Introduction to FNS Policies 3

This chapter introduces FNS policies.

Policy Overview	*page 26*
Examples of Composite Names	*page 29*
How FNS Policies Relate to NIS+	*page 30*
Target Client Applications of FNS Policies	*page 32*

XFN defines policies for naming objects in the federated namespace. The goals of these policies are

- To allow easy and uniform composition of names
- To promote coherence in naming across applications and services
- To provide a simple, yet sufficiently rich, set of policies so that applications need not invent and implement ad hoc policies for specific environments
- To enhance an application's portability
- To promote cross-platform interoperability in heterogeneous computing environments

FNS policies contain all the XFN policies as well as extensions for the Solaris environment.

3

Policy Overview

Computing environments now offer worldwide scope and a large range of services. Users expect to have access to services at every level of the computing environment. Figure 3-1 on page 27 shows that FNS policies provide a common framework for the three levels of services: global, enterprise, and application.

What FNS Policies Specify

FNS provides applications with a set of policies on how name services are arranged and used at the enterprise level:

- Name services for enterprise objects: organizations, hosts, users, sites, and services. (these name services support contexts that allow other objects to be named relative to these objects)
- The relationships among the organization, host, user, site, and service name services, and the names used to refer to these name services
- The syntax of names in these name services
- How to federate the enterprise namespace so that it is accessible in the global namespace
- Names and bindings present in the initial context of every process

What FNS Policies Do Not Specify

The FNS policies do not specify the specific names used within name services. In addition, naming within the application is left to individual applications or groups of related applications.

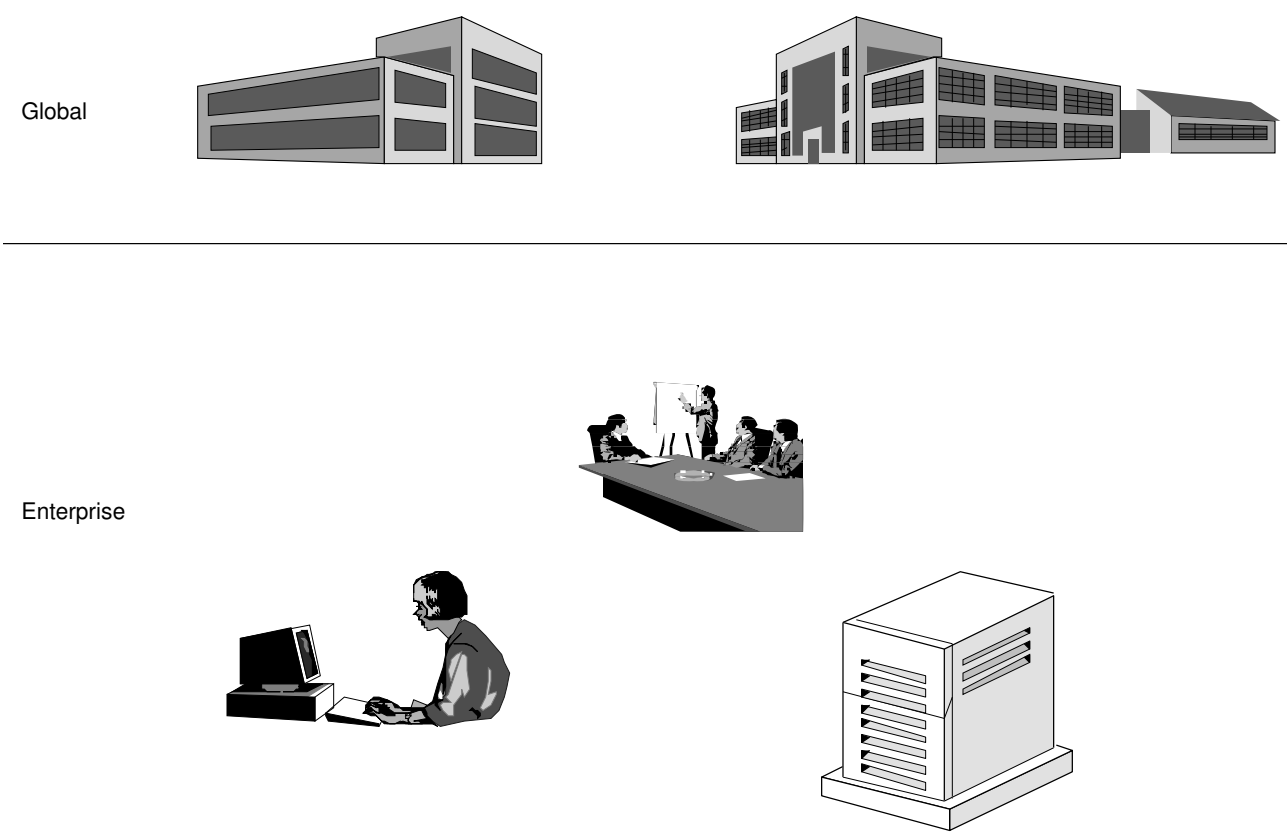

Figure 3-1 Different Levels of Name Services

3

What FNS Enterprise Policies Arrange

The FNS enterprise policies deal with the arrangement of objects within the enterprise namespace. This section introduces the types of objects named within an enterprise and how they are arranged, as shown in Figure 3-2. These entities are described in greater detail in Chapter 4, "Policies for the Enterprise Namespace." The policies are summarized in Table 4-2 on page 43.

- **Organization** — Entities such as departments, centers, and divisions. Sites, hosts, users, and services can be named relative to an organization. The XFN term for organization is *organizational unit*.
- **Site** — Physical locations, such as buildings, machines in buildings, and conference rooms within buildings. Sites may have files and services associated with them.
- **Host** — Computers. Hosts may have files and services associated with them.
- **User** — Human users. Users may have files and services associated with them.
- **Service** — Services such as printers, faxes, mail, and electronic calendars.
- **File** — Files within a file system.

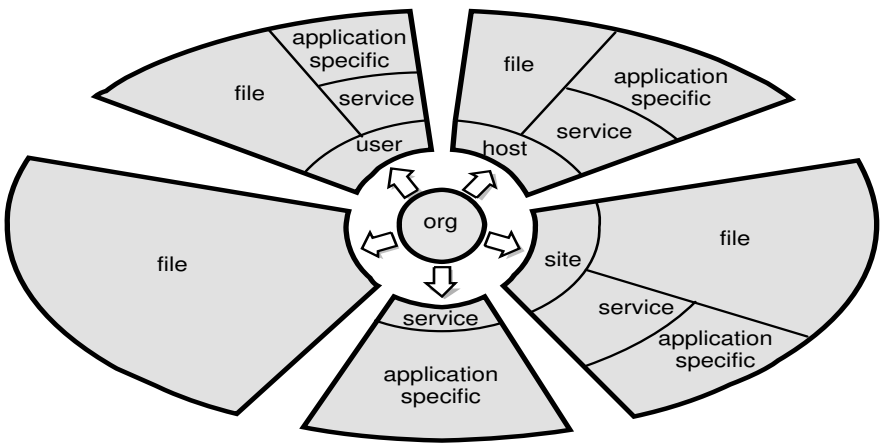

Figure 3-2 What FNS Policies Arrange

Examples of Composite Names

This section shows examples of names that follow FNS policies. The specific choices of organization names, site names, user names, host names, file names, and service names (such as "calendar" and "printer") are illustrative only; these names are not specified by FNS policy.

Composing Names Relative to Organizations

The naming systems to be found under an organization are: `user`, `host`, `service`, `fs`, and `site`.

`org/csl.parc/site/videoconference.northwing`
: names a conference room `videoconference` located in the north wing of the site associated with the organization `csl.parc`.

`org/csl.parc/user/mjones`
: names a user `mjones` in the organization `csl.parc`.

`org/csl.parc/host/inmail`
: names a machine `inmail` belonging to the organization `csl.parc`.

`org/csl.parc/fs/pub/blue-and-whites/CSL92-124`
: names a file `pub/blue-and-whites/CSL92-124` belonging to the organization `csl.parc`.

`org/csl.parc/service/calendar`
: names the `calendar` service for the organization `csl.parc`. This might manage the meeting schedules for the organization.

Composing Names Relative to Users

The naming systems associated with users are `service` and `fs`.

`user/jsmith/service/calendar`
: names the `calendar` service of the user `jsmith`.

`user/jsmith/fs/bin/games/riddles`
: names the file `bin/games/riddles` under the home directory of the user `jsmith`.

Composing Names Relative to Hosts

The naming systems associated with hosts are `service` and `fs`.

`host/mailhop/service/mailbox`
names the `mailbox` service associated with the machine `mailhop`.

`host/mailhop/fs/pub/saf/archives.91`
names the directory `pub/saf/archives.91` found under the root directory of the file system exported by the machine `mailhop`.

Composing Names Relative to Sites

The naming systems associated with sites are `service` and `fs`.

`site/B5.MountainView/service/printer/speedy`
names a printer `speedy` in the `B5.MountainView` site.

`site/B5.MountainView/fs/usr/dist`
names a file directory `usr/dist` available in the `B5.MountainView` site.

How FNS Policies Relate to NIS+

If you are not familiar with NIS+ and its terminology, refer to *NIS+ and DNS Setup and Configuration Guide.* You will find it helpful to be familiar with the structure of a typical NIS+ environment.

NIS+ Domains and FNS Organizational Units

FNS names organization, user, and host objects within the Solaris enterprise naming system, NIS+. An NIS+ domain is comprised of logical collections of users and machines and information about them, arranged to reflect some form of hierarchical organizational structure within an enterprise.

FNS is implemented on NIS+ by mapping NIS+ domains to FNS organizations. An organizational unit name corresponds to a NIS+ domain name and is identified using the fully qualified form of its NIS+ domain name, or its NIS+ domain name relative to the NIS+ root. The top of the FNS organizational namespace is mapped to the NIS+ root domain and is accessed using the name `org/` from the initial context.

In NIS+, users and hosts have a notion of a *home domain*. It is the primary NIS+ domain that maintains information associated with them. A user or host's home domain can be determined directly using its NIS+ *principal name*. An NIS+ principal name is composed of the atomic user (login) name or the atomic host name and the name of the NIS+ home domain. For example, user `jsmith` with home domain `wiz.com.` has an NIS+ principal name `jsmith.wiz.com.`

A user's NIS+ home domain corresponds to the user's FNS organizational unit. Similarly, a host's home domain corresponds to its FNS organizational unit.

Trailing Dot in Organization Names

The trailing dot in an organization name indicates that the name is a fully qualified NIS+ domain name. Without the trailing dot, the organization name is an NIS+ domain name to be resolved relative to the NIS+ root domain.

For example, if the NIS+ root domain is `wiz.com.`, with subdomains `eng.wiz.com.` and `sales.wiz.com.`, the following pairs of names refer to the same organization:

`org/`	`org/wiz.com.`
`org/eng`	`org/eng.wiz.com.`
`org/sales`	`org/sales.wiz.com.`

The name `org/eng.` (with trailing dot) would not be found, because there is no NIS+ domain with the `eng.` name.

NIS+ Users and FNS Users

Users in the NIS+ namespace are found in the `passwd.org_dir` table of a domain. Users in an FNS organization correspond to the users in the `passwd.org_dir` table of the corresponding NIS+ domain. FNS provides a context for each user in the `passwd` table.

NIS+ Hosts and FNS Hosts

Hosts in the NIS+ namespace are found in the `hosts.org_dir` table of a domain. Hosts in an FNS organization correspond to the hosts in the `hosts.org_dir` table of the corresponding NIS+ domain. FNS provides a context for each host in the `hosts` table.

Target Client Applications of FNS Policies

One goal of the FNS policies is to address coherence across the most commonly used tools, including the file system, the DeskSet™ tools, such as Calendar Manager, Print Tool, File Manager, and Mail Tool, and services that support these tools, such as RPC, email, and print subsystems.

Note – Some of these examples are not currently implemented in the Solaris environment. They are listed here as a way of illustrating how FNS may be used.

- **Calendars** — Instead of using names of the form *username@hostname* to access someone's calendar, you would simply supply a user's name, in most cases. You should also be able to use composite names to name calendars. For example, names of the following form would be accepted by calendar manager:

  `jsmith`
  `user/mjones`
  `site/clarke.b5.mtv` (calendar for Clarke conference room)

- **Printing** — Instead of naming a specific printer by its name, you should be able to name a printer relative to a user, site, or organization. For example:

  `jsmith` (`jsmith`'s default printer)
  `org/dct` (an organization's default printer)
  `site/clarke.b5.mtv` (printer in the Clarke conference room)

- **File access** — You should be able to use composite names to name file systems and files. The automounter should use FNS to make resolution of composite names possible. For example, you should be able to use a file name like `/xfn/user/mbrown/fs/.cshrc` to reference the `.cshrc` file for user `mbrown`.

- **RPC** — Instead of addressing services by their host name, program, and version numbers, you should be able to name the service using a composite name. For example, you should be able to name an RPC service relative to a user or an organization.
- **Mail** — Similarly, composite names can be used to name mail destinations. You should be able to use names such as the following:

  `jsmith`
  `user/jsmith`
  `org/dct` (an organization's mailing list)
  `site/clarke.b5.mtv` (mailbox of the conference room coordinator)
- **Other desktop applications** — You should be able to pass composite names to other desktop applications such as spreadsheets, document preparation tools, fax tools, and so on. Some of these applications would attach their own namespace to the service namespace, thus becoming part of the FNS federation.

Example Application: Calendar Service

This is a description of how one application, a calendar service, could be modified to use FNS policies. This example illustrates how FNS composite names might be presented to and accepted from users.

The DeskSet's calendar service is typical of client-server applications. Calendar servers runs on some set of machines and maintain the users' calendars. Calendar Manager (`cm`) runs on the desktop and contacts the appropriate server to obtain the calendars of interest.

The calendar service could benefit from FNS using a simple registry/lookup model as follows:

- **Binding** — Upon startup, the server registers the calendars that it manages by binding a reference containing its own ONC+ RPC address (*host*, *program*, *version*) to the composite name for each calendar it manages, such as `user/jsmith/service/calendar`.
- **Lookup** — When using `cm`, the user specifies another user's calendar simply by entering the user's name (for example, `jsmith`) or selecting it from a list of names previously entered. Given the user name `jsmith`, `cm` composes

the composite name `user/jsmith/service/calendar` and uses this to look up the RPC address that it needs to communicate with the server that manages that calendar.

In the previous example, we used the name "`calendar`" to denote a calendar binding. The developers of the calendar service should register the name "`calendar`" with the FNS administrator, much as RPC programs are registered with the RPC administrator. Refer to "Service Name and Reference Registration" on page 40.

Note – The name "`calendar`" used here is an example. FNS policy does not specify the names of specific services.

The calendar service could take further advantage of FNS policy by allowing calendars to be associated with sites, organizations, and hosts, while still naming them in a uniform way. For example, by allowing calendars to be associated with a conference room (a site), the service can be used to "multibrowse" the conference room's calendar as well as a set of user calendars to find an available time for a meeting in that room. Similarly, calendars can be associated with organizations for group meetings and hosts for keeping maintenance schedules.

`cm` could simplify what the user needs to specify by following some simple steps.

1. `cm` uses a tool for accepting composite names from the user and constructing the name of the object whose calendar is of interest. The object would be the name of a user, a site, a host, or an organization. For example, the user might enter the name `mjones` and the calendar manager would generate the composite name `user/mjones`. This tool could be shared amongst a group of DeskSet applications.
2. `cm` uses the XFN interface to compose this name with the suffix `/service/calendar` to obtain the name of the calendar.

3. This calendar name is then resolved relative to the process's initial context. Continuing with the example, this would result in the resolution of the name `user/mjones/service/calendar`. Similarly, if the user enters the name of a site, `clarke.B5.mtv`, `cm` would generate the name `site/clarke.B5.mtv/service/calendar` for resolution.

Other services such as printing and mail could take advantage of the FNS policies in a similar way.

3