3.3 Release Notes:
The UserManager Application

This file contains release notes for the UserManager application.   Items specific to Release 3.3 are listed first, and notes from previous releases follow.

Notes Specific to Release 3.3

New features

· You can now open multiple windows for individual user accounts or user groups.

· User templates are supported. When you create a new user account, the new user's home directory contents are copied from a template user. One template (named **user**) is provided in the NEXTSTEP Release 3.3 software. You can create additional templates of your own.

A template is actually a local user account. If you create a new template named **engineer**, a new user account named **engineer** is created on the local system. To customize the **engineer** template, you log in as the user named **engineer** and add icons to the dock, create files and subdirectories in this user's home directory, change application preferences, and so on. You

must log out to complete making the changes to the **engineer** template. When you then create users using the **engineer** template, their home directories, workspace setup, Mail messages, and application preferences are copied from the **engineer** user. Subsequent changes to the **engineer** template don't affect these users.

Template user accounts are created in the local NetInfo domain. The home directory for a template is created in **/usr/template/loca**l. Any subdirectory of **/usr/template/local** will be regarded as a user account template, including template directories NFS-mounted from remote systems.

· Extra resources can be added for new user accounts according to the new user's language. When a user's home directory is created, the contents of the language-specific directory **/usr/template/languages/*Language*.lproj** (where *Language* is a language that has been installed on the computer) are copied to the user's home directory. NEXTSTEP Release 3.3 software includes a language-specific **Mailboxes** directory. You can add other language-specific resources to these directories, which are copied to new users' home directories.

· A special file named **#DEFAULTS#** in each language-specific directory is not copied to new home directories. This file contains a set of user defaults that are installed in the user's defaults database. The file contains lines of the form:

```
owner name value
```

These lines are written, one at a time, to the user's defaults database. See the documentation for the **dread** and **dwrite** commands for more information on the defaults system.

· UserManager writes a log of all of its operations to a file. Included in the log are the calling arguments of all program hooks and their return values. Additionally, any output printed by a program hook is also written to the log file. The default log file is **/usr/adm/nu.log**.

· You can now create and delete user accounts in ªbulk.º Several input file formats are supported for bulk user account creation, including files containing just names as well as **/etc/passwd** format files. UserManager doesn't allow duplicate names or UIDs. You can specify several formats for user names as well as starting values for UIDs, home directory

locations, shells, languages, and so on. Passwords can be supplied from an **/etc/passwd** format file, or they can be generated randomly. UserManager displays a table of all account values. You can edit entries in this table and save a copy of the table to a report file before you create the accounts. The report records, in plain text, all randomly generated passwords, so the file should be protected appropriately.

· UserManager offers several options for deleting users in bulk. For example, you can delete all users with UIDs in a certain range or all users in a particular user group. UserManager displays a table of all the user accounts that match your choice. You can selectively remove entries from the table so those accounts aren't deleted. You can also choose to disable the accounts rather than delete them.

· Default user and group parameter values can be assigned for each NetInfo domain. UserManager no longer reads default values and configuration information from the file **/etc/nu.cf**. Defaults are now set for each NetInfo domain. These parameters include the starting value for UIDs, default group GIDs, shell, template, language, and so on.

· Programming hooks are supported for all user and group management operations. You can specify that a custom program run before and/or after any major operationÐsuch as before a user account is added to NetInfo, after a user's home directory is created, or after a user's account is disabled. Your program is called with all pertinent information (for example, the user's name, UID, and default group) passed on the command line. For details, see Chapter 5 of the on-line *Network and System Administration* manual.

It's important to note that program hooks are not required. You only need to write program hooks if you want to customize or override UserManager's operation.

Program hooks are assigned to UserManager through the Defaults panel. Note that program hooks (and other defaults) are set up independently for each NetInfo domain. This means you customize user account management independently for any of your site's NetInfo domains.

Changes and bug fixes

- You can disable and reenable user accounts without resetting passwords. To disable an account, UserManager prepends an asterisk (*) character to the encrypted value of a user's password. The account is enabled by removing the asterisk.

- Group membership can be managed for all parent domains as well as a user's domain. In previous versions of UserManager, group assignments were limited to the domain in which the user's account was defined. You can now assign group memberships in a user's domain and all domains above it in your site's NetInfo hierarchy. The user's default group is marked with an asterisk in the group's browser.

- UserManager no longer reads defaults from **/etc/nu.cf**, nor does it run the shell scripts in **/etc/nulib**. These files are still used by **nu**, which has not been changed for Release 3.3.

Notes Specific to Release 3.2

New Features

The following features have been added to the UserManager application since Release 3.0.

- UserManager sets the default language of the user when the user's home directory is created.

- UserManager is integrated with SimpleNetworkStarter to allow the selection of home directory servers from a pop-up list in the simple entry form.

- UserManager starts adding users after 100 instead of 21.