

Enterprise Portal Core Concepts

The Enterprise Access Service Portal (EASP) is a stand-alone Web application that runs in a J2EE-compliant Web application container. It has a published API for interacting with the EASP information model.

Topic	Page
Background Information	3-3
Enterprise-Site-Access Hierarchy	3-3
Subscriptions for Enterprises	3-4
Subscription Sessions	3-5
Service Parameters	3-5
Substitutions and the Parameter Acquisition Path	3-6
Power of Substitutions	3-7
Strict Access Control for LDAP Data	3-8
Using the Enterprise API	3-8

The API consists of a set of interfaces that can be called to:

- Authenticate managers
- Navigate among retailers, enterprises, sites and accesses
- Create, delete, activate, and deactivate SSP subscriptions
- Get feedback from the sessions that a subscription generates. This feedback, which comes directly from the SAE managing the session, includes whether the session is really active in the network and the values used for the parameters of the service of the subscription.
- Create, delete, and modify substitutions
- Create, delete, and modify the password and privileges of manager accounts

To demonstrate the features of the enterprise API, a Web application framework, two sample enterprise Web applications, and sample data are included in the distribution of the EASP.

The framework consists of a set of JavaBeans. The JavaBeans process accepts user actions, performs the requested actions on the information model, and aids in rendering HTML pages that represent the state of the part of the information model that the manager is viewing.

The EASP includes two sample enterprise management portals that demonstrate the framework's usage and features:

- The first demonstration portal demonstrates all the functionality provided by the framework with little attention to abstraction and user friendliness.
- The second demonstration portal is an example of what a service provider may deploy to its enterprise customers. It is a realistic portal that exposes only some of the framework's features, but it is done in a very user-friendly way.

The sample directory data for these portals includes mono-site/mono-access enterprises, mono-site/multi-access enterprises, and multi-site enterprises. The sample data also includes example SSP subscriptions. The sample directory data includes parameterized enterprise services as described in the *SDX Product Overview Guide*.

Java documentation for the API and beans is available for service providers and system integrators. Source code for the sample Web applications is available for service providers and system integrators to use in developing EASPs. The framework allows for customization of icons and color schemes.

The sample EASP Web applications are implemented entirely with server-side scripting and require no applets or JavaScript on the client side.

Background Information

For further information regarding enterprise services, refer to the *SDX Product Overview Guide* and the *SDX Administration Guide*. For information about the location of Javadocs and sample Web applications associated with the enterprise portal, see Table 3-1.

Table 3-1 Locations of enterprise Javadocs and sample Web applications

Item	Location on SDX Software CD	Location in Which Item Is Installed
Java documentation for the enterprise API and for the sample enterprise Web applications	UMCentdoc.pkg	opt/UMC/doc/custdoc/ent
Sample enterprise Web applications	UMCent.pkg	<ul style="list-style-type: none"> • opt/UMC/httpd/webapps/ent • opt/UMC/httpd/webapps/tagsEntdemo
Source code for the Java beans that control the /ent Web application	UMCentdoc.pkg	opt/UMC/httpd/webapps/src
Source code for the /tagsEntdemo Web application	UMCent.pkg	.jsp pages of this Web application

This chapter provides the background knowledge required by enterprise portal developers. After understanding these concepts, enterprise portal developers will have sufficient knowledge to begin working with the enterprise portal API to create Web applications that expose the SDX application's enterprise functionality.

Enterprise-Site-Access Hierarchy

Enterprises are represented as entries in the SDX application's LDAP directory. An enterprise is represented by multiple LDAP entries, arranged in an enterprise-site-access hierarchy beneath a retailer.

For example, under a given retailer entry, a company Xyz Inc. will have one enterprise entry. This entry stores information about the company in general. Under the Enterprise entry the company will have a set of site entries. Each site entry stores information about one of the company's physical locations (for example, its Chicago office site, its New York office site). Under each site entry the company will have a set of access entries. Each access entry stores information about one of the site's network access connections (for example, a primary Frame Relay access, a backup DSL access). Note that the site entries are optional; accesses can be placed directly under an enterprise. In other words, enterprises are a collection of accesses, optionally grouped by site.

Subscriptions for Enterprises

Each of the enterprise-site-access entries plays a role similar to a residential subscriber. Each is a type of subscriber. As such, each can be subscribed to SSP services, and those subscriptions can be activated. When an access has a subscription that is activated, service policies are applied to the E-series router interface associated with that access, and accounting data is collected from that interface. When a site has a subscription that is activated, service policies are applied to all the E-series router interfaces associated with all the accesses belonging to the site. When an enterprise has a subscription that is activated, service policies are applied to all the E-series router interfaces associated with all the accesses in all the sites in the enterprise. In other words, each entry in the enterprise-site-access hierarchy effectively inherits the subscriptions of its parent entry.

Subscribing an enterprise or site or access to a service requires adding a subscription entry below the enterprise or site or access in LDAP. This can be done by the enterprise customer through an enterprise Web portal running on any Web server. The enterprise Web portal must call the *subscribe()* method on a subscriber object obtained from the enterprise portal API.

Activating a subscription requires modifying the subscription entry in LDAP. When a subscription entry is marked as active in LDAP, all SAEs will notice this. Each SAE will apply the service policies to all the E-series router interfaces that they manage and that are associated with the relevant accesses. Marking a subscription entry as active can be done by the enterprise customer through an enterprise Web portal running on any Web server. The enterprise Web portal must call the *setActive()* method on a subscription object obtained from the API.

At its core, an enterprise Web portal is thus a Web application that, by invoking the SDX functionality exposed in the enterprise portal API, lets enterprise customers modify LDAP data in order to subscribe to services and to activate the resulting subscriptions.

Subscription Sessions

Activating a subscription that exists under an access will affect only that access (that is, the E-series router interface associated with that access). By contrast, activating a subscription that exists under a site or an enterprise can affect multiple accesses. The enterprise portal API introduces the concept of sessions to manage this relationship. For each subscription, there will exist an associated set of sessions. The set will contain one session for each access that the subscription affects.

Normally, all of the sessions associated with a given subscription will be activated and deactivated simultaneously, when their associated LDAP subscription entry is modified. In reality, however, this may not always occur. One or more session activations may fail because of network problems or other reasons. The enterprise API therefore provides detailed information about every session's current state. For example, the API may report that a session should be active, that it is currently inactive, and that it is inactive because the associated E-series router interface is down.

The enterprise API allows local sessions to be activated and deactivated directly, rather than indirectly through modifications to their subscription. A session is local if the enterprise portal that makes the request to manipulate the session is running on the SAE that is responsible for that session's access (that is, on the SAE that is managing the E-series router interface affected by the session).

Service Parameters

Subscribing to and activating services are only part of the functionality available through the enterprise portal API. An enterprise portal can also expose the power of service parameters.

An enterprise service is, at its core, a set of policies that affect network traffic when they are applied to the E-series router interfaces associated with some subset of an enterprise's accesses. When these service policies are defined by the service provider, they can contain parameters. For example, a service that provides protection against denial-of-service attacks may limit the traffic on a specific port to a specific percentage of the bandwidth available on an E-series router interface. Both the port and the percentage can be expressed as parameters in the service's network policies.

Service parameters allow for some very powerful functionality. For example, they allow the service provider to define a generic service that can be precisely customized for specific enterprises or for specific sites or accesses within an enterprise. This customization can be performed at any

time (even while the service is active) by the enterprise customer through an enterprise Web portal. The Web portal must invoke a method in the enterprise API to provide the value for each parameter.

Substitutions and the Parameter Acquisition Path

For each parameter in a service's policies, a value must be obtained. In the example above, the denial-of-service protection policies have two parameters: port number and bandwidth percentage. Each of those parameters in a service's network policies results in the creation of a variable. The name of the variable is determined by the creator of the policies. Each such variable must have a value assigned to it (unless it already has a default value). The enterprise portal can obtain that value from the enterprise customer. The enterprise portal must then call a method in the API to assign that value to the variable. The API will record this by writing a substitution into an LDAP entry. A substitution is an LDAP entry attribute that, at its simplest, just assigns a value to a variable.

More than one substitution can exist for a given variable. Substitutions for a given variable can exist in any LDAP entry on the acquisition path. The acquisition path is a path through a sequence of LDAP entries. It begins with a most specific entry and ends with a most general entry. When the value for a given variable is specified through substitution attributes in multiple LDAP entries on this path, it is only the most specific entry's substitution that is actually used. The ordering of the LDAP entries in the acquisition path is always the same. Starting from the most specific, they are the:

- 1 SSP subscription entry under the access entry (if one exists for the service in question)
- 2 Access entry
- 3 SSP subscription entry under the site entry (if one exists for the service in question)
- 4 Site entry
- 5 SSP subscription entry under the enterprise entry (if one exists for the service in question)
- 6 Enterprise entry
- 7 Relevant localized version of the SSP service entry (if one exists)
- 8 SSP service entry

The acquisition path allows values assigned to variables at a more general place in the acquisition path to be overridden by values assigned at a more

specific place in the acquisition path. This method enables an enterprise to subscribe to a given service, to specify values for that service's parameters at a more general place in the acquisition path, and then to override those values at a more specific level according to the needs of local enterprise IT managers who control a given site or access.



Note: *Each session of a subscription uses a different acquisition path (since each is associated with a different access). This means that each session of a subscription may end up with different values for a given service parameter. For each session, the enterprise API exposes detailed information about the actual values used for every service parameter.*

Power of Substitutions

Substitutions allow much more than the assignment of values to the variables that are used as service parameters. A substitution can declare that the value it assigns is fixed. When this is done, substitutions for the same variable that exist in more specific places in the acquisition path are ignored (that is, the fixed value cannot be overridden). More important, a substitution can specify the value for a variable as an expression that includes other variables. A substitution can also introduce new variables. The new variables are then available for use in other substitutions at any more specific point on the acquisition path. Portals that expose these features allow enterprises to define their own way of presenting and managing service parameters. For more detail on service parameters, the acquisition path, and the uses of substitutions, see the *SDX Administration Guide*.

Strict Access Control for LDAP Data

An enterprise portal can use the API to create managers. A manager is the API representation of a person, usually an employee of an enterprise customer, who has permission to make certain modifications to data in a specific subtree of the LDAP directory. The root of the LDAP subtree that a manager can modify is always one of a retailer, enterprise, site, or access entry. The manager can modify that entry and everything below it, but nothing else. The modifications that a manager can perform are:

- Subscribe to and unsubscribe from services
- Modify substitutions
- Activate and deactivate subscriptions
- Create and delete other managers

A manager can be granted any subset of these powers. Normally, the service provider will create one manager for each enterprise customer, and that enterprise manager will use an enterprise Web portal to create other managers and delegate to them responsibility for specific sites and accesses. (This could include the responsibility for specifying values for variables created by higher-level managers controlling more general parts of the acquisition path.)

Using the Enterprise API

Before invoking any API functionality, an enterprise portal must demand a username and password that identify a manager, and pass them to the API. The API will return a retailer, enterprise, site, or access object that reflects the manager's scope of control. The returned object will be bound to the LDAP directory using the supplied credentials, and the LDAP directory itself will ultimately enforce the manager's access controls. All subsequent operations by the manager will be performed through the returned retailer, enterprise, site, or access object.