use the Automation Packs Properties property sheet to archive instances of all processes in an automation pack. When this option is checked, it will override and *force* an archive, regardless of the individual process archival setting.

For more information about archiving processes, see Archiving Process Instances, page 7-27.

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Properties**.

**Step 2** Click the **General** tab, click **Always archive all process instances in this automation pack**, then click **OK**.

# Validating Automation Packs

Automation pack authors are not always aware of best practices for creating content. In fact, the challenge is that the further removed one is from the product team, the less awareness there is of best practices.

Style checkers are sets of validation rules that run against an automation pack and its member objects to ensure that the automaton pack follows best practices. Some of the style checkers can be run against a process under development in the Process Editor. Some of the benefits that style checkers provide include:

- Automating best practice checks that QA performs on packaged content, including:
  - Ensuring that the content follows best practice guidelines.
  - Identifying issues with automation pack objects (broken references, missing required values).
- Raising the quality of packaged automation and automation created by Cisco Services, delivery partners, or customers.
- Tuning preferred checks per automation pack. Style checkers can be turned on or off based on need. For example, automation packs developed for in-house use might want to turn off some style checkers (such as target, or users included in automation pack).
- Embedded help for identified issues. Help topics notify users why a style check is added, and what the consequences might be if the warning is not resolved.

To run a style checker against an automation pack:

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Properties**.

**Step 2** Click the **Style Checker Selection** tab to select the rules you want to validate against an automation pack (see Controlling the Customization of Automation Pack Elements, page 11-8), then click **OK**.

**Step 3** To start the validation process, you can either:

- Right-click on the automation pack and choose **Validate** from the context menu.
- On the Process Editor, choose **File > Validate** to display the Style Checkers Selection dialog box.

Note that:

- The automation pack validation action can be run on-demand or at export.
- This menu item is disabled if you click an automation pack that you did not create.
- Validation of an automation pack could take several minutes.

**Step 4** Click **OK**. The process validator will start processing the selected validation rules against the current process.

# Controlling the Customization of Automation Pack Elements

Use the Settings dialog box to define the customization settings for the objects included in an automation pack. These settings can only be modified by the automation pack author.

**Note** This functionality (the Settings dialog box and Customization settings) is not available on "Community" automation packs; anyone can modify objects in those types of automation packs. For more information, see Creating an Automation Pack, page 11-3.

When the automation pack author imports a newer version of the automation pack, the objects in the automation pack will replace the objects in the database.

When a third-party user imports a newer version of the automation pack, the customization setting of an object will determine whether user-customized values will be preserved.

The following are instances where an object customization will be lost:

- The author of the object no longer allows users to customize the object. Any customization of the object will be lost on upgrade.

  For example, the global variable customization level is set to *Limited*. Users change the variable value. However, in the updated automation pack, the author no longer allows users to change the variable value, because the automation pack must have a particular value. Therefore, the customization level is now set to *No*.

- The object customization level is unchanged, but the schema of the object is changed. The old object is now obsolete and needs to be replaced.

To change the customization of an object:

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Properties**.

**Step 2** Click the **Objects** tab, highlight the appropriate object, then click **Settings**.

**Step 3** In the Settings dialog box, click the **Customizable** drop-down list and choose the appropriate option, then click **OK**.

# Executing a Circular Dependency Check

Use the Circular Dependency Check command to perform an automation pack circular dependency check on the selected automation pack.

### Before You Begin

- You can only run this test if you are the "author".
- This command relies on the automation pack dependencies and their dependent automation packs dependencies (recursively) to be accurate. The command will not continue if it detects the automation pack or its dependent automation pack is not up to date.

To run this check, choose **Administration > Automation Packs**, highlight the appropriate automation pack (you must be the author), and choose **Circular Dependency Check**.

# Reviewing Automation Pack Validation Rules

Before you export an automation (see Controlling the Customization of Automation Pack Elements, page 11-8), you should review the adapter and automation pack dependencies.

To review automation pack dependencies:

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Properties**.

**Step 2** Click the **Dependencies** tab to view the dependent automation pack and required adapter lists for the automation pack.

The dependencies will be updated when the automation pack properties is first opened or when there is a change to the Objects tab, such as adding or removing an object.

**Step 3** Review the list of automation packs and adapter information referenced and required by the objects in the automation pack (see the Automation Pack Details and Required Adapters tables in Controlling the Customization of Automation Pack Elements, page 11-8), then click **OK**.

# Selecting Automation Pack Dependencies

To select automation pack validation rules:

**Step 1** Choose **Administration > Automation Pack**, highlight the appropriate automation pack, right-click and choose **Properties**.

**Step 2** Click the **Style Checker Selection** tab to select the rules you want to validate against an automation pack (see Controlling the Customization of Automation Pack Elements, page 11-8).

**Step 3** Click **OK** to save your changes.

# Exporting an Automation Pack

The automation definitions included in Process Orchestrator Automation Packs can be exported. Exporting automation definitions allows them to be moved from one environment to another, creates a backup of the automation definition, and allows automation to be shared with another organization.

**Note** Third-party automation packs cannot be exported. You can only export self-authored or community automation packs.

Use the Automation Pack Export Wizard to export an automation pack file to a designated file path on your computer.

To export an automation pack:

**Step 1** Choose **Administration > Automation Packs**, highlight the appropriate automation pack, right-click and choose **Export**.

**Step 2** On the Welcome to the Automation Pack Export Wizard panel, click **Next**.

**Note** For information about the field parameters for this wizard, see the online help.

**Step 3** On the Objects to Export panel, review the list of objects associated with the process being exported, then click **Next**.

**Step 4** On the Dependencies panel, review the list of automation packs and adapter information referenced and required by the objects in the automation pack, then click **Next**.

**Step 5** On the Export Location panel, in the File name text box, specify the name of the file and where the automation pack will be exported onto the computer, then click **Next**.

**Step 6** On the Validate Objects panel, review the following information about the objects being validated, then click **Next**.

**Step 7** Click **Close** to complete the automation pack export process.


# Advanced Authoring Concepts

Cisco Process Orchestrator provides integrations with various technologies. Although it supports numerous types of integrations, at the core Process Orchestrator provides the individual steps that make up the workflows in processes; these are called *activities*. For example, Process Orchestrator provides:

- A set of String activities that you can use to manipulate string text and characters.
- A set of Table activities that you can use to modify the format for existing defined table variables. The available table variables can be defined as either a global or process variable.
- A Set Variable activity that you can use to place a collection of bits in a Process Orchestrator number.
- Knowledge base articles provide information to help understand the results of an activity, including a summary of what has occurred, the possible cause of the result, and suggested actions to take to resolve issues with an activity.

Note that:

- Modifying a process does not automatically modify an activity. Activity definitions are included in a process workflow and the activity properties must be modified separately from the process properties.
- Activities can only be modified in the Process Editor. With the appropriate rights from the Operations view, the Process Editor is launched when accessing the process properties.
- When user rights are restricted, the Process Viewer is launched with the properties displaying a display-only view after determining that the user cannot edit the activity.

The following topics describe these activities:



### Related Topics

For instructions about inserting a variable in an activity, see Inserting Variable References, page 7-17.

# Using String Activities

You can use String activities to search, replace, and modify string content in the objects within Process Orchestrator. For example, you can:

*Table 12-1 Sample String Activities*

| Activity Type | Purpose |
|---|---|
| Find String | Search for specific content in a string. |
| Split String | Split a string into multiple parts around matches of the given delimeter or delimeters. |
| String Escape | Specify characters that must be escaped within a string so that the string can be inserted into messages or file names. |
| String Lowercase | Lowers the text case in a string |
| String Uppercase | Capitalize the text case in a string |
| Substring | Return part of a string starting with the characters in the start position and ending with the character in the specified end position. For example:<br>Input string = `dictionary`<br>Start position = `4`<br>End position = `6`<br>Dictionary<br>0123456 <- using 0 as the starting point<br>Positions 4 through 6 yield ‘ion’ |
| Trim String | Trim characters from the content in a string. When no character is specified, the activity will trim all leading and trailing whitespace characters, including empty lines at the beginning and at the end. |

### Related Topics

See the Cisco Process Orchestrator online help for a list of:

- Common string escapes
- Common regular expressions
- Regular expression examples
- Commonly-used wildcard special characters
- Comparison operators

# Using Table Activities

You can use Table activities to modify the format for existing defined table variables.

**Note** Any boolean and numeric values that are used by the Table activities in the product are not localized. Numbers use [ . ] for the decimal format regardless of the regional and language options. Boolean values for substitutable boolean must be *true* or *false* regardless of the installed language.

For example, you can:

***Table 12-2 Sample Table Activities***

| Activity Type | Purpose |
|---|---|
| Add Row to Table | Append new rows to a table variable. The row is added to the end of the table.<br>This activity will not work on tables that are outputs of other activities. |
| Analyze Table | Summarize and analyze data in a table variable or property using basic aggregation functions. |
| Highlight Row | Highlight selected rows of a table variable. Highlighted rows can only be displayed in an automation summary. Therefore you will not see the highlighted rows in Process Orchestrator unless you are viewing the automation summary. |
| Select From Table | Query and select rows from a source table using specified criteria. The user can also determine the order of the rows selected as well as limit the number of rows displayed. |
| Set Table Variable | Modify the value of a defined table variable. You must specify a name for the activity, choose the defined variable that you want to modify, and enter a new value for the table variable using a different table variable property.<br>The table variables used must match in their entirety, including the name of columns, number of columns, and the type of columns or the activity will fail. |
| Update Row in Table | Update the selected rows of a table variable. The new rows become part of the variable being modified. You will not see the modified rows in the automation summary, but you can see the rows in the variable property pages in Process Orchestrator.<br>This activity will not work on tables that are outputs of other activities. |
| Remove Row from Table | Remove one or more rows from a table variable based on specified criteria.<br>This activity will not work on tables that are outputs of other activities. |

*Table 12-2 Sample Table Activities (continued)*

| Activity Type | Purpose |
|---|---|
| Read Table from Text | Read a comma-separated value (CSV) string variable and convert the text into a table with a specified set of columns. |
| Read Table from XML | Read an XML snippet and convert it into a table with a specified set of columns. For example:<br>**Row XML Element Name:**<br>`MyRow`<br>**Columns to read:**<br>`Name String`<br>`Age Integer`<br>**Source XML:**<br>`<MyData>`<br>`  <MyRow>`<br>`  <Name>Jeff</Name>`<br>`  <Age>32</Age>`<br>`  </MyRow>`<br>`  <MyRow>`<br>`  <Name>Mark</Name>`<br>`  <Age>31</Age>`<br>`  </MyRow>`<br>`  <MyRow>`<br>`  <Name>Jay</Name>`<br>`  <Age>30</Age>`<br>`  </MyRow>`<br>`</MyData>`<br>This produces a table with two columns (name, age), with three rows. |

### Related Topics

See the online help for:

- A list of table expressions
- An example of the WHERE clause
- Commonly-used wildcard special characters
- A list of comparison operators

# Using Core Activities

Core activities are a set of items that span applications. For example, use the following activities to:

*Table 12-3 Sample Core Activities*

| Activity Type | Purpose |
|---|---|
| Calculate Date | Calculate a new date/time value based on a specified base date/time and adjustments. |
| Format Date | Convert the date and time into a string text format. For example:<br>**Format string**<br>`yyyyMMdd hh:mm:ss tt`<br>**Original Date/Time**<br>`10/23/2009 10:23:00 PM`<br>**Result**<br>`20091023 10:23:00 PM` |
| Correlate Process Event | Check whether a process event occurred within a certain amount of time of another problem (see Creating Correlate Event Activities, page 8-6). |
| Create Automation Summary | Generate an automation summary including data selected activities in a process. To generate the data output, choose the activity, then specify the section in the automation summary in which to output the data.<br>An automation summary is a collection of data summarizing the objects included in the process and the data retrieved by the processing of the of the objects.<br>The share path specified in the Core Functions Adapter properties will be used when viewing the automation summary reports. |
| Find Targets | Among other use cases supported by this activity, you can find targets whose properties have certain values. |
| Insert Event | Define the configuration properties to be used to insert one event into Process Orchestrator Reporting Database. |
| Publish Metric | Define the performance metric properties to be published into the Process Orchestrator Reporting Database and the Windows Management Instrumentation (WMI) provider.<br>The metrics are published under the *root\TEO* name space through the WMI class, *TEO_PerformanceMetric*. For a description of the class properties, see the online help. |
| XPath Query | XPath queries navigate through XML documents and search through the document nodes. Use the XPath Query dialog box to enter the XPath properties to query from the source XML code.<br>For additional information, see:<br>• Adding an XPath Namespace Definition, page 12-6<br>• XPath Query Example, page 12-7 |
| XSL Transform | Apply an XSLT transformation to a specific XML text. XSLT transformation can transform XML into plain text, HTML, or other XML. |

### Related Topics

See the online help for a list of:

- Common regular expressions
- Regular expression examples
- Custom date and time format strings
- Commonly-used wildcard special characters
- Comparison operators

## Adding an XPath Namespace Definition

An XML namespace provides a way to avoid element and attribute name conflicts within an XML document. An XML namespace is uniquely identified by a Uniform Resource Identifier (URI) and is assigned a prefix (unique to the XML document) used for its element and attribute names.

To add an XPath namespace:

**Step 1** In the Process Editor, choose **Toolbox > Core Activities > XPath Query**, then drag and drop the activity onto the Workflow pane.

**Step 2** Choose the **XPath Query** tab, then choose **Namespaces > New**.

**Step 3** In the XPath Namespace Definition dialog, enter the prefix and URI for the new namespace, then click **OK**.

In the following example, the elements prefixed with xdc are associated with a namespace whose name is http://www.xml.com/books, while those prefixed with h are associated with a namespace whose name is http://msdn.microsoft.com/en-us/library/ms950779.aspx.

**Example XML**

```
<h:html xmlns:xdc="http://www.xml.com/books”
        xmlns:h=”http://www.w3.org/HTML/1998/html4”>
 <h:head><h:title>Book Review</h:title></h:head>
 <h:body>
  <xdc:bookreview>
   <xdc:title>XML: A Primer</xdc:title>
   <h:table>
    <h:tr align=”center”>
     <h:td>Author</h:td><h:td>Price</h:td>
     <h:td>Pages</h:td><h:td>Date</h:td></h:tr>
    <h:tr alignment”>
     <h:td><xdc:author>Simon St. Laurent</xdc:author></h:td>
     <h:td><xdc:price>31.98</xdc:price></h:td>
     <h:td><xdc:pages>352</xdc:pages></h:td>
     <h:td><xdc:date>1998/01</xdc:date></h:td>
    </h:tr>
   </h:table>
  </xdc:bookreview>
 </h:body>
</h:html>
```

For additional examples, see http://msdn.microsoft.com/en-us/library/ms950779.aspx.

# XPath Example Syntax

For path expressions that can be used when selecting nodes in a path expression of a XPath query, see the Cisco Process Orchestrator online help.

# XPath Query Example

The following is an example of source XML.

**Example XML**

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<bookstore>
<book>
  <title lang="eng">Harry Potter</title>
  <price>29.99</price>
</book>
<book>
  <title lang="eng">Learning XML</title>
  <price>39.95</price>
</book>
</bookstore>
```

The following example path expressions and related results are based on the preceding source XML.

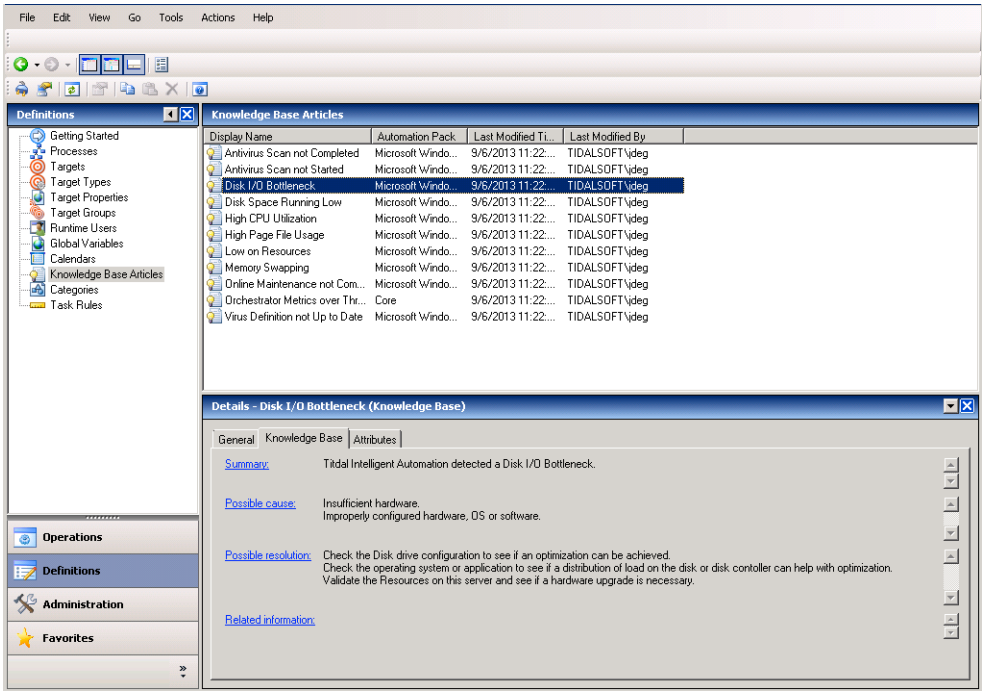
| Path Expression | Description |
|---|---|
| bookstore | Selects all child nodes of the bookstore element |
| /bookstore | Selects the root element bookstore<br>**Note** If the path starts with a slash ( / ) it always represents an absolute path to an element. |
| bookstore/book | Selects all book elements that are children of bookstore |
| //book | Selects all book elements no matter where they are in the document |
| bookstore//book | Selects all book elements that are descendant of the bookstore element, no matter where they are under the bookstore element |
| //@lang | Selects all attributes that are named lang |

For additional examples, see http://www.w3schools.com/xpath/xpath_syntax.asp.

# Authoring Knowledge Base Articles

Knowledge base articles provide information to help understand the results of an activity, including a summary of what has occurred, the possible cause of the result, and suggested actions to take to resolve issues with an activity. Knowledge base articles can make automation summaries more usable, explaining how to use the information that is surfaced.

***Figure 12-1 Sample Knowledge Base Article***

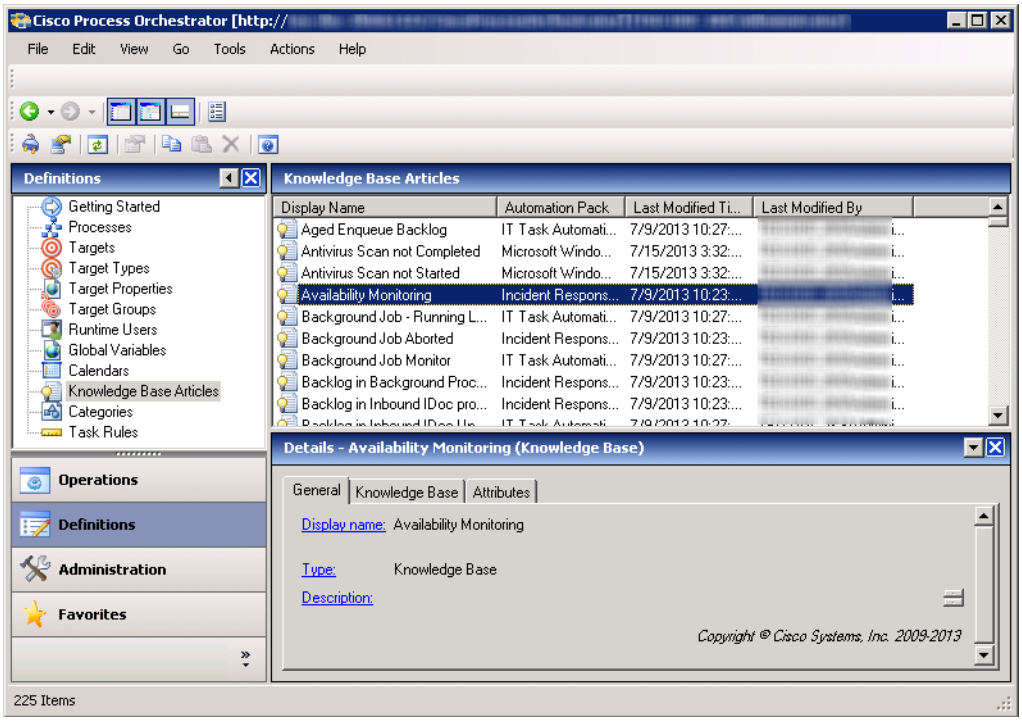



Knowledge base articles are referenced by specific activities and can be viewed in the following locations in Process Orchestrator:

- Operations Workspace—When viewing activity instance properties, you can view the display-only properties of the associated knowledge base articles by selecting the Knowledge Base tab.
- Process Editor—You can reference a knowledge base article to an activity when defining the activity properties in the Process Editor.
- Automation Summary—All referenced knowledge base articles will display on the automation summary.

The Definitions > Knowledge Base view displays the available knowledge base articles that can be assigned to a process activity or to a trigger. When Knowledge Base Articles is selected in the Navigation pane, the defined knowledge base articles display in the Results pane.

***Figure 12-2 Definitions > Knowledge Base Articles***




Process Orchestrator ships with predefined knowledge base articles, or you can create your own knowledge base articles to associate with activities in the process.

To create a knowledge base article:

**Step 1** Choose **Definitions > Knowledge Base**, right-click and choose **New > Knowledge Base**.

**Step 2** On the **Knowledge Base** property pages, define the properties.

**Note** For information about the field parameters, see the online help.

**Step 3** Click **OK** to close the dialog box.

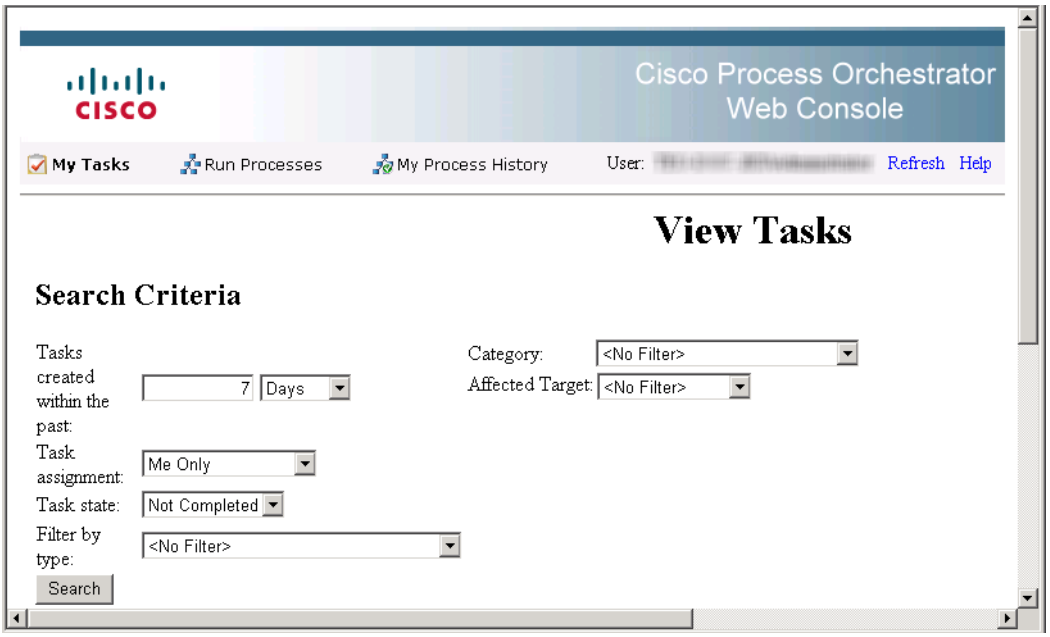

# Using the Web Console

The Process Orchestrator Web Console provides a simplified web-based view to allow end users to interact with automation. The web console does not allow users to define processes but does allow them to start processes, monitor processes they started, or interact with human steps in processes called *tasks* (such as approvals). This interface is purposefully simplified to minimize the learning curve of occasional users.

**Note** The Web Console is typically not used when Process Orchestrator is used in conjunction with Cisco Prime Service Catalog. In these cases, end-user interaction occurs through the catalog rather than Process Orchestrator interfaces. The Web Console is typically only used in cases when Process Orchestrator is used independently.

For more information about the Process Orchestrator Web Console, see Web Console, page 1-12.

***Figure 13-1 Cisco Process Orchestrator Web Console***



For the Web Console to work properly, Internet Information Services must be installed on the computer on which the Web Console is installed. For information about installing and configuring IIS, refer to the *Cisco Process Orchestrator Installation Guide*.

You cannot create or modify processes on the Web Console. To modify a process, you must have the appropriate security rights and must access the Process Orchestrator console. If you think you should have access to the expert Process Orchestrator console, contact your manager to authorize your access.

Use the following sections for instructions on navigating and performing actions in the Web Console:

- Launching the Web Console, page 13-2
- Performing Web Console Actions, page 13-2

# Launching the Web Console

To use the Process Orchestrator Web Console, you must log in using your Windows credentials. If you have Windows integrated authentication configured, then logging in might not be required.

To access the Web Console, choose one of the following:

- Open the Cisco Process Orchestrator console and choose **File > Environment Properties** to see the web console location.
- Choose **Operations > Getting Started > Launch Web Console**.
- Choose **Start > All Programs > Cisco > Cisco Process Orchestrator > Cisco Process Orchestrator Web Console**.

# Performing Web Console Actions

The following sections describe some of the typical actions you can perform using the Web Console.

## Refreshing the Display

The Refresh option updates the information on the current web page.

To refresh the current view, use *one* of the following methods:

- Press **F5**.
- Click the **Refresh** tool.
- On the navigation bar, click **Refresh**.

The display refreshes with the latest information generated from the Process Orchestrator server.

## Viewing the My Tasks Page

Use the My Tasks page to navigate and manage Process Orchestrator tasks in the Web Console.

### Before You Begin

To make sure the default search criteria is available after the initial search on the Process Orchestrator Web Console, enable the cookies in your browser.

To view the My Tasks page:

**Step 1** Launch the Web Console and click **My Tasks**.

**Step 2** Enter the search criteria, then click **Search**. Tasks can be filtered to display:

- Tasks created within a selected time period
- Tasks assigned to the currently logged in user or user group
- Tasks filtered according to the state of the task
- Tasks filtered according to the type of task
- The affected target (for example, the default SMTP server, automation service, or Process Orchestrator server) that the alert or incident describes
- Tasks filtered according to the category assigned to the task

The <No Filter> option allows the page to display all tasks regardless of the assigned state.

**Step 3** To view the task details, highlight the appropriate task, then click **View Details**.

# Resolving Tasks in the Web Console

To resolve a task:

**Step 1** Launch the Web Console, click **My Tasks**, then enter the search criteria to filter the display.

**Step 2** Highlight the appropriate task, then click **View Details**.

**Note** For additional information about the property fields, see the online help.

**Step 3** From the Status drop-down list, choose the appropriate status for the task resolution.

**Step 4** In the Add Notes text box, if available, enter any notes related to the task resolution or status update.

**Step 5** Click **Submit** to resolve the task.

# Taking Ownership of a Task

These steps are used when deciding to take ownership of a task that is unassigned or assigned to an entire group of which the current logged in user is also included.

To take ownership of a task:

**Step 1** Launch the Web Console, click **My Tasks**, then enter the search criteria to filter the display.

**Step 2** Highlight the appropriate task, then click **View Details**.

**Step 3** In the Add Note text box, add notes to the task, as necessary.

**Step 4** Check the **Take sole ownership of this task** check box, then click **Submit**.

# Starting Processes in the Web Console

The Run Processes page displays the processes that the user has access to run, are enabled, and have the Started by User trigger.

To start a process:

**Step 1** Launch the Web Console and click **Run Processes**, then enter the search criteria to filter the display.

To filter the display by category, choose the appropriate category and click **Search**.

**Step 2** On the appropriate process, click **Start**.

**Step 3** If process has input variables, under **Start the process with the following parameters**, verify or modify the values:

| Column | Description |
|---|---|
| Name | Name of the input variable |
| Value | Value assigned to the variable |
| Description | Brief description of the variable |
| Data Type | Type of variable<br>• Boolean—Indicates whether the set of elements is true or false<br>• Hidden String—Holds data that must be protected from other Process Orchestrator users<br>• Identity—Represents the value of a user identity<br>• Numeric—A single whole or decimal number assigned (positive and negative)<br>• String—Sequence of characters, such as letters, numbers, and punctuation marks displayed in a string of text<br>• Table—Used to store a set of records in a table format |

**Step 4** If the process has multiple start points and you want to start a process from a specific point, check the **Start from start point** check box, then select the appropriate starting point. The first activity after the specified starting point will run first.

**Step 5** Under **This process will execute against [Target Name]. Would you like to override?**, choose *one* of the following options:

- Yes—To specify a target or target group different from the defined process target
- No—To continue with the process execution

**Step 6** Click **Start** to start the process; click **Observe** to review the process execution details.

# Overriding the Process Target

With the appropriate rights, you can override the target of any manually-started process before process execution begins. To override the current process target, see Starting Processes in the Web Console, page 13-4, Step 5.

**Note** Targets cannot be modified from the Web Console.

# Viewing My Process History

The My Process History page displays the processes that you have run in Process Orchestrator. You can filter the display of the processes you have launched and the details of the processes.

To view the process history:

**Step 1** Launch the Web Console, click **My Process History**, then enter the search criteria to filter the display. You can filter by:

- Processes that were started in a specified time period.
- Process state—Display all processes in all states (same as <No Filter>), only the processes that have succeeded, or only the process that have failed.
- Category—Filter by the category associated with the processes in the list. The <No Filter> option displays all processes regardless of the assigned category.

**Step 2** Click **Search** to modify the process list display.

**Step 3** On the appropriate process, click **View Details**. The Process Details page displays the summary information for the process.

# Viewing an Automation Summary

An automation summary is a collection of data summarizing the objects included in the process and the data retrieved by the processing of the objects. On the web console, an automation summary can be launched from the following locations:

- Task detail page—Displays the task details of the task
- Process history page—Displays the details of the process activity

To view an automation summary:

**Step 1** Launch the Web Console.

**Step 2** Use *one* of the following methods:

- Choose **My Tasks**, highlight the appropriate task, then click **View Details**. After the task opens, choose **Automation Summary > View**.
- Choose **My Process History**, highlight the appropriate process, click **View Details,** then click **View Automation Summary**.

The Automation Summary page displays the following information.

| Field | Description |
|---|---|
| Situation Analysis | After Process Orchestrator requires action, it performs deep analysis based on the type of situation identified.<br>During situation analysis, Process Orchestrator captures volatile state and diagnostic information that may otherwise be difficult or impossible to capture manually.<br>The *Situation Analysis* section displays below the overview information.<br>Click the link in the upper portion of the summary to navigate directly to the *Situation Analysis* section or simply scroll to the section. |
| Context Analysis | Process Orchestrator analyzes all data points in context with each other to identify a situation that may require action.<br>This information can be viewed in the *Context Analysis* section of the summary. The *Context Analysis* displays the symptom and possible causes. |


# Process Orchestrator Console Components

The following topics describe the main components of the Process Orchestrator console:



# Console Menus

The Process Orchestrator console menus provide general task items, such as feature customization options. These menus are displayed on the Console menu bar.

The following topics describe the contents of each of the Process Orchestrator Console menus.

## Using the File Menu

The File menu contains general actions related to the configuration of Process Orchestrator. The File menu contains the following items:

| Menu Option | Description |
|---|---|
| Connect to Server | Opens the Select Server dialog box where you can specify the server to which to connect.<br>The Server drop-down list displays the list of previously connected servers to the Console. You can also enter the specific server address to be connected. |
| Recent Servers | Displays the most recent servers that have been connected to the Console. |
| Environment Properties | Displays the Environment Properties dialog box for the connected Process Orchestrator server. |

| Menu Option | Description |
|---|---|
| Update License | Launches the Update License Wizard (see Viewing and Modifying Object Properties, page B-2) to enter your product license information. |
| Exit | Closes the Console |

# Using the Edit Menu

The Edit menu contains the following items for modifying the configuration.

**Note** The items enabled under the Edit menu depend on the objects selected in the navigation and detail panes of the Console.

| Menu Option | Description |
|---|---|
| Undo | Reverses the last action by the user |
| Redo | Reinstates the previously reversed action |
| Copy | Copies the selected text or object to the clipboard |
| Paste | Pastes the previously copied item from the clipboard |
| Delete | Deletes the selected item |
| Select All | Selects all items on the dialog box or text field |
| Find | Launches the Find dialog box to locate specific text on a dialog box. |

# Using the View Menu

The View menu contains items that determines how the information displays in the results pane. Select one of the following items to adjust the Console display:

| Menu Option | Description |
|---|---|
| Large Icons | Displays large icons with the associated item |
| Small Icons | Displays small icons with the associated name |
| List | Displays icons with the associated name in list format |
| Details | Displays icons with the associated name and details in list format |
| Choose Columns | Opens the Choose Columns dialog box, which is used to customize the displayed columns and the order in which they display (see Configuring Columns, page B-4). |
| Refresh | Updates the current view |
| Navigation Pane | Enables/disables the Navigation pane on the Console. When disabled, the Results pane and the detail pane, if selected, expand the width of the entire window |
| Detail Pane | Toggles the Details pane *On* or *Off* |

| Menu Option | Description |
|---|---|
| Toolbars | Determines the toolbars (Standard Toolbar, Advanced Toolbar, Actions Toolbar) to display on the Console. The check mark to the left of the toolbar name indicates the toolbar is selected. |
| Status Bar | Enables/disables the status bar that appears at the bottom of the Console |

# Using the Go Menu

The Go menu contains the following shortcuts that allow you to navigate from one view to another on the Console:

| Menu Option | Description |
|---|---|
| Back | Returns to the previously selected view |
| Forward | Option is active if you have clicked Back to access a previous view. Use this option to navigate to the next navigation item in the list. |
| Go To | Submenu displays all previously selected views to which you can return |
| Operations | Opens the Operations workspace |
| Definitions | Opens the Definitions workspace |
| Administration | Opens the Administration workspace |
| Folder List | Opens the Folder List workspace |
| Favorites | Opens the Favorites workspace |

# Using the Tools Menu

The Tools menu contains the following items that configure the appearance of the Console:

| Menu Option | Description |
|---|---|
| Customize | Launches the Customize dialog box. Use this dialog to change the fonts and colors that display in the Console. |
| Options | Displays the Options dialog box. Use this dialog to modify basic server interface functions. |

# Using the Actions Menu

The Actions menu contains items that perform specific actions associated with the selected item in the navigation pane. The available items on the Action menu are dependant upon the selected item on the Navigation pane.

For example, the options available when the Processes view is selected are different from those that are available when Global Variables view is selected.

## Understanding the Help Menu

The Help menu contains the following items:

| Menu Option | Description |
|---|---|
| Help | Displays dialog box-specific information in a separate pane to the left.<br>On the toolbar, click the **Help** tool to display and remove the Help pane. |
| Contents | Displays the Table of Contents tab used to navigate and view the comprehensive help information. |
| Index | Displays the Index tab used to navigate and view the indexed help information. |
| Search | Displays the Search tab used to enter search criteria and locate specific information in the comprehensive help. |
| About Cisco Process Orchestrator | Displays the About Cisco Process Orchestrator dialog box, which contains the product version and license information. |

# Toolbars

The Console has three toolbars (Standard Toolbar, Advanced Toolbar, Actions Toolbar), which can be used to configure the selected navigation view, provide quick navigation between views, and perform item-specific actions.

***Figure A-1 Console Toolbars***


To enable or disable a toolbar, choose **View > Toolbars**, then click the appropriate toolbar.

## Standard Toolbar

The Standard toolbar displays shortcuts to general actions and functions. The following icons appear on every view on the Console.

| Icon | Name/Action |
|---|---|
| | Connect to Server—Connects to the server to which you were last connected |
| | Server Properties—Displays the properties for the connected server |
| | Refresh—Updates the current view with the latest information from the server |

| Icon | Name/Action |
|---|---|
| | Properties—Displays defined properties for the selected object. This icon is unavailable if the current selection does not support property pages. |
| | Cut—Cuts the selected text to the clipboard |
| | Copy—Copies the selected text to the clipboard |
| | Paste—Pastes the selected items to the location identified by the cursor |
| | Delete—Deletes the selected object |
| | Help—Displays or hides the Help pane |

# Advanced Toolbar

The Advanced toolbar displays shortcuts to general navigation and Console configuration icons.

| Icon | Name/Task |
|---|---|
| | Back—Permits you to select the page to be displayed from a drop-down list of the most recent history pages. |
| | Forward—Permits you to select the page to be displayed from a drop-down list of the most recent forward-history pages. |
| | Navigation Pane—Displays or hides the Navigation pane. |
| | Detail Pane—Displays or hides the Details pane. |
| | Status Bar—Displays or hides the Status Bar. |
| | Options—Displays the Options dialog box. |

# Actions Toolbar

The Actions toolbar displays options that are associated with the selected item in the Navigation pane or Results pane. The available items on the Action toolbar depend on the item selected.

For example, the options available when the Processes item is selected are different from those that are available when Global Variables is selected.

| Menu Option | Description |
|---|---|
| Start Process | Click to manually start any enabled process displayed in the Process View. |
| Enable | If a process is manually disabled, click **Enable** to before executing. |

| Menu Option | Description |
|---|---|
| Disable | Click to disable a selected process instance on the process view from executing. |
| | Unlock—If a process is manually locked, click to unlock it. |
| | Lock—If a process is manually unlocked, click to lock it. |
| Advanced | Click this button and select **Break Lock** if a process is locked by another user and you have Break Lock permissions. |

# Workspaces

The Console contains the following workspaces. Each workspace contains a group of objects that perform specific actions within the application.

| Workspace | Description |
|---|---|
| Operations | The Operations workspace displays the status of processes and activities executing, scheduled or recently executed by the application. This workspace also displays alerts, incidents, and approvals in the product. Additionally, the Operations workspace also displays system messages about status and errors within the Process Orchestrator server or adapters.<br>For more information, see Chapter 3, "Monitoring Operations." |
| Definitions | The Definitions workspace displays the components that are used in processes. The options provided by this workspace include:<br>• View Current Processes - Displays the Definitions—Processes view where you can view and manage defined processes.<br>For additional information about monitor and manage processes, see Chapter 3, "Monitoring Operations."<br>• Create a New Process - Launches the Process Editor, which you can use to view and define the properties for a new process as well as construct the process workflow.<br>For additional information about the Process Editor, see Accessing the Process Editor, page 7-2<br>• View Operations - Displays the Operations workspace that shows what processes are in progress, scheduled or have been completed.<br>For additional information about managing operations, see Chapter 3, "Monitoring Operations." |
| Administration | The Administration workspace displays configuration options for Cisco Process Orchestrator.<br>For more information, see Administration, page 4-1. |
| Folder List | The Folder List view displays a list of all the navigation items available from the Console. The items are sorted according to the view displayed on the Console. The Results pane displays the view associated with the selected item. |

| Workspace | Description |
|---|---|
| Favorites | The Favorites workspace displays shortcuts to user-selected navigation items. This view is used for quickly accessing navigation items that are frequently used. Any item in the Navigation pane can be added to the Favorites view.<br><br>To add an item as a favorite, highlight the item, right-click and choose **Add to Favorites**. A shortcut to the item is added to the Favorites workspace. |

# Details Pane

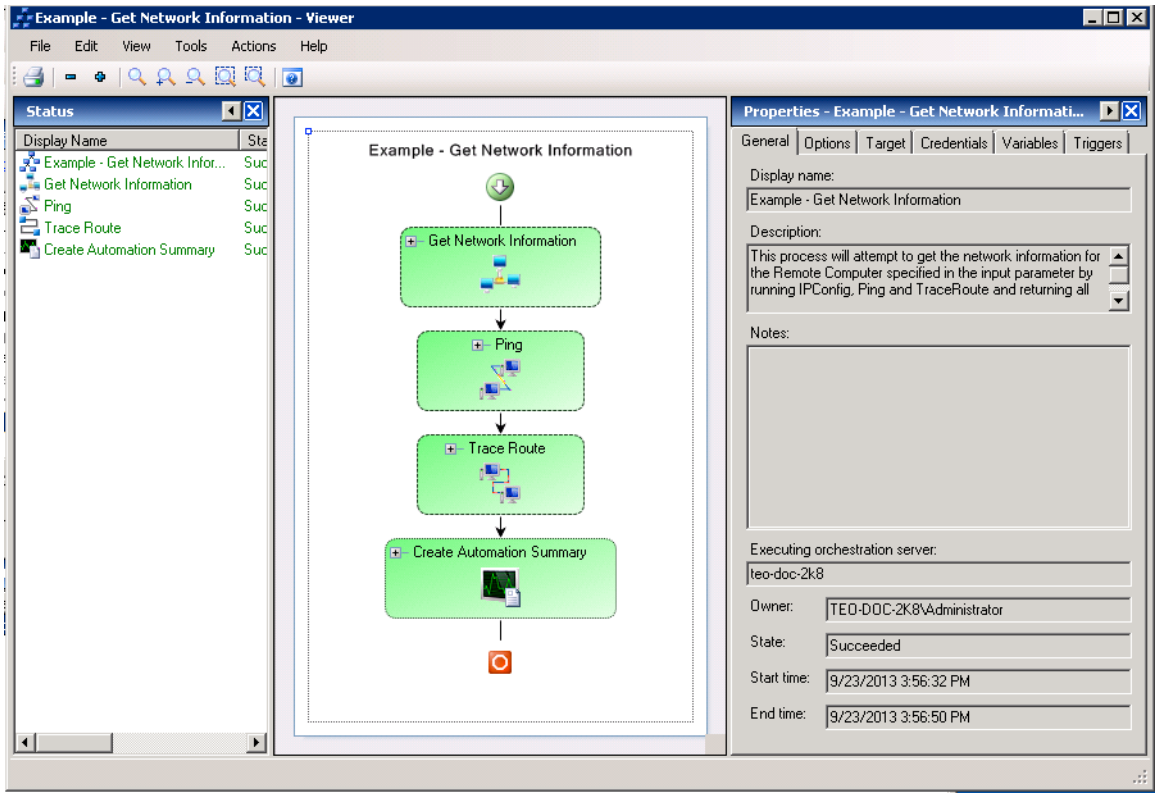
The Details pane displays information about the selected object.

| Tab | Description |
|---|---|
| General | Displays general information about the object including the name, type, value, a brief description, and the ability to enable or disable the object. |
| Attributes | Displays the dates, times and process owner associated with the creation and modification of the object. |

# Process Viewer

The Process Viewer displays a graphical view of the process workflow after a process has been launched. The information displayed on this viewer is read-only.

***Figure A-2 Process Viewer***




The Process Viewer contains this information:

| Pane | Description |
|---|---|
| Status Pane | Displays a flat list of all activity instances that were executed by the process instance. |
| Workflow Pane | Displays the graphical workflow of the selected process and its activities. |
| Properties Pane | Read-only. Displays the properties for the process and activity instances.<br><br>To view the activity instance properties, select the activity instance in the Workflow pane. To return to viewing the process instance properties, click in the white space or use the menu. |

# Process Editor Menus

## Process Editor File Menu

The File menu contains the following items:

| Menu Option | Description |
|---|---|
| Save | Saves the process workflow |
| Save Copy As | Saves a copy of the process workflow under a different name |
| Validate | Verifies whether the process has variables that are actually used in the process |
| Revert | Returns the workflow to the previous saved version |
| Print | Prints the contents of the workflow pane |
| Process Properties | If an activity properties pane is displayed, the option toggles to the process properties pane |
| Exit | Closes the editor |

## Process Editor Edit Menu

The Edit menu contains items to be used for modifying the process in the workflow pane. For information about using this menu, refer to Using the Edit Menu, page A-2.

## Process Editor View Menu

The View menu contains the following items:

| Menu Option | Description |
|---|---|
| Collapse All | Collapses all the activities in the workflow pane |
| Expand All | Expands all the activities in the workflow pane |
| Zoom In | Increases the size of the contents of the workflow pane |
| Zoom Out | Decreases the size of the contents of the workflow pane |
| Zoom Default | Resizes the contents of the workflow pane according to the default zoom value |
| Zoom Fit | Resizes the workflow to fit the size of the pane |
| Zoom Fit Selection | Resizes the selected item in the workflow pane to the size of the pane |
| Toolbar | Displays the Editor toolbar |
| Status Bar | Displays the status bar at the bottom of the Editor |
| Toolbox | Displays the toolbox pane |
| Properties | Displays the process properties pane |

# Process Editor Tools Menu

The Tools menu contains the following items that allow you to configure the appearance and behavior of the editor.

| Menu Option | Description |
|---|---|
| Customize | Launches the Customize dialog box where users can change the fonts and colors that display in the editor.<br>For information about using this dialog box, refer to Customizing the Fonts and Colors, page B-2. |
| Options | Launches the Options dialog box which is used to configure basic Process Orchestrator console behavior. |

# Process Editor Actions Menu

The Actions menu contains the following items that apply to a selected item in the workflow pane:

| Menu Option | Description |
|---|---|
| Move Back | Moves the position of an activity in the workflow back one position at a time. |
| Move Forward | Moves the position of an activity in the workflow forward one position at a time. |
| Add Result Handler | Inserts a Condition Branch into the workflow. A Condition Branch is executed only if the specified condition is met. |
| Expand | Expands the selected activity |
| Collapse | Collapses the selected activity |

# Process Editor Help Menu

The Help menu contains information that provides documentation assistance to work in Process Orchestrator. For information about using this menu, refer to Understanding the Help Menu, page A-4.

# Process Editor Toolbar

The toolbar displays icons to be used as shortcuts for performing actions in the workflow pane. To display the toolbar, choose **Toolbar** from the View menu.

| Icon | Description |
|---|---|
|  | Saves the process |
|  | Starts the process |
|  | Prints the contents of the Workflow pane |

| Icon | Description |
|---|---|
| | Copies the selected information to the clipboard |
| | Pastes the selected information to the location identified by the cursor |
| | Deletes the selected item |
| | Collapses the contents of the workflow pane |
| | Expands the contents of the workflow pane |
| | Resizes the contents of the workflow pane according to the default zoom value |
| | Increases the size of the contents of the workflow pane |
| | Decreases the size of the contents of the workflow pane |
| | Resizes the workflow to fit the size of the pane |
| | Resizes the selected item in the workflow pane to the size of the pane |
| | Displays or hides the Help pane |

# Changing the Process Display Icon

The following steps provide instructions on how to change the icon for the process that displays in the Process Views.

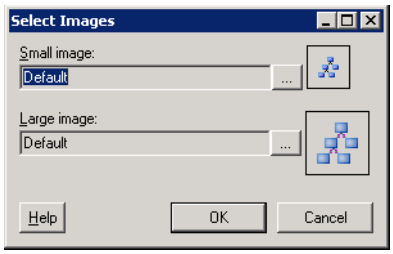
To change the process view icon:

**Step 1** On the Process Properties pane, click the **Options** tab.

**Step 2** Under Display, check the **Use custom icon** check box and click **Select Icon**.

**Step 3** In the Select Images dialog box, the currently selected icons display to the right of the of the fields. To the right of both Image fields (*Small* or *Large*), click **"..."** to change the icon, then click **OK**.

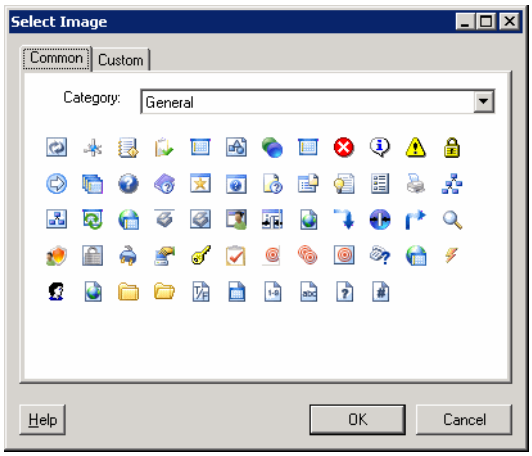
*Figure A-3 Select Images Dialog Box*




**Step 4** In the Select Image dialog box, use *one* of the following options:

- To choose from the display of default icons, on the Common tab, click the appropriate icon and then click **OK**.

*Figure A-4 Select Image Dialog Box—Common Tab*




- To choose an icon from a different location, click the **Custom** tab.
  - To the right of the Image path field, click **"..."** to insert the file path to the appropriate icon.
  - In the Display Name field, enter the name of the icon and click **OK**.

**Step 5** Click **OK**.


# Typical Process Orchestrator Actions

The topics in the following sections describe some of the more typical actions you can perform from the Process Orchestrator console:



## Refreshing the View

The Refresh option updates the information on the current view.

To refresh the current view, choose **View > Refresh**, or click **F5**.

# Viewing and Modifying Object Properties

**Step 1** Click **Definitions > [Object]**, highlight the appropriate object, right-click and choose **Properties**.

**Step 2** On the [Object] Properties property sheet, click the tabs to view and update the various properties for that object.

**Step 3** Click **OK** to close the dialog.

# Configuring the Navigation Pane

The Navigation pane includes views that you use to perform specific functions in the product. You can customize the views that display in the Navigation pane and the order in which they display.

To configure Navigation pane:

**Step 1** In the bottom right corner of the Navigation pane, click the **>>** icon.

**Step 2** Select the option for the appropriate action to be performed:

| Field | Description |
|---|---|
| Show More Buttons | Select this menu item to add buttons (views) to the Navigation pane. This option is available only if all of the available views are not currently displayed. |
| Show Fewer Buttons | Select this menu item to remove buttons (views) from the Navigation pane. This option is available when there are navigation views that can be removed from the Navigation pane. |
| Button Configuration | Select this menu item to display the Button Configuration dialog box, where you can arrange the order of the navigation views. |
| Add or Remove Buttons | Select this menu item to display the list of views that can be displayed or hidden in the Navigation pane. |

# Customizing the Fonts and Colors

The default formatting for the Console display can be updated through the Customize dialog box. The available items to update depend on the category selected.

To customize the product fonts and colors:

**Step 1** Choose **Tools > Customize**.

**Step 2** From the Category drop-down list in the Customize dialog box (see Using the Actions Menu, page A-3), select any of the following items, as necessary:

| Field | Description |
|---|---|
| Console | Updates the display of the Console panel caption |
| Navigator | Updates the display of the Navigation pane |
| Activity | Updates the displayed processes on the Operations—Activity view |
| Calendar | Updates the display of the Calendar dialog box |
| Target | Updates the display of the color status on targets in the Operations workspace. |
| Definition | Updates the color status of the processes in the Definitions workspace |
| Task | Updates the color status of tasks in the Operations—Task view |

**Step 3** Under Display items, select the item to be modified. The available items to update depend on the category selected.

**Step 4** Modify the following items as necessary, then click **OK**.

| Field | Description |
|---|---|
| Font | Provides font and font size display options |
| Foreground | Provides a list of colors to appear on different items in the foreground.<br>Click **Custom** to define a color not displayed in the list. |
| Background | Provides a list of colors to appear on different items in the background.<br>Click **Custom** to define a color not displayed in the list. |
| Gradient Start | Provides a list of colors to appear graded from top the bottom.<br>Click **Custom** to define a color not displayed in the list. |
| Gradient End | Provides list of colors to appear graded from bottom to top.<br>Click **Custom** to define a color not displayed in the list. |

# Configuring the Display Refresh Rate

Use the Refresh tab to specify the refresh rate for the Activity and Process Views.

**Note** Use the scroll buttons to the right of the text fields to change the numeric display one second at a time.

To determine the automatic refresh rate:

**Step 1** Choose **Tools > Options**, then click the **Refresh** tab.

**Step 2** Under Refresh, select the appropriate process and activity instance refresh rate (see Using the Actions Menu, page A-3), then click **OK**.

# Configuring Columns

The Choose Columns dialog box determines which columns display for a selected navigation item and the order in which the columns display. The available column headings vary depending on the selected item.

To configure columns in the Results pane:

**Step 1** In the Navigation pane, select the appropriate item.

**Step 2** In the Results pane, right-click and choose **Choose Columns**.

**Step 3** Under Columns, use one of the following options to select the appropriate column heading to display:

- Check the check box for the column heading.
- Highlight the column heading and click **Show**.

**Step 4** Choose any of the following options:

**a.** To remove a column heading from the display, use one of the following options:

- Uncheck the check box.
- Highlight the column heading and click Hide.

**b.** To change the order in which a column appears in the results pane, select the column heading and click **Move Up** or **Move Down**.

**c.** To determine the width of a column, select the column heading and specify a value in the **Width of selected** column field.

**d.** To reset the column settings to the default, click **Reset**.

**Step 5** Click **OK** to save the settings and close the dialog box.

# Searching for an Object

The Find dialog box locates an item with specific text in the current view..

To find an item on a dialog box:

**Step 1** Choose **Edit > Find**.

**Step 2** In the **Find what** text field, enter specific terms for the search query.

**Step 3** Check the following check boxes, as necessary:

- **Match case**—Selected when searching for term using the specific case in the **Find what** text field
- **Use regular expressions**—Select this item to query items that can contain special characters

**Note** For a list of common regular expressions, see the online help.

**Step 4** Click one of the following buttons to begin the search:

- **Find Previous**—Locates previous match in the search query
- **Find Next**—Locates next match in the search query

# Modifying the Console Time Zones

Use the Date & Times tab to specify the time zone that will be used to display date and time values throughout the application.

To configure the console time zone:

**Step 1** Choose **Tools > Options**, then click the **Date & Times** tab.

**Step 2** Under **Show dates and times using**, choose one of the following options to specify the time zone to be used, then click **OK**.

| Field | Description |
|---|---|
| The current time zone of my computer | Select this radio button to display the date and time zone settings of the computer on which the application is installed. |
| The specified time zone | Select this radio button to select the time zone to be displayed throughout the application. |

# Enabling and Disabling an Object

Most objects are enabled by default. If an object is manually disabled, it must be enabled before it is available. The disabled object is not removed from the list of objects in the Definitions > [Objects] Results pane.

To enable or disable an object:

**Step 1** Choose **Definitions > [Objects]**, highlight the object, then in the Results pane, right-click and choose one of the following:

- **Enable**—The Enabled column on the Results pane changes to *True*.
- **Disable**—The Enabled column on the results pane changes to *False*.

**Step 2** If necessary, click the **Refresh** tool to update the view.

# Deleting an Object

**Note** If the object definition is shipped as a part of product or you do not have the appropriate rights, the Delete option will be disabled.

Use the Definitions > [Object] view to delete an object.

Before deleting an object, access the object's properties and click the **Used By** tab to view where this object is being referenced. This ensures that deleting the object does not affect any processes or activities.

**Notes**

- The deletion process will fail if the object is referenced by other objects.
- A process variable is not completely deleted from a process until after the process has been saved.

# Connecting to a Server

Use this option when you want to connect to a different server used for process monitoring.

To connect to a server:

**Step 1** Choose **File> Connect to Server**.

**Step 2** Choose a previously connected server from the Server drop-down list or enter the server's name, then click **OK**.

# Specifying Server Credentials

The Specify Credentials check box on the Select Server dialog box displays the log in pane used for entering the information to authenticate the server connection.

To specify server credentials:

**Step 1** Choose **File > Connect to Server**.

**Step 2** In the Select Server dialog box, enter the server URL into the Server field or choose the server from the list.

**Step 3** In the Credentials section, choose one of the following options, then click **OK**.:

- Current user

- The following user (enter the user name required to access the server and their password)

# Displaying Recent Servers

The Recent Servers option displays all the servers recently accessed during the current session.

To display the recent servers:

**Step 1** Choose **File > Recent Servers.**

**Step 2** To adjust the number of recent servers displayed:

a. Choose **Tools > Options**.

b. Choose **General > Most recently used servers**.

c. Enter the number of servers to display on the Recent Servers list on the File menu, then click **OK**.

# Suspending Server Automation

Use the Suspend Automation check box to suspend the execution of processes on the server. The enabled check box suspends any new process from being created and the server will no longer create instance of activities or processes.

To suspend server automation

**Step 1** Choose **File > Environment Properties**.

**Step 2** In the Environment Properties dialog box, check the **Suspend automation** check box.

# Viewing Environment Properties

The Environment Properties option provides a view of the properties for the entire high availability Process Orchestrator environment.

To view the environment properties:

**Step 1** Choose **File > Environment Properties**.

**Step 2** For information about the property parameters, see the Cisco Process Orchestrator online help.

# Viewing the Used By Properties of an Object

Use the Used By tab to display the objects that directly reference the selected object in their configuration. Because the Used By tab displays objects in a tree view, you can also display the objects that directly reference the top level objects for the selected object.

The objects at the top level are the objects that reference the selected object directly, but you can expand the listed objects and see their referenced objects. For example, if *object A* is used by *objects X* and *Y* and *object X* is used by *object Q*, then on the property pages of *object A*, you will see *X* and *Y* listed. If you expand the (+) o*bject X*, then *object Q* will display.

To view Used By objects:

**Step 1** Click **Definitions > [Object]**, highlight the appropriate object, right-click and choose **Properties**.

**Step 2** On the [Object] Properties property sheet, click the **Used By** tab to view the objects that use that object.

**Step 3** Click **OK** to close the dialog.

# Viewing the History of an Object

Use the History tab to view a history of changes that have been made to an object. This is a display-only tab that does not require any user interaction.

To view the history of an object:

**Step 1** Click the **History** tab to view the changes made to the object.

The following information about the history of the object is displayed:

| Column | Description |
|---|---|
| Created by | The user name of the person who created the object |
| Created time | The date and time the object was created |
| Time | The date and time the action occurred |
| User | The user name of the person that performed the action |
| Type | The action that occurred |
| Description | Information about the action that was performed |

**Step 2** To view the audit history for a specific action, highlight the appropriate time, right-click and choose **Properties**.

**Step 3** Click **OK** to close the property sheet.


# Advanced Process Orchestrator Actions

The topics in the following sections provide information about managing the server connections for Process Orchestrator. You can launch multiple consoles, change your server connections, and modify the behavior of the server connection upon startup.



**Related Tasks**



## Launching Multiple Consoles with the Same Server

To launch multiple Consoles using the same server:

**Step 1** Choose **Start** > **Programs** > **Cisco** > **Cisco Process Orchestrator**> **Cisco Process Orchestrator Console**.

**Step 2** In the Select Server dialog, enter the server and authentication credentials, then click **OK**.

**Note** For additional information about connecting to a server, see Connecting to a Different Server from the Current Console, page C-10.

**Step 3** Repeat as necessary to open additional consoles connected to the same server.

# Launching Multiple Consoles with Different Servers

To launch multiple Process Orchestrator Consoles with a different server:

**Step 1** After the first console is launched, choose **Start > Programs > Cisco > Cisco Process Orchestrator> Cisco Process Orchestrator Console** to launch an additional Console.

**Step 2** In the Select Server dialog, enter the server and authentication credentials, then click **OK**.

**Step 3** Choose **File > Connect to Server**.

**Step 4** In the Select Server dialog, enter the server and authentication credentials for the new server, then click **OK**.

**Note** If the Console was recently connected to the server, click the arrow to select the server from the drop-down list.

The Console display changes and retrieves the Console configuration for the selected server.

**Step 5** Repeat, as necessary, to open additional Consoles.

# Connecting to a Different Server from the Current Console

Use this process when connecting to a different server while working in an open Console. This option may be useful when wanting to connect to a different server used for process monitoring.

To connect to a server:

**Step 1** Choose **File > Connect to Server**.

**Step 2** In the Select Server dialog, enter the server and authentication credentials for the new server, then click **OK**.

**Note** If the Console was recently connected to the server, click the arrow to choose the server from the drop-down list.

If the server is properly connected, the Cisco Process Orchestrator License Check dialog box displays. If the server does not properly connect, the Connect to Server error dialog box displays.

# Managing Cisco Process Orchestrator Server Connections

The following instructions provide information about managing the server connections for Process Orchestrator. You can launch multiple consoles, change your server connections, and modify the behavior of the server connection upon startup.

## Configuring Server Connection Behavior

When you connect to a server and open the Process Orchestrator Console, you can choose to open the last group and item you were using before you closed the Console, or you can open a specific group and item every time you start the Process Orchestrator Console.

To determine the behavior of the server connection:

**Step 1** Choose **Tools > Options**, then click the **General** tab.

**Step 2** Under Startup behavior, choose *one* of the following functions to occur when connecting to a server, then click **OK**:

- **Automatically select the group and item most recently selected** to open the last group and item selected before the Console was closed.
- **Always select the following group and item** from the dropdown list to specify the group and item to open upon startup.

## Modifying the List of Recent Servers

To adjust the number of recent servers displayed:

**Step 1** Choose **Tools > Options**, then click the **General** tab.

**Step 2** Under **Most recently used servers**, enter the number of servers to display on the Recent Servers list on the File menu, then click **OK**.

# Configuring the Console Global Settings

Use the following sections to configure the global display settings for the Console. This section provides information on how to customize the fonts and colors of the Console, configure the refresh rate for the Operations views, and the default date and time settings.

## Configuring Process Orchestrator Objects Default Owner

By default, the objects that are created in Process Orchestrator use the currently logged in Windows user as the default owner. Use Security tab to specify the currently logged in Windows user as the default user or to specify a different default owner.

To set the Process Orchestrator default owner:

**Step 1** Choose **Tools > Options**, then click the **Security** tab.

**Step 2** Under Default Owner, select the appropriate option to indicate the default ownership setting for objects, then click **OK**.

| Field | Description |
|---|---|
| The current user | Select this radio button to use the user account that is currently logged in to assign the owner on any new objects created. |
| The specified user | Select this radio button, then click **"..."** to launch the Select User or Group dialog box and specify the default user for objects. |

## Modifying Process Launch Settings

Use the Windows and Layout tab to determine the action required when launching a process.

To set the Console behavior when starting a process:

**Step 1** Choose **Tools > Options**, then click the **Windows and Layout** tab.

**Step 2** Under Process Launching, from the When starting a process drop-down list, choose *one* of the following:

| Field | Description |
|---|---|
| Always Launch Immediately with Default Input Value | Launches the process immediately when the process has specified or default values without confirmation. |
| Always Prompt to Confirm | Default option launches Confirm Start Process dialog box to confirm whether the selected process should be launched. |
| Only Prompt when Process has Inputs | Launches the Confirm Start Process dialog box if the process has input variables. If the process does not have input variables, starts the process immediately without confirmation. |

**Step 3** From the **After a process is started** dropdown list, choose *one* of the following options to determine what is displayed after a process begins:

| Field | Description |
|---|---|
| Show Instances | Displays the processes initiated in the Start Process Results popup window. |
| None | Does not display the Start Process Results popup window. |

**Step 4** Click **OK**.

## Restoring Console Default Settings

The Reset User Preferences button on the Windows and Layout tab restores the Console configuration to the default value.

If the window position and size, columns and column sizes, and other configurations in the Console have been modified, this option restores the settings to the Console default settings. The product will automatically close when you reset user preferences.

To restore the default Console configuration:

**Step 1** Choose **Tools > Options**, then click the **Windows and Layout** tab.

**Step 2** Under Saved user preferences, click **Reset User Preferences**, then click **OK**.

After confirming the action, the Console will close automatically.

# Collecting Process Orchestrator Diagnostics

The Cisco Process Orchestrator Diagnostic Utility collects various information about the server environment, installation, as well as any errors or exceptions. The information is compiled into a zip file and saved according to the user preference. The user can then send the information to Cisco Process Orchestrator Technical Support to help diagnose the problem with the Process Orchestrator installation.

**Note** Run this utility only when directed to do so by Customer Support.

The following data is collected by the Utility:

- Diagnostic logs written by the Cisco Process Orchestrator Service, console and other components (*.log files)
- Information about the computer: including memory, processors, OS versions, installed hotfixes and applications (ComputerInfo.txt)
- Information about files installed in the Process Orchestrator installation folder (FileInfo.txt)
- Snapshots of the Windows Event logs (Application, System, as well as Process Orchestrator-specific logs) (.evt files)

- Snapshots of the Process Orchestrator registry key (.reg files)
- Process Orchestrator component configuration files (.config files)

To generate diagnostic logs:

**Step 1** Choose **Start > All Programs > Cisco > Cisco Process Orchestrator > Diagnostic Utility**.

**Step 2** To collect diagnostic information:

a. Click **Collect diagnostic information about Cisco Process Orchestrator**, then click **Next**.

b. On the Information to Collect panel, check the check box of the appropriate logs.

| Field | Description |
|---|---|
| General information | Generates log information from the Process Orchestrator server<br>• Process Orchestrator registry key<br>• Process Orchestrator configuration files<br>• Installed file information |
| Server Information | • Application event log—Log information of application events<br>• Process Orchestrator event log(s)—Log information of the Process Orchestrator server<br>• Server debug logs—Server log information containing debug logs produced by the Process Orchestrator server<br>• System event log—Event information of system logs<br>• Web Console debug logs—Event information of Web Console logs |
| Client Information | • Client debug logs—Contains debug logs produced by the Automation Pack Import Wizard.<br>• Cached client file information—These log files are collected on client-only computers. These files contain information about files that were downloaded by the Console Loader from the server. The information also includes time stamps and version numbers for all files in the loader cache.<br>• Console settings file—Contains debug logs produced by the Process Orchestrator console |

c. Click **Next** to continue.

d. In the Full path to the diagnostics archive file, verify the default file path, then click **Next**.

e. After the information is collected and saved to the appropriate directory, click **Close**.

**Step 3** To review logging settings and manage debug log files:

a. Click **Review logging setting and manage debug log files**, then click **Next**.

b. In the Logging Settings panel, select the log files for the component you want to view (such as client, server, or Java adapter debug logs).

c. To open the folder of log files for a component, click **Open**. Use an ASCII editor to view the contents of a specific log file.

**d.** Click **Finish**.


# Using the PowerShell Snap-in CLI

The Cisco Process Orchestrator Command Line Interface (CLI) Snap-In leverages the Windows PowerShell console to execute scripts using Cisco Process Orchestrator commandlets. If you have the appropriate privileges, you can use the CLI commands to more easily control system administration and accelerate automation. The CLI is included in the standard Process Orchestrator installation and will only execute against the local Process Orchestrator server under the credentials of the current user running the CLI command.

Process Orchestrator PowerShell CLI Snap-in consists the of commandlets that allow you to view details of the existing processes and targets in the Process Orchestrator server. Using the PowerShell CLI Snap-in, you can generate a list of processes and targets, as well as enable and disable specific processes and targets.

To see a list of all the Process Orchestrator commandlets, run the following commandlet in the PowerShell console:

```
get-command *>All Programs>Cisco>Cisco Process Orchestrator>Cisco Process Orchestrator
PowerShell
```

### Related Topics

- For additional information about Microsoft Windows PowerShell, refer to the Windows PowerShell Owner's Manual.
- If PowerShell is not installed, then refer to How to Download Windows PowerShell 1.0 on the Microsoft web site to access the free Windows PowerShell download.


## A

**Active Directory Adapter** Provides access to Active Directory groups, containers and organizational units.

**Active Directory Domain Target** Connection information for an Active Directory domain target.

**Active Directory Group** Target group contains the targets in a group within an Active Directory domain.

**Active Directory OU** Specifies targets that belong to an organizational unit or container within an Active Directory domain.

**Adjustment** Number of units to increase or decrease the time frame. Enter minus (-) prior to the value in order to decrease or enter plus (+) prior to the value to increase. (ex. -5).

**Administration Workspace** Contains the tools which perform administrative actions within the product.

**Administrators Role** When assigned to this group, you have access to everything in the product. Users can view or modify any definition, process, or setting.

**Affected Configuration Item** Name of the configuration item (IT component) to which the alert pertains.

**Affected Organizations** Organizations that consume the IT service affected by the alert or incident.

**Affected Services** IT Service affected by the alert or incident.

**Affected Target Configuration Item** Name of ITIL configuration item (IT component) which the alert or incident describes.

**Affected Target Configuration Item Key** Key of the affected target configuration item in the CMDB.

**Affected Target Configuration Item Name** Display name for the affected target configuration item.

**Affected Target Configuration Item Source** CMDB Source location for the affected target configuration item.

**Affected Target Configuration Item Type** Type of affected target configuration item in the CMDB.

**Affects Configuration Item** Name of the configuration item (IT component) to which the alert pertains.

**Affects Target Configuration Item** Name of the system on which the condition was detected.

**Alert** Alerts reflect potential problems that a user may want to investigate and possibly diagnose the problem.

**Alert class** Numeric value indicating the class of the alert.

**All Matches** All matches to the regular expression.

**All rows** All rows in the source table variable will be removed. Use a where clause-like criteria to specify which rows from the source table to remove. For where clause examples, see the online help.

**Approval Request** Specifies the message and choices for the assignee who is approving the task.

**Assign Task Rule** Task rules assigns users to a specific type of task.

**Assigned** Indicates task is assigned to a user.

**Assigned to** User name of the person assigned to the task.

**Assignees** User name(s) or group assigned to the task.

**Auditors Role** Security role allows the user assigned to this group, the user to view any definition, such as running processes.

**Automation Pack** Collection of Process Orchestrator configuration objects, such as processes, variables, categories, targets, target groups, and more, all stored in one file.

**Automation Summary** Collection of data summarizing the objects included in the process and the data retrieved by the processing of the of the objects.

**Automation Summary Path** File path for the automation summary.

**Automation summary style sheet** Type of template to be used for the automation summary. By default, the Situation Analysis Report style sheet is selected.

**Automation summary URL** Optional file path for the automation summary report associated with an event.

**Available aggregate SQL functions** Values in the aggregate columns that are computed for each unique combination of values in the key columns. (Sum, Count, Avg, Max, Min, StdDev, Var)

**Avg** Aggregate column containing the average of the values in the source column and is available for source columns of numeric data types (such as Integer and Decimal).

## B

**Boolean Target Property** Target property that indicates whether a set of elements should be true or false.

**Boolean Variable** Variable indicates whether a set of elements should be true or false.

**Bypassed** Status indicates a step within the task operation has been skipped.

## C

**Calculate Date Result** Resolved date and time after adjustment.

**Cancel** Security permission which allows the user to cancel process or activity instances.

**Canceled** Status indicates the task, activity, or process has been canceled by the user.

**Categories** Provides the ability to organize objects within the product, including processes and tasks.

**Change Owner** Security permission allows the user to change the owner of the object. Change ownership permission includes the Read permission and requires the Update permission to change ownership on an object.

**Change Request** Task requests a modification to the configuration of an object or system.

**Change Type** Indicates the type of action that occurred to an object.

**Cisco UCS Manager** Target connection information used to connect to the Cisco UCS Manager instance.

**Cisco UCS Manager Adapter** Provides activities to automate Cisco Unified Computing System Manager actions.

**Class** String or numeric value indicating the class of the alert or incident.

**Closed** Status indicates a task is closed.

**Company name** Name of the organization that owns the automation pack. The company name must be entered exactly as provided by Cisco. This field is case-sensitive.

**Completed** Status that signals an activity is completed and terminates the process. Status also indicates a task is completed and has been either approved or declined.

**Completed Date** Date task was completed.

**Completed Time** Indicates the time period the task was completed (Local Time, Universal Time).

**Condition Block** Logic component which executes two or more condition branches in a process.

**Condition Branch** Logic component creates a branch in the process and executes the branch only if the specified condition is met.

**Configuration Audit Grooming** Indicates the number of days before configuration audit data is deleted from the database.

**Configuration Item** Name of the configuration item (IT component) to which the alert pertains.

**Configuration Item Key** Key of the configuration item in the CMDB.

**Configuration Item Name** Display name for the configuration item.

**Configuration Item Source** CMDB Source location for the configuration item.

**Configuration Item Type** Type of the configuration item in the CMDB.

**Connect to Server** Option launches the Select Server dialog box where you can specify the server to which to connect.

**Context Analysis** Process Orchestrator analyzes all data points in context with each other to identify a situation that may require action. After a situation is analyzed in context with other situations, the symptom and cause is displayed in the Context Analysis section of the automation summary.

**Core Functions Adapter** Provides the functionality objects that is required by the Automation Server.

**Count** Aggregate column containing the count of values in the source column and is available for source columns of all data types.

**Counter name** Counter name of the metric.

**Create** Security permission allows the user to create new objects and specify all properties of the object. Create permission includes Change Owner and Read permissions. This permission is required to perform Copy or Import functions.

**Create Ownership Role** Security group provides users with full control over any objects that they own. All users are assigned to this group.

**Created by** The user name of the person who created the object.

**Created time** The date and time the object was created.

**Custom** Button displays color options to be used to create custom colors for text or backgrounds.

**Customer name** Name of the individual client or department. This field is customer-specified and is not case-sensitive.

**Customizable** Indicates the customization setting for the object in the automation pack.

## D

**Data type** Type of variable being used for the parameter value (Boolean, Hidden String, Identity, Numeric, String, or Table).

**Database name** Name of the report database (default Process OrchestratorReporting).

**Database Type** Database type for the credentials.

**Date List** Calendar contains an explicit list of dates which can be assigned to a process. The processes to which this calendar is assigned will execute on the specified dates in the calendar.

**Days** Gets the value of the current TimeSpan structure expressed in whole and fractional days.

**DB2 Database Target** Connection target information for a DB2 database.

**DB2 Database Adapter** Provides activities to access DB2 database objects and execute SQL queries.

**Definition Role** Security group allows users the ability to view or modify any definition or workflow.

**Definitions workspace** Displays all the components that are used in defining and executing processes.

**Delete** Security permission allows the user to delete objects.

**Description** Brief description of an object, such a process, activity, or event.

**Destination** Path for the FTP site.

**Disabled** Status indicates an object is disabled and is not available for execution.

**Display items** The list of available items that can be modified for the selected category.

**Display name** Name assigned to an object.

**Domain** Windows domain in which the user account resides.

**Due Date** Indicates the time period the task should be resolved (Local Time, Universal Time).

**Duplicate** Indicates whether the alert is a duplicate.

**Duplicate Task ID** Task ID of the duplicated alert.

**Duration** Amount of time taken for the activity to complete.

## E

**Email Account (IMAP) Target** Target connection information to access the IMAP email server.

**Email Account (POP3) Target** Target connection information to the POP3 email server.

**Email Adapter** Provides support for sending email via SMTP and receiving email via POP3 and IMAP email accounts.

**Enabled** Indicates whether an object is enabled (True, False). Objects that are not enabled are unavailable for execution.

**End Time** End time of the activity as it completed or failed. If the instance is in progress, 1/1/0001 12:00:00 AM is displayed.

**Error** Brief description of any error that occurred.

**Event category** Category assigned to the event.

**Executor** Target on which the process or activity is executing.

**Expiration Date** Indicates the time period the task should expire (Local Time, Universal Time).

**Expired** Indicates task due date is expired, but still needs to be completed.

**Expression** Expression associated with the column. The expression is configured based on a column from the source data table.

**Target Property** Allows users to define a specific target property value and provides flexibility for others to set or access the values to be used to customize process behavior.

## F

**Failed (Canceled)** Indicates that the process or activity was canceled manually.

**Failed (Not Completed)** Indicates that the process or activity has failed and did not complete the execution.

**Favorites Workspace** Displays shortcuts to user-selected navigation items. This view is used for quickly accessing navigation items that are frequently used. Any item in the Navigation pane can be added to the Favorites view.

**First Match** First match of the regular expression.

**Folder List** List of all the navigation items available from the Console.

**For Each** Logic component used to add the activities to the process that should be executed one time for each item in the target source.

**Foreground** Selects the foreground color to be used in window titles, captions, cells, or selections.

**Format string** Submitted format of the string to be parsed or formatted.

**Formatted Date** Resolved string text for newly formatted date and/or time.

## G

**Generic (OLEDB) Adapter** Provides activities to access database objects and execute SQL queries.

**Generic Data Source (OLEDB)** Target connection information for a data source such as an Excel spreadsheet or a database not supported by other target types.

**Gradient Start and Gradient End** Displays color options to be used for the background color of window titles, captions, cells, or selections.

**Group Calendar** Collection of other defined calendars that are to be included or excluded in a group calendar.

**Guided Operation** Task details the steps a user should take to complete a task.

## H

**Hidden String Target Property** Encrypted target property value that is protected from other Process Orchestrator users and from auditing operations performed by Process Orchestrator.

**Hidden String Variable** Encrypted variable value that must be protected from other Process Orchestrator users and from auditing operations performed by Process Orchestrator.

**Highlighted Row Count** Number of highlighted rows in the table.

**Hours** Gets the hours component of the time interval represented by the current TimeSpan structure.

## I

**ID** Unique ID number assigned to the object.

**Identity** Represents the value of a user identity.

**Import Date** Date the automation pack was imported into the product.

**In Progress** Status indicates user is working on the task.

**Incident** Task requires an operator to take action in order to resolve an issue.

**Input Request** Task requires input from an individual or group.

**Input string** Original date or time string to be parsed

**Install Date** Date and time the adapter was installed

**Instance name** Name of the instance. Also known as the name of the performance metric.

## K

**Key** Key (unique) column instructing the Analyze Table activity to summarize source table data according to the values in that column.

**Knowledge Base Article** Information used to help understand the results of an activity, including a summary of what has occurred, the possible cause of the result, and suggested actions to take to resolve issues with an activity.

## L

**Last Modified By** User name of the person who last modified the object.

**Last Modified Time** Latest date and time the object was modified

**License code** Product license code as provided by Cisco. The license code must be entered exactly as provided by Cisco. This field is not case-sensitive.

**Licensed** Indicates whether the automation pack is a licensed product of Process Orchestrator.

**Listener** Port listener

## M

**Match Count** Number of matches to the regular expression.

**Max** Aggregate column containing the maximum of the values in the source column and is available for source columns of numeric data types (such as Integer and Decimal).

**Message** Brief message of the event associated with the Message ID.

**Message ID** Number associated with a specific condition on the server for the event.

**Milliseconds** Gets the milliseconds component of the time interval represented by the current TimeSpan structure.

**Min** Aggregate column containing the minimum of the values in the source column and is available for source columns of numeric data types (such as Integer and Decimal).

**Minutes** Gets the minutes component of the time interval represented by the current TimeSpan structure.

**Missed** Indicates the scheduled time for process was missed.

**Money Saved** Cost of manual process execution.

**Most recent state** Current status of a process or activity [Ex. Succeeded, Running, Failed (Canceled) or Failed (Not Completed)].

## N

**Name** Name of an object or configuration item (IT component).

**New** Status indicates a task is new.

**New Row Index** Index number of the new row.

**Normal** Status indicates there are no known problems with this target.

**Not Resolved** Status indicates task is not resolved.

**Notify Task Rule** Task rule adds an entry to the notification list of a task.

**Numeric** Data type that indicates a single whole or decimal number is assigned (positive and negative).

## O

**Object Key** ID for the specific record in the CMDB which contains the configuration item.

**Object name** Object name of the metric.

**Object Source** Name of the CMDB which contains the configuration item.

**OLAP Database Adapter** Provides activities to access OLAP database cubes and execute MDX queries.

**Operations Workspace** Displays the status of processes and activities executing, scheduled or recently executed by the application. This workspace also displays alerts, incidents, and approvals in the product.

**Operators Role** Security role allows users to have full access to all processes. All processes can be viewed, started, and canceled, but cannot be modified.

**Oracle Database Target** Target connection information to an Oracle database.

**Oracle Database Adapter** Provides activities to access Oracle database objects and execute SQL queries.

**Original date** Original date or time of the variable.

**Output format** Displays the output format in the automation summary and activity instance. (HTML, XML, TEXT)

**Owner** Owner of the object. This is typically the user name of the person who initially created the object.

## P

**Parallel** Used to run two or more branches of a process simultaneously.

**Parameter** Any parameters associated with a specific object.

**Password** Password assigned to the user account.

**Past Due** Status indicates the task is past due and still needs to be completed.

**Pending** Status indicates the task is in a user's task list.

**Port Number** Port number of the internet protocol.

**Prefix** Prefix element associated with the namespace URI reference in the attribute value.

**Prerequisites** Displays whether the prerequisites for the adapter for the adapter are installed.

**Preview** Displays the effect of the dialog box changes.

**Principal** Name of the user account or group whose members have been assigned the role within the product.

**Priority** Status indicates the priority of the task (Low, Medium, High).

**Process Database** Process Orchestrator Processes Database displays the properties for the Process Orchestrator performance database.

**Process Instances Grooming** Indicates the older process instances are groomed according to the number of days before completed process instance data is deleted from the database.

**Produced by** User who generated the alert or task.

**Protocol** Internet protocol for transferring data (HTTP, SMTP).

**Public-key Authenticated Admin Runtime User** Credentials required to allow public key authentication and an administrative password to perform privileged operations.

**Publish Date** Date the automation pack was exported to a file.

## R

**Recurring** Indicates the start and end date for a recurring calendar and the number of days in the recurrence cycle. The time period and duration indicates when the process repeats execution.

**Regular Expression** String represents the regular expression to be used in matching.

**Related Task IDs** Task ID of the related task.

**Release Date** Date the adapter is available in the product.

**Remedy Adapter** Provides activities for creating, querying, and updating Remedy incidents.

**Remedy Server Target** Target connection information used to access a Remedy server which is used for processes to run against.

**Removed Row Count** Number of rows removed from the table.

**Report Database** Process Orchestrator Reporting database generates reports for viewing process execution history and to audit process changes. You must be logged in with an account that has administrator privileges on the machine where the report database is being created, whether it is the local machine or a remote machine. The user account must also have Reporting Services privileges to create or modify the reports.

**Reports URL** File path to the report server to where the reports reside.

**Reset to Defaults** Returns the values for the selected Category to the default values.

**Resolved** Status indicates task is resolved.

**Response Time** Time taken for site to respond to ping.

**Result Table** Table containing results generated from an activity.

**Revert** Returns the workflow to the previous saved version.

**Review** Task assigns a document for review.

**Role** The security role to which the principal is assigned.

**Row at the specified index** Index number of the row to highlight or clear highlighting.

**Rows matching specified criteria** Criteria used to indicate when to highlight rows in the table.

**Rows with indices in the specified range** The beginning and end of index tows to be highlighted.

**Running** Status indicates the process is in progress.

**Runtime Admin User** User administrative credentials required to access a Cisco IOS Device.

**Runtime User** Runtime user record stores information about the user security context and passes this information to the adapters for activity execution, event monitoring and some target operations (such as availability monitoring and discovery).

**Runtime User Account** Credentials used for a generic runtime user record consisting of a user name and password pair.

## S

**SAP ABAP Adapter** Provides activities to access SAP monitoring objects and execute SAP ABAP functions.

**SAP BI Warehouse** Represent the target connection to an SAP BI Warehouse server.

**SAP Java Adapter** Provides activities to access SAP monitoring objects and execute SAP Java functions.

**SAP System Target** Target connection and logon information used to execute objects against the SAP system.

**SAP User** Credentials used to assign a SAP user.

**Save Copy As** Saves a copy of the process workflow under a different name.

**Scheduled** Status indicates a process is scheduled to run.

**SCOM 2007 Adapter** Provides access to alerts and performance counters in SCOM 2007.

**SCOM Management Server Target** Target connection to the SCOM management server.

**Seconds** Gets the seconds component of the time interval represented by the current TimeSpan structure.

**Section** Name of the section of the automation summary where the data will be stored.

**Sequence** Used to run one or more activities sequentially.

**Server name** *Display-only*. Name of the database server.

**Server Properties** Displays the Server Properties dialog box for the connected server.

**Set variable to** Updated value of the selected variable.

**Severity** Indicates the severity of an event or task.

**Situation Analysis** After Process Orchestrator puts all data points in context to identify a situation that requires action, it performs deep analysis based on the type of situation identified. After a situation that requires action is identified, the state and diagnostic information is displayed in the Situation Analysis section of the automation summary.

**Size** Displays the available font sizes. The font size is limited to the listed font sizes.

**Sleep interval (in seconds)** Number of seconds in the sleep interval.

**SNMP Adapter** Provides access to SNMP alerts and performance metrics.

**SNMP User** Credentials used for a SNMP runtime user.

**Source XML** Source XML text to query.

**Source XML to transform** Source XML text to be transformed.

**Specify XSL path** Option reads the XSL text from a specific file path.

**SQL Server (Integrated Windows Authentication Credentials)** Uses the credentials of the user logged into Microsoft Windows.

**SQL Server (SQL Authentication)** SQL server credentials to use for creating the database (Login ID and Password).

**SQL Server Database Target** Target connection information for a SQL server database.

**SQL Server Database Adapter** Provides activities to access SQL Server database objects and execute SQL queries.

**Start** Security permission allows the user to start processes ad-hoc.

**Start Point** Logic component indicates the points within a process in which a user can start the process.

**Start Time** Start time of the activity.

**Started By** User ID of the person or trigger who initiated the activity instance.

**State** Current status of the operations activity.

**Status** Indicates the status of an object.

**StDev** Aggregate column containing the statistical standard deviation of values in the source column and is available for source columns of numeric data types (such as Integer and Decimal).

**String target property** Defines a target property containing a string of text.

**String variable** Defines a variable containing a string of text.

**Subject** Name of the event

**Subtract Date** Date/time variable reference to be calculated for the difference in time frame.

**Succeeded** Indicates process or activity has completed successfully.

**Sum** Aggregate column containing the sum of all the values in the source column and is available for source columns of numeric data types (such as Integer and Decimal).

**Summary Table** Summary data from the analyst table results

**Suspend automation** Suspends the connection to the Process Orchestrator server.

## T

**Table** Data type stores a set of records in a table format for a variable or target property.

**Target** Specific environments, such as computers, database connections, or application servers, where activities, triggers and processes can be executed.

**Target CI** Name of the system on which the condition was detected.

**Target Reference** Assigns a variable or target property reference from one target to another target.

**Target Type Group** Contains Process Orchestrator targets of a specified type that satisfy an optional criteria type of target to be included.

**Task Grooming** Grooms completed tasks older than the number of days before completed or expired task data is deleted from the database.

**Task Status** Indicates status of the task. The statuses displayed depend on the currently open task.

**Task URL** URL of the task. This URL can be used to bring up the Web Console for viewing and editing the task properties. For example, this is the URL which could be placed in an email to notify the user of the task, enabling them to connect to the Web Console to view the task.

**Terminal Adapter** Provides support for executing commands and scripts using SSH and Telnet connections.

**Terminal Target** Connection information used to access a SSH or Telnet device used for processes to run against.

**This is a CMDB reference** Indicates that the true source of the CI is in the CMDB, so the configuration item properties reference a CMDB entry.

**Ticks** Number of ticks that represent the value of the current TimeSpan structure.

**Cisco Server Provisioner User** User credentials for the user account that connects to the Cisco Server Provisioner target.

**Time** The date and time an action occurs. Also indicates the current date and time in the local time zone.

**Time Raised** Time the metric is raised.

**Time Saved** Time taken to execute the process manually.

**Total Days** Gets the days component of the time interval represented by the current TimeSpan structure.

**Total Hours** Gets the value of the current TimeSpan structure expressed in whole and fractional hours.

**Total Milliseconds** Gets the value of the current TimeSpan structure expressed in whole and fractional milliseconds.

**Total Minutes** Gets the value of the current TimeSpan structure expressed in whole and fractional minutes.

**Total Seconds** Gets the value of the current TimeSpan structure expressed in whole and fractional seconds.

**Triggers** Name of the trigger associated with the process.

**Type** Describes the type of Process Orchestrator object

**Type Description** Brief description of the object type

## U

**Unix/Linux System** Target connection information used to access the SSH device used for processes to run against.

**Unknown** Indicates the status of the target is unknown.

**Unreachable** Process Orchestrator is experiencing problems connecting to the target and executing activities.

**Update** Security permission allows the user to modify all properties of the object, except the owner. Change Owner is the permission required to change the owner of an object.

**Update License** Option launches the Update License Wizard to enter your product license information.

**Update Task Rule** Task rule specifies the properties to be updated in a task.

**Updated Row Count** Number of rows updated in the table.

**URI** The uniform resource identifier (URI) identifies the attribute value of the namespace name or resource on the internet.

**Use** Security permission allows the user to reference objects. This permission is not available to all objects, but only for those that can be referenced. For example, this permission on a target or target group would allow the user to run activities and processes against the target group.

**Use local time zone** Indicates the date should be formatted using the local time zone of the Process Orchestrator server instead of using Coordinated Universal Time (UTC).

**User** The user name of the person that performs a specific action. Also is the user name assigned to a runtime user account.

## V

**Validate** Verifies whether the process has variables that are actually used in the process.

**Value** Value assigned to a variable or performance metric.

**Var** Adds an aggregate column containing the statistical variance of values in the source column.

**Variable data type** Data type assigned to the selected variable.

**Variable Updated** Displays the selected variable specified in the process definition.

**Variable value after execution** The new value of the variable after the process has been executed.

**Version** Version number of an object, including the adapter, automation pack, or the installed Process Orchestrator server.

**Virtual Groups** Target group includes a collection of any defined target.

**VMware Adapter** Provides users with activities to connect a virtual infrastructure and automate the process of managing their virtual machines and hosts.

**VMware ESX Server Target** Target connection information used access an ESX/ESXi server.

**VMware Virtual Center Server Target** Target connection information used to access a virtual center server.

## W

**Waiting** Task status indicates the task is waiting for more information.

**Web Console location** Displays the URL to the Web Console.

**Web Form XSL File Name** Name of the source XML text file to transform the task XML into HTML for viewing in the Web Console.

**Web Service Adapter** Provides activities to execute Web Services.

**While** Logic component allows the process to execute a sequence of child activities (contained in the While Block) that continues as long as a specified condition is true.

**Windows Adapter** Provides access to Windows activities and objects.

**Windows Computer** Target connection information used to access the Windows computer.

**Windows User** Indicates the credentials for a Windows user.

## X

**XPath query** XML path expression to query.

**XSL Location** File path of the XSLT file that can be accessed by the Process Orchestrator server, as well as the client. This option displays only when Specify XSL path is selected.


## A



## C



## D

## E



## F



## G



## H



## I



## K



## L



## M

## O



## P



## R



## S



## T

## U



## V



## W