

## 3.3 Release Notes: Installer

This file contains release notes for both the 3.1 and 3.0 releases of the Installer application (there are no new notes for Release 3.3 or 3.2). Items specific to Release 3.1 are listed first, and the Release 3.0 notes follow.

### Notes Specific to Release 3.1

#### New Features

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

- The ability to create and install multi-architecture packages.

If the source directory contains multi-architecture (<sup>a</sup>fat<sup>o</sup>) files, the package created will automatically reflect this fact. Multi-architecture packages look essentially the same as the <sup>a</sup>non-fat<sup>o</sup> ones, except that the BOM file now is in binary form. This feature is completely

compatible with the 3.0 Installer and its packages, except as noted below.

- Support for the removal of unwanted architectures during installation.

A multi-architecture package can optionally be <sup>a</sup>thinned<sup>o</sup> to a specified set of target architectures at installation time. This will save space on the destination file system, but will mean that the resulting executables only will run on the selected system types. You would have to redo the installation if you later wanted to <sup>a</sup>fatten<sup>o</sup> a thinly installed package. Only packages created using the 3.1 Installer can be thinned during installation; old 3.0 packages can still be installed, but only in their full form.

## Bugs Fixed in Release 3.1

These bugs have been fixed in Release 3.1:

Reference	29686
Problem	chunkPackage: \$destPkg undefined when destdir is <sup>a</sup> . <sup>o</sup>
Description	The chunkPackage script file under Installer.app does not work properly. It splits the package up into separate files, but does not create the separate chunks needed to go on the diskettes.
Reference	30074
Problem	Installer clobbers environment when execing installer_util
Description	It makes its own environment to contain DO cookies, but doesn't propagate all

the rest of the variables. This means no pre or post-install script can access apps if public window server is off.

## Notes Specific to Release 3.0

These notes were included with the Release 3.0 version of the Installer application.

### Scripting

The Release 3.0 Installer lets developers add up to four programs (typically shell scripts) to Installer packages. Two of these programs, the `^pre_install^` and `^post_install^` programs, will be executed before and after the Install and Expand operations if the programs exist inside the Installer package. Similarly the `^pre_delete^` and `^post_delete^` programs, if they exist, will be run before and after Delete and Compress operations. These programs must be written by the developer and can be used for a wide variety of pre- and post-processing tasks.

### Localizable Packages

The Release 3.0 Installer supports localization of Installer packages by letting developers put localized `.info` files, `.tiff` files, and installation scripts inside packages, organized by language-specific subdirectories. When a user opens such a package, the information that is displayed in the Installer Package window is read from the `.info` file and `.tiff` file that are chosen in the standard way based on the user's language preferences. Also, scripts in localized packages are chosen for execution based on the user's current language settings.

## **Remote packages**

The Release 3.0 Installer supports <sup>a</sup>remote packages<sup>o</sup> whose content archives are stored on public FTP servers. A remote package contains the usual files that make up Installer packages (including scripts and localized files) except for the package archive (the .tar.Z file), which is not contained in the remote package, but instead is stored remotely on a machine that services anonymous FTP access. Remote packages are small since they don't contain the package contents, and hence can be mailed or posted electronically without major resource consumption. A user that receives a remote package via electronic mail or bulletin board can install the package in the usual way by opening the package icon. When installation is performed, an anonymous FTP connection is created to the FTP server and the package contents are retrieved and installed on the user's machine.