

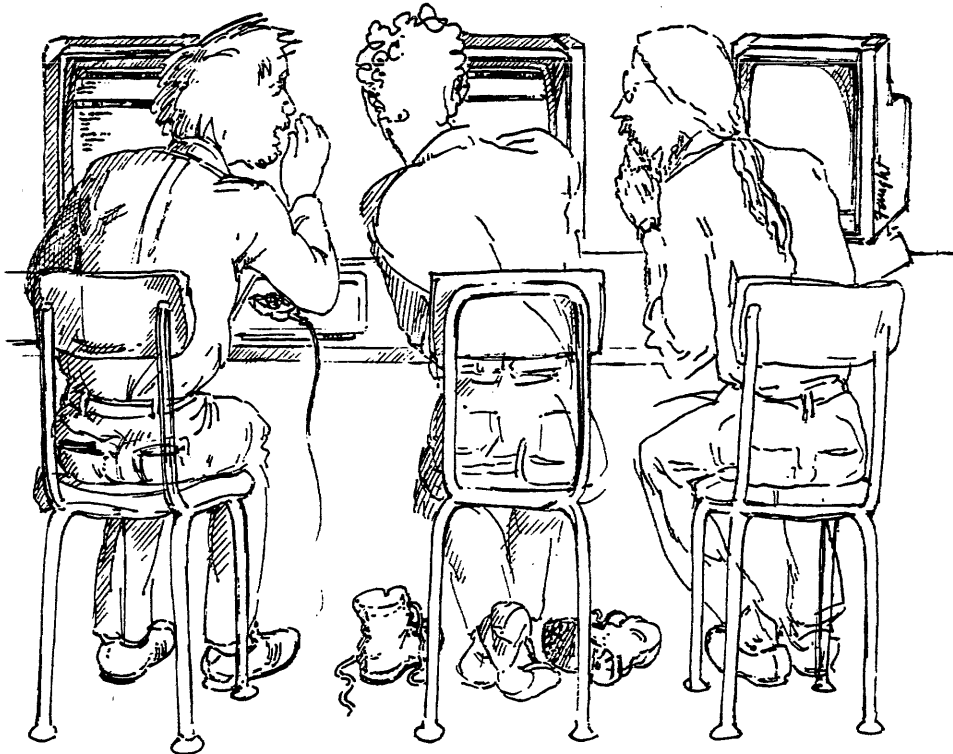
CARNEGIE-MELLON UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

SPICE PROJECT

Update: A File Transfer Facility

Dario Giuse



25 August 1984

Spice Document S165

Location of machine-readable file: `[cfs]/usr/spice/spicedoc/aug84/intro/update/update.mss`

Copyright © 1984 Carnegie-Mellon University

This is an internal working document of the Computer Science Department, Carnegie-Mellon University, Schenley Park, Pittsburgh, Pennsylvania 15213 USA . Some of the ideas expressed in this document may be only partially developed, or may be erroneous. Distribution of this document outside the immediate working community is discouraged; publication of this document is forbidden.

Supported by the Defense Advanced Research Projects Agency, Department of Defense, ARPA Order 3597, monitored by the Air Force Avionics Laboratory under contract F33615-81-K-1539. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Projects Agency or the U.S. Government.

Table of Contents

1 Introduction	1
2 Retrieval	1
2.1 Retrieving a Set of Files	1
2.2 Full Path Retrieval	3
2.3 Retrieval Switches	3
3 Storage	5
3.1 Creating a Storage Command File	5
3.2 Using Update to Store a Set of Files on the Vax	6
3.3 Switches	7
3.4 Commands for Storage Command Files	8
A Background Information	9

1 Introduction

Update is a program for transferring sets of files between the Perqs and a remote host. Update will run between a Perq and any Unix Vax. However, Spice *primarily* uses the CFS and Spice Vaxen. Update allows Spice developers to store on a Vax software that has been created on a Perq, and it allows users to transfer the newest version of Spice software from a Vax to their Perqs. Update's special feature is that it allows users to transfer only those files that are different from the ones already on their Perqs. This is accomplished by comparing the date of creation of each local file (on the Perq) in a set of files (SOF) with the creation date of remote files (those on a Vax). Because Update minimizes ethernet traffic and facilitates standardization of Spice software versions, it is used instead of Cmuftp for transfer of Spice software.

The version of Update described in this manual differs from earlier versions in that it allows SOFs to be stored and retrieved by logical names rather than full path names. This change enables the user to retrieve most Spice SOFs without knowing their location.

This manual is organized as follows: Section 2 describes how to use Update for retrieving Spice software. It includes a list of the applicable command switches, with explanations. Section 3 describes in detail the format of logical names for file storage, storage command files, and special command switches for file storage.

Before continuing with this document, check to be sure Update is present on your Perq. Look in the **boot:** partition for the file:

Update.run

If the program is not present, turn to the Appendix for instructions for retrieving Update.

2 Retrieval

2.1 Retrieving a Set of Files

A set of files is identified by a logical name. The logical name may contain alphanumeric characters and the underbar (`_`). The logical name usually includes, by convention, the name of the system (e.g., `accent`, `boot`, `cmuftp`, `dover`) plus some other information. The standard command format is:

```
update LogicalName { -Switch[= Value]}
```

The logical name usually consists of the system and, in some cases, other information required to enable you to retrieve the desired version of the system. There are, for example, four sets of Spice system files corresponding to the three types of Perqs and the two versions of the microcode. There are four logical names for retrieving these sets of files.

Update perq1a_OIO

Update Perq1a_CIO

Update Perq2
Update Perqt2

When you use the perq type as the logical name, as in the above examples, you will retrieve all the Spice .run files appropriate for your Perq type (all the .run files in [cfs]/usr/spice/). For most retrievals, you will only need to specify a logical name that corresponds to the name of the system you want to retrieve. If you want to retrieve DP, for example, type:

update dp

If you don't know the logical name of the SOF you need, a complete list is maintained in /usr/pupd/list on the Spice and CFS Vaxen. To see the complete list, type:

update -list

To see only those logical names containing a given string (including wildcards if you want), type

update -list = 'string'

For example,

Update -list = '*mint*'

You will generally find several logical names stored under a given program name for the standard Spice software. For example, the list command above might show you

```

mint           : /usr/spice/mint/run
mintrun        : /usr/spice/mint/run
mintrun_s      : /usr/spice/mint/run
mintrun_a     : /usr/spice/dev/mint/run
mint_pbh      : /usr/pbh/mint

```

The underscore suffix (**_s**, **_a**, or a username) tells you which version of the program is stored under that logical name: **_a** is the accent experimental version; **_s** is the spice experimental version (usable test version); and an individual's username denotes his own (usually non-standard, often developmental) version of the program. The word **run** in the logical name means it contains .run files, **src** means it contains source code, **bin** denotes binary code, etc. If two logical names refer to the same directory, they are two names for the same SOF; **mint** and **mintrun** are the same. You will usually only need to update run files.

If the SOF you are looking for does not have a logical name, you will need some other information in order to retrieve it. See Section 2.2.

If the SOF you want to retrieve is present on the CFS (or the host you have specified) and if it has been assigned the logical name you have typed, messages indicating that a retrieval is in process will appear on the Perq screen. These messages contain the path names of the files in the SOF and various combinations of periods (.) and pound signs (#). The pound signs indicate that files are being transferred, while the periods indicate pauses in file transfer due to heavy traffic on the Ethernet. When all files have been retrieved, a

message to that effect will be displayed. Update will then print out the Change Log to the SOF, a record of changes made by the maintainer to the files. If you don't want to see this, type 'n'.

Sometimes, instead of retrieval information, you will see a message describing a problem. A likely error is that either the logical name you have specified is incorrect or the SOF has not been assigned a logical name. Check the list of logical names. If you do not find anything resembling the logical name, you will need to use the second version of the Update command format, described in Section 2.2.

2.2 Full Path Retrieval

To retrieve an SOF that has not been assigned a logical name, you must know the full path name to the directory it was stored in. The alternate form of the command looks like this:

```
update 'FullPathName' {-Switch[= Value]}
```

For example, if you didn't know the logical name for the mintrun files, but knew it was in the /usr/spice/mint/run directory, you would type:

```
update '/usr/spice/mint/run'
```

2.3 Retrieval Switches

For most retrievals of current versions from the CFS Vax, switches are not needed, since retrieve is the default operation, CFS is the default host, and current is the default version. If you want to do something other than retrieve the current version of an SOF from the CFS Vax, you will need to include switches in the command line.

Retrieval Switches

Check	Give list of files that would be transferred but do not actually do any transfers.
List = 'spec'	List all logical names which match spec. It accepts wild cards.
Retrieve	Retrieve files (default). If maintainer switch is given the default changes to test version and it tries to assert maintainers rights (this insures that the maintainer has the latest version). Mutually exclusive with -Store .
Host = HostName	Specify the remote host. If storing, the default is "Spice". If retrieving the default is "CFS".
Test	Get test version.
Current	Get current version (default).
Old	Get old version.

Version = <i>Version</i>	Specify the version to retrieve. The default is "current".
Supersede	If the file being stored/retrieved is older than the one already there, overwrite anyway. If neither Supersede nor NoSupersede is used and an older file is found, the user is asked.
NoSupersede	Opposite of Supersede.
Document	Get on-line documentation, terminate immediately after.
NoAsk	Do not ask user anything after initial options.
Help	Get help, terminate immediately after.
AccentMode	Along with -PosMode controls the line termination character of text files being retrieved. Under Accent, the end-of-line character is an LF (line-feed) and under POS it is CRLF (carriage-return and line-feed). The default is to make the text files retrieved conform to the convention under the operating system (accent or POS) currently booted; AccentMode changes the end-of-line characters to LF's even if you are in POS, and PosMode changes them to CRLF's if you are in Accent.
PosMode	See -AccentMode
Searchlist	Under retrieval, the default is for Update to look for a version of the SOF only in the current directory of the Perq. Thus, it is possible that Update will try to compare dates between the Vax SOF and the Perq, not find it on the Perq, and ship the files to the current directory on the Perq—even if the SOF is already locally present in another directory. This switch makes Update look for the SOF in every directory on the current searchlist on the Perq, not just the current one, and it will not retrieve files to the current directory if they are already present in a directory on the searchlist.

If, for example, you have retrieved the current version of DP and you want to find out which test version files are different from the current version, you would type:

```
update dp -test -check
```

Here, the first switch changes the version from current to test, and the second switch changes the operation from retrieve to check.

In the previous example, the host was the default (i.e., CFS). Although most users will probably want to retrieve SOFs from the CFS Vax, there may be times when you want to retrieve from the Spice Vax; for example, for software that is not part of the standard Spice system. (You may retrieve experimental Spice

SOFs from the Spice Vax from one of two directory areas, /usr/spice/dev or /usr/accnet; the current release is usually in /usr/spice.) If the SOFs have been assigned logical names, they will be identical to those on the CFS Vax. You may also retrieve SOFs from your own or someone else's directory. If you want to retrieve the test version of DP from the Spice Vax, type:

```
update dp -host=x -test
```

3 Storage

This section describes how to use Update to store a set of files. It also lists command switches useful in storing and maintaining Spice SOFs for public use on the CFS and Spice Vaxen, as well as storage command file syntax.

Storage of a set of files is a two-stage process. The first stage involves creation of a storage command file. The second involves using Update to store the logical name and the SOF on the Vax.

3.1 Creating a Storage Command File

A storage command file is a list of all the files to be stored in the SOF. This file name consists of the logical name to be used for the SOF followed by the extension .upd. The file begins with its own name as its first line, followed by the names of all files in the SOF (one per line). Commands may also be included in the storage command file. All commands are prefixed by an @ at the beginning of a line. Commands and command formats appear in section 3.4.

Note that for simple cases where one wants to store all the files in a directory one can create the storage command file by issuing the command "dir * ~ *LogicalName.Upd*" and then editing the file to remove the directory header and trailer.

Here is a very simple storage command file:

```
@! MyProg.Upd:
MyProg.Upd
Module1.Pas
Module1.SEG
Module2.Pas
Module2.SEG
MyProg.Pas
MyProg.RUN
MyProg.SEG
```

Here is a more complex example:

```
@! OwnProg.Upd:
OwnProg.Upd
Module1.pas
Module1.SEG
Module2.pas
```



```
Module2.SEG
@Shell MakeDir SubDir>1
@Path SubDir>
NewProg.pas
MorProg.pas
@Mode Binary
ThisIs.Bin
```

A storage command file can contain calls to store (or retrieve) other SOFs, which act as separate calls to Update, as in:

```
@!      AllMyProgs.Upd
AllMyProgs.Upd
@@MyProg
@@OwnProg
```

When used with the `-store` switch, this Storage command file would store AllMyProgs.Upd, then store the SOFs designated by MyProg.Upd and OwnProg.Upd. With the `-retrieve` switch, of course, the additional SOFs would be retrieved.

3.2 Using Update to Store a Set of Files on the Vax

Once you have written the storage command file, you must define the logical name of your SOF in Update. To define the logical name TestProgram as /usr/bovik/run, you would type:

```
Update TestProgram -Enter = '/usr/bovik/run'
```

After you have typed this information, you will be asked to enter your login name and your password. After these items have been entered, the logical name you have selected will be assigned to your SOF in /usr/pupd/list.

To store the SOF, you would type

```
Update TestProgram -Store
```

Your SOF will then be stored on the Spice (X) vax. If you want to store files on a vax other than the Spice vax, use the `-host` switch.

When storing a set of files for the first time, Update will automatically create a versions file, which will be incremented every time the files are stored. You will only see a message about this the first time you store the SOF and Update informs you it is creating the versions file.

When you store your SOF, you will be prompted for a Change Log entry. If you are re-storing a set of files that are used by others, you should make a Change Log entry describing the changes you have made.

¹ This command will work a little differently in POS. Shell commands will not execute until Update is finished, and so Update will crash if it attempts to write to a directory created by a shell command in the Storage Command File.

3.3 Switches

Switches applicable to storage and maintenance are:

- Abdicate** Release maintainer rights. This leaves the logical name, and the SOF under it, in limbo—it can't be stored by anyone until someone retrieves it using the **-Login** or **-Maintainer** switch. Use of this feature is discouraged.
- Enter = 'Path'** Specify the remote path that the logical name is associated with. To be used for the case where the Set Of Files already exists. Can only be done once, after that causes a warning. Note that the quotes are necessary so that the path, which contains slashes, does not get confused with switches.
- Kill** Remove logical name from /usr/pupd/list. Does *not* destroy the SOF in storage, just means it can no longer be stored or retrieved under '*logical name*'.
- Store** Store files. The files are specified in a special file with a name made by appending .upd to the logical name. See the section on Storage Command Files for more information. Mutually exclusive with **Retrieve**
- Install** Push test version into current without doing any file transfers (a form of store, the store switch has no effect).
- Unlock** Remove any read locks from the given logical name. Use with extreme caution, if someone is doing a read at the time it could be messed up.
- Login = *UserName*** When transferring files log in under the user. Default is not to login for retrieving, and to ask the user when storing. This name is used as the maintainer name.
- Maintainer = *MaintainerName*** Same as **Login**.
- Server = '*ServerName*'** Specify a server to send the programs to on the other end. Useful mainly for maintainers of this program when installing a new server.
- Push** Push after storing (old test => current, etc).
- NoPush** Do not push (just overwrite test, default).
- Debug** Used for debugging, no transfers take place. Do not use unless you are maintainer.

Only the storage and retrieve commands are exclusive. If no action is specified then a retrieve is done. Otherwise exactly those switches specified are executed in the following order: Enter, List, Retrieve/Store (Install), Abdicate, Kill. Notice that some of these commands are mutually exclusive, but update does not check to see that the switches executed will not cause problems.

3.4 Commands for Storage Command Files

The file name for a storage command file is made by appending `.upd` to the logical name the files are to be stored under. The storage command file lists all the files to be stored and may have optional commands and comments. A line beginning with a `@!` denotes a comment. Commands in the file start with `@`:

<code>@!</code>	Comment
<code>@Shell</code>	Execute a shell command (while retrieving only)
<code>@StoreShell</code>	Execute a shell command (while storing only)
<code>@@StorageCommandFile</code>	Store (or retrieve) the SOF designated by <i>StorageCommandFile</i> (as if issuing a separate Update command)
<code>@@StorageCommandFile FullPathName</code>	Store (or retrieve) the SOF designated by <i>Storage Command File</i> and <i>Full Path Name</i> even if no logical name has been defined for the SOF. Do not use quotes (unlike the full-path retrieval command).
<code>@Path</code>	Set the local and remote paths (does not affect remote prefix). The format is: <i>@path localprefix localsuffix remotesuffix</i>
<code>@Mode</code>	Set the mode to one of { <code>text</code> <code>binary</code> <code>image</code> <code>auto</code> } Auto (the default) tries to guess the mode from the file extension. If the extension is not a recognized one, Update uses text mode. All files after such a command are transferred in the new mode, until the next mode command is issued.

Appendix A. Background Information

Before working with Update, you should know the Perq type and operating system you are using.

There are three types of Perqs and two operating systems currently in use at CMU. Sets of files for each Perq type/operating system combination are identified by logical names on the Vaxen; therefore, before running Update, you need to know which type of Perq and which operating system you are using.

Identifying Perq Type

The following descriptions will help you to identify the type of Perq you are using:

- Perq 1A:** Has a 24 mb disk and 16k CPU; is on the 10MHz Ethernet.
- Perq 2:** Has the logo *Perq 2* on front cover; is on the 10 MHz Ethernet.
- Perq t2:** Has the logo *Perq 2* on front cover; is on the 10 MHz Ethernet. Has the boot (Restart) button on the front of the main processor cabinet.

If you are unable to identify your Perq type from the above description, send mail to Spice@Spice specifying the Perq's physical location.

Identifying the Operating System

The following information should help you to determine which of the two operating systems you are using:

- Accent:** The standard Spice operating system, which is booted automatically if no boot keys are pressed.
- POS:** The old Perq operating system, which is used as a back-up at CMU; POS is booted when the "b" key is pressed.

Identifying the microcode version

If you are using a PERQ1a it is important to know what version of the microcode your PERQ is running.

If you are currently using the S4 system, boot POS and type,

```
Details /sys
```

If the switch /sys does not exist, you are running OIO. Otherwise, you are running CIO.

If you are currently using the S5 system, type

```
Details -io
```

This will tell you what version you are using.

Once you have identified the microcode version, use

```
Update Perq1a_OIO
```

or

Update Perq1a_CIO

to retrieve the current version of the system.

Retrieving Update

If your Perq has an old version of Update, it can be used to retrieve the current version. Look in the boot area for the file:

update.run

If the program is present, use the procedures described earlier to retrieve it. If you do not have an **update.run** file, you will have to use CMUFTP to obtain it. To use CMUFTP to retrieve a current version of Update requires specifying path names which are likely to change as new versions of Update are released by the maintainers. If you have difficulty locating the right directory, send mail to Spice@Spice or ask an expert.