

---

**Search Strategies for Resolving Alias Records**    Tips and Tricks

One of the key features of the **Alias Manager** is the search strategies built into the alias-resolution functions. The search strategies are designed to find the original target of an **AliasRecord**, even if the target has been moved, renamed, copied, or restored from backup.

The **Alias Manager** provides two basic alias-resolution algorithms: a fast search and an exhaustive search. This section describes the search algorithms. For descriptions of the functions that perform the searches, see **Resolving Alias Records** under **Using the Alias Manager** and the **ResolveAlias** , **MatchAlias** , and **GetAliasInfo** functions.

The first step in any nonrelative search is to identify the volume on which the target resides. The volume search considers the volume's name, creation date (which acts almost as a unique identifier for a volume), and type (for example, a hard disk, a 3.5-inch floppy disk, or an AppleShare volume).

The **Alias Manager** first looks for a volume that matches all three criteria: name, creation date, and type. The search succeeds if the volume is mounted and if its name and creation date have not changed since the record was created. If the search fails, the **Alias Manager** attempts to match by creation date and type only. This step locates volumes that have been renamed. Finally, the **Alias Manager** attempts to match by volume name and type only.

If the target is on an unmounted AppleShare volume, the **Alias Manager** attempts to mount the volume. It presents a name and password dialog box if appropriate. If the designated target is on an unmounted ejectable volume, the **Alias Manager** displays a dialog box prompting the user to insert the volume. Your application can suppress the automatic mounting, as explained in the description of the **MatchAlias** function.

<b>Fast Search</b>

The fast-search algorithm is designed to find the target of an **AliasRecord** quickly.

Depending on how you invoke it, the fast-search algorithm starts with either a relative search (described in **About Alias Records**) or a direct search (described in this section). Fast search can perform a relative search whether or not it has identified the target volume, but it must identify the volume before it can perform a direct search.

In a direct search, the fast-search algorithm first looks for the target by file ID (if the target is a file) or directory ID (if the target is a directory). (File IDs and directory IDs are described in the **File Manager** .) Even if a file has been renamed or moved on a volume, the **Alias Manager** can find it quickly through its file ID.

If the search by file ID or directory ID fails, fast search looks for the target by name in the original parent directory. This search locates the target if its file or directory ID has changed but it still exists by the same name in the parent directory (for example, if the target was restored from backup). Fast search compares file numbers on files found by name in the correct parent directory. If the file numbers do not match, the file is treated as a possible match—that is, it is put on the list of candidates and the search continues. If the

target is not found by name in the parent directory, fast search looks for a file by file number in the parent directory. A file with the same file number but a different name replaces a file with the same name but a different file number in the list of matches.

If the search by file ID or directory ID fails and if fast search cannot find the original parent directory, it searches for the target by full pathname. This search finds the target if it resides in the same location on the volume but the directory ID of its parent directory has changed (for example, if the entire parent directory was restored from backup).

If the search by full pathname fails, fast search attempts to find the file by tracing partial pathnames up through all parent directories, using parent directory IDs instead of directory names. For example, consider this full pathname:

Loma Prieta:MyReports:October:Sales Report

If the search by full pathname fails, fast search first looks for the partial pathname :Sales Report in the directory with the ID that the directory Loma Prieta:MyReports:October had when the AliasRecord was created. If that search fails, it looks for :October:Sales Report in the directory with the ID that Loma Prieta:MyReports had, and so on.

If you do not ask for a search by relative path first but do provide a starting point for a relative search, and if the AliasRecord contains relative path information, fast search performs a relative search after the direct search. The relative search succeeds if the relative path is the same as when the record was created and if the names of the target and its intervening parent directories have not changed.

### Exhaustive Search

The exhaustive-search algorithm scans an entire volume to look for possible matches

The **Alias Manager** typically performs an exhaustive search by calling the **File Manager** function **PBCatSearch**, searching for files or directories with a matching creation date, creator, and type. (See the **File Manager** for a description of **PBCatSearch**.)

**PBCatSearch** is available only on HFS volumes, not on MFS volumes. (See the **File Manager** for a description of the two file systems.) **PBCatSearch** is also available only on systems running version 7.0 and later. When **PBCatSearch** is not available, exhaustive search performs a search of the entire volume by making a series of indexed **File Manager** calls, searching for objects with matching creation date, type, creator, or file number.