NAME

automount – install automatic mount points

SYNOPSIS

/usr/sbin/automount [–**t** *duration*] [**-v**]

AVAILABILITY

SUNWcsu

DESCRIPTION

automount is a command that installs **autofs** mount points and associates an **automount** map with each mount point. The **autofs** filesystem monitors attempts to access directories within it and notifies the **automountd**(1M) daemon. The daemon uses the map to locate a filesystem, which it then mounts at the point of reference within the **autofs** filesystem. You can assign a map to an **autofs** mount using an entry in the **/etc/auto_master** map or a direct map.

If the file system is not accessed within an appropriate interval (five minutes by default), the **automountd** daemon unmounts the file system.

The file **/etc/auto_master** determines the locations of all **autofs** mount points. By default, this file contains four entries:

```
# Master map for automounter
#
+auto_master
/net          -hosts        -nosuid
/home         auto_home
/xfn          -xfn
```

The **+auto_master** entry is a reference to an external NIS or NIS+ master map. If one exists, then its entries are read as if they occurred in place of the **+auto_master** entry. The remaining entries in the master file specify a directory on which an **autofs** mount will be made followed by the automounter map to be associated with it. Optional mount options may be supplied as an optional third field in the each entry. These options are used for any entries in the map that do not specify mount options explicitly. The **automount** command is usually run without arguments. It compares the entries **/etc/auto_master** with the current list of **autofs** mounts in **/etc/mnttab** and adds, removes or updates **autofs** mounts to bring the **/etc/mnttab** up to date with the **/etc/auto_master**. At boot time it installs all **autofs** mounts from the master map. Subsequently, it may be run to install **autofs** mounts for new entries in the master map or an direct map, or to perform unmounts for entries that have been removed.

OPTIONS

–**t** *duration* — Specify a *duration*, in seconds, that a file system is to remain mounted when not in use. The default is 5 minutes.

-v — Verbose mode. Notify of **autofs** mounts, unmounts or other non-essential information.

USAGE

Map Entry Format

A simple map entry (mapping) takes the form:

key [–*mount-options*] *location* . . .

where **key** is the full pathname of the directory to mount when used in a direct map, or the simple name of a subdirectory in an indirect map. *mount-options* is a comma-separated list of **mount** options, and *location* specifies a file system from which the directory may be mounted. In the case of a simple NFS mount, *location* takes the form:

*host***:***pathname*

host is the name of the host from which to mount the file system (it may be omitted if the pathname refers to a local device on which the filesystem resides) and *pathname* is the pathname of the directory to mount.

Replicated Filesystems

Multiple *location* fields can be specified for replicated NFS filesystems, in which case **automount** chooses a server with preference given to a server on the local subnet or net.

If each *location* in the list shares the same *pathname* then a single *location* may be used with a comma-separated list of hostnames:

*hostname***,***hostname...***:***pathname*

Requests for a server may be weighted, with the weighting factor appended to the server name as an integer in parentheses. Servers without a weighting are assumed to have a value of zero (most likely to be selected). Progressively higher values decrease the chance of being selected. In the example,

man –ro alpha,bravo,charlie(1),delta(4):/usr/man

hosts **alpha** and **bravo** have the highest priority; host **delta**, the lowest.

Note: Server proximity takes priority in the selection process. In the example above, if the server **delta** is on the same network segment as the client, but the others are on different network segments, then **delta** will be selected — the weighting value is ignored. The weighting has effect only when selecting between servers with the same network proximity.

In cases where each server has a different export point, you can still apply the weighting. For example:

man –ro alpha:/usr/man bravo,charlie(1):/usr/share/man delta(3):/export/man

A mapping can be continued across input lines by escaping the NEWLINE with a ‘ \ ’ (backslash). Comments begin with a ’#’ (number sign) and end at the subsequent NEWLINE.

Map Key Substitution

The '&' (ampersand) character is expanded to the value of the **key** field for the entry in which it occurs. In this case:

jane **sparcserver:/home/&**

the **&** expands to **jane**.

Wildcard Key

The '*' (asterisk) character, when supplied as the **key** field, is recognized as the catch-all entry. Such an entry will match any key not previously matched. For instance, if the following entry appeared in the indirect map for **/config**:

* **&:/export/config/&**

this would allow automatic mounts in **/config** of any remote file system whose location could be specified as:

*hostname***:/export/config/***hostname*

Variable Substitution

Client specific variables can be used within an **automount** map. For instance, if **$HOST** appeared within a map, **automount** would expand it to its current value for the client's host name. Supported variables are:

ARCH	The output of uname -m.	The architecture name. For example "sun4"
CPU	The output of uname -p.	The processor type. For example "sparc"
HOST	The output of uname -n.	The host name. For example "biggles"
OSNAME	The output of uname -s.	The OS name. For example "SunOS"
OSREL	The output of uname -r.	The OS release name. For example "5.3"
OSVERS	The output of uname -v.	The OS version. For example "beta1.0"

If a reference needs to be protected from affixed characters, you can surround the variable name with '{}' (curly braces).

Multiple Mounts

A multiple mount entry takes the form:

key [–*mount-options*] [[*mountpoint*] [–*mount-options*] *location* . . .] . . .

The initial **/**[*mountpoint*] is optional for the first mount and mandatory for all subsequent mounts. The optional *mountpoint* is taken as a pathname relative to the directory named by **key**. If *mountpoint* is omitted in the first occurrence, a *mountpoint* of **/** (root) is implied.

Given an entry in the indirect map for **/src**:

```
beta –ro \
      /              svr1,svr2:/export/src/beta   \
      /1.0           svr1,svr2:/export/src/beta/1.0   \
      /1.0/man       svr1,svr2:/export/src/beta/1.0/man
```

automount would automatically mount **/src/beta**, **/src/beta/1.0**, and **/src/beta/1.0/man**, as needed, from either **svr1** or **svr2**, whichever host is nearest and responds first.

Other Filesystem Types

The automounter assumes NFS mounts as a default filesystem type. Other filesystem types can be described using the **fstype** mount option. Other mount options specific to this filesystem type can be combined with the **fstype** option. The location field must contain information specific to the filesystem type. If the location field begins with a slash, a colon character must be prepended, for instance, to mount a CD filesystem:

```
cdrom  -fstype=hsfs,ro     :/dev/sr0
```

or to perform an **autofs** mount:

```
src    -fstype=autofs      auto_src
```

Mounts using CacheFS are most useful when applied to an entire map as map defaults. The following entry in the master map describes cached home directory mounts. It assumes the default location of the cache directory, **/cache**.

```
/home  auto_home           -fstype=cachefs,backfstype=nfs
```

Indirect Maps

An indirect map allows you to specify mappings for the subdirectories you wish to mount under the **directory** indicated on the command line. In an indirect map, each **key** consists of a simple name that refers to one or more filesystems that are to be mounted as needed.

Direct Maps

Entries in a direct map are associated directly with **autofs** mount points. Each **key** is the full pathname of an **autofs** mount point. The direct map as a whole is not associated with any single directory.

Included Maps

The contents of another map can be included within a map with an entry of the form

+*mapname*

If *mapname* begins with a slash then it is assumed to be the pathname of a local file. Otherwise the location of the map is determined by the policy of the name service switch according to the entry for the automounter in **/etc/nsswitch.conf**, such as

automount: files nis

If the name service is **files** then the name is assumed to be that of a local file in **/etc**. If the key being searched for is not found in the included map, the search continues with the next entry.

Special Maps

There are three special maps available: –**hosts**, –**xfn**, and –**null**. The –**hosts** map is used with the **/net** directory and assumes that the map key is the hostname of an NFS server. The **automountd** daemon dynamically constructs a map entry from the server’s list of exported filesystems. For instance a reference to **/net/hermes/usr** would initiate an automatic mount of all exported file systems from **hermes** that are mountable by the client. References to a directory under **/net/hermes** will refer to the corresponding directory relative to **hermes** root.

The **-xfn** map is used to mount the initial context of the Federated Naming Service (FNS) namespace under the **/xfn** directory. For more information on FNS, see **fns**(5), **fns_initial_context**(5), **fns_policies**(5), and the Federated Naming Service Guide.

The –**null** map, when indicated on the command line, cancels a previous map for the directory indicated. This is most useful in the **/etc/auto_master** for cancelling entries that would otherwise be inherited from the **+auto_master** include entry. To be effective, the –**null** entries must be inserted before the included map entry.

Executable Maps

Local maps that have the execute bit set in their file permissions will be executed by the automounter and provided with a key to be looked up as an argument. The executable map is expected to return the content of an automounter map entry on its stdout or no output if the entry cannot be determined.

Configuration and the auto_master Map

When initiated without arguments, **automount** consults the master map for a list of **autofs** mount points and their maps. It mounts any **autofs** mounts that are not already mounted, and unmounts **autofs** mounts that have been removed from the master map or direct map.

The master map is assumed to be called **auto_master** and its location is determined by the name service switch policy. Normally the master map is located initially as a local file **/etc/auto_master**.

FILES

/etc/auto_master	master automount map.
/etc/auto_home	map to support automounted home directories.
/etc/nsswitch.conf	the name service switch configuration file.

SEE ALSO

automountd(1M), **mount**(1M), **fns**(5), **fns_initial_context**(5), **fns_policies**(5),

NFS Administration Guide

NOTES

The –**hosts** map must mount all of the exported NFS filesystems from a server. If frequent access to just a single filesystem is required, it is more efficient to access the filesystem with a map entry that is tailored to mount just the filesystem of interest.

Autofs mount points must not be hierarchically related. **automount** does not allow an **autofs** mount point to be created within another **autofs** mount.

Since each direct map entry results in a new **autofs** mount such maps should be kept short.

If a directory contains direct map mount points then an **ls –l** in the directory will force all the direct map mounts to occur.

Entries in both direct and indirect maps can be modified at any time. The new information is used when **automountd** next uses the map entry to do a mount.

New entries added to a master map or direct map will not be useful until the automount command is run to install them as new **autofs** mount points. New entries added to an indirect map may be used immediately.

An **autofs** directory associated with an indirect map shows only currently-mounted entries. This is a deliberate policy to avoid inadvertent mounting of every entry in a map via an **ls** –**l** of the directory.

The multiple location feature for NFS mounts allows the **automountd** daemon to choose the most appropriate server at mount time. While such a mount is in effect, the daemon does not monitor the status of the server. If the server crashes, **automountd** will not select an alternative server from the list.

Default mount options can be assigned to an entire map when specified as an optional third field in the master map. These options apply only to map entries that have no mount options.

The Network Information Service (NIS) was formerly known as Sun Yellow Pages (YP). The functionality of the two remains the same.