

## Using multiple Macs in your system :

You can monitor and control XTension from any other Mac that you have connected to your home network, and even from a dial-in.

So you might have an old Mac in the basement or closet, and control it from other Macs in the house.

With XTension running on your 'Host' Mac, copy the database to another Mac and start up XTension there in 'Monitor' mode.

The 'Monitor' copy of XTension tells the Host to transfer a 'snapshot' of the current database values to the Monitor.

The two machines then 'Target' each other. This means that any events which occur at the Host will be sent to the Monitor. These events appear on the Monitor as if they were occurring locally.

And, commands issued manually at the Monitor will be sent to the Host where they are executed as if they were issued at the Host.

Thus, you can start and stop the Monitor copy anytime and easily bring it up later on the same or different Mac.

Since you may have an older Mac as the Host in the closet, you won't want to burden it with a lot of graphic displays to update with each event that occurs.

However, on your Mac in the Den with that large color screen, you can display all the windows you want without burdening the Mac that is doing the real work.

You can create a system where you have two Macs, two X-10 interfaces, and one of the Macs is in 'Monitor' mode. As described above, the second Mac would not actually issue commands, but **would test** the 'Host' periodically. Should the current Host fail, then the Monitor Mac could take over the system. This is a very useful feature.

There are many configurations which you can create which may not all be logically reasonable.

## Verbs for coordinating multiple Macintoshes :

### Monitor Mode

**set monitor mode:** Set monitor mode on or off

set monitor mode true/false

If Monitor mode is False, XTension will perform normally.

If Monitor mode is True, XTension will not attempt to send commands directly to the X-10 interface.

If you set Monitor mode true, and 'target' this Mac at the 'Host' in the closet, then whenever you issue a command from the Monitor Mac, it will be sent to the Host to be performed.

This verb is specially useful when you have redundant XTension systems. If you happen to have two Macs, two controllers, and AppleTalk, you can run both at the same time.

Although it is possible to have both doing the same or different things, one should be normally in Monitor mode, and the other normally assumes that it's the Master whenever it starts up.

The Monitor machine normally starts up in Monitor mode, but it could initiate a scheduled event to run a 'System Integrity' script every minute to three minutes.

This script sends a simple message to the Master, and expects a response. If the response is missed for two times in a row, the script will automatically display a dialog, ring alarms, and then use the set monitor mode False command to assume Mastership of the automation system.

You must create this script if you want automatic backup switchover. See the Technote : TN.MultiMacs on the website for an example script.

Obviously this example is simple, and incomplete, but further elaboration would require knowledge of the installation.

## Targeting

**set target to:** choose a target Mac for AppleEvents

```
set target to string -- Chooser name of desired target Macintosh  
[ in zone string ] -- AppleTalk/EtherTalk zone name
```

```
example : set target to "Kitchen" in zone "First Floor"  
or      : set target to "Kitchen" in zone "*"
```

This verb allows you to dynamically select another copy of XTension

running on another Mac on your network to receive notification of all local XTension events.

This means that you can have a Mac in the basement connected to a X-10 interface, doing the real work, and another in the Kitchen or Office, which receives all XTension events and looks as if it had a controller attached.

If you also set the 'target' of the 'Monitor' Mac to the 'Host', then events which you initiate on the 'Monitor' Mac will be sent to the 'Host' who will then actually perform the command.

Thus the 'Monitor' Mac copy of XTension will appear to perform as if it were the 'Host'.

If you have only one zone, you must use : in zone "\*"

**remove target:** clear any target  
remove target -- No argument necessary

example : remove target

This verb allows you to dynamically clear the settings created by the 'set target to' verb, and effectively nullify the mode. This machine will no longer send AppleEvents to the previous 'target'.

**target name :** the address of the 'targeted copy of XTension'  
target name -- a variable which contains the 'current target' for this copy of XTension.

This should only be used as an interim device, as there will probably be future changes which obviate the need for this feature ?

## Synchronizing the databases between two/more Macs

**save snapshot:** save current values in database to disk file  
save snapshot file "file descriptor"

example : save snapshot file " "HD:Desktop Folder:XTensionf:Snap1"

**load snapshot:** load current values in file to database  
load snapshot file "file descriptor"

example : load snapshot file " "HD:Desktop Folder:XTensionf:Snap1"

These verbs (and menu items), will cause XTension to save a small file

to disk which contains **only** the current state and values for each of the units in your database. Using the load snapshot verb, you can transfer this to another copy of XTension ( and the same database ). The result is that you synchronize the values between databases on two Macs.