

Technote 1191

USB Software Update

CONTENTS

[Software Update Process Overview](#)

[USB Software Update Description](#)

[Registering With DTS](#)

[Using The USBEditor Web Page](#)

[Installing USB Shims and other software](#)

[Apple Installer Script Tips](#)

[Testing Your Live Update](#)

[Tips](#)

[Summary](#)

This Technote describes the live software update process under Mac OS 9.0 and the support that it provides for USB devices. It explains how developers can register with DTS to have access to the USB Update Database. The developer updates the database to provide information that the Software Update process uses to download the appropriate software for the USB Device.

Software Update Process Overview

Software Update, a new feature provided in Mac OS 9, makes it possible for system software to be updated over the Internet, as newer software is made available. At present this feature is limited to Apple system software. A derivative of this functionality has been provided for USB devices: finding a driver for an unrecognized USB Device. This means that if a USB device is attached for which there is no recognized driver, Software Update will ask the user to search for the necessary driver on the Internet. If a suitable driver is registered, the software package is downloaded.

For USB developers, there are six important aspects towards implementing support of Software Update for their products:

1. Registering with DTS to enable access to the USB Software Update Web page,
2. Entering the device data into the USB Editor staging server database along with the URL to your web server where the software will be downloaded,
3. Packaging your software so that it can be installed properly,
4. Using the Software Locator Switch to set Software Update to search the USB Editor staging server database,
5. Testing the USB Software Locator process by connecting a USB device when no USB driver is present to support the device, and
6. Setting the USB Editor database record to move the record to the production server.

Note also that the current implementation only supports finding driver modules over the Internet when no matching driver can be found on the host system. There is no support at this time for updating existing USB driver software, or for finding "better" USB driver software. In addition, there is no support by Software Update for other types of third- party software except for USB. Another limitation, is that the Software Update process is only available for Mac OS 9.0 and later. There is no support for Software Update in earlier releases of the Mac OS.

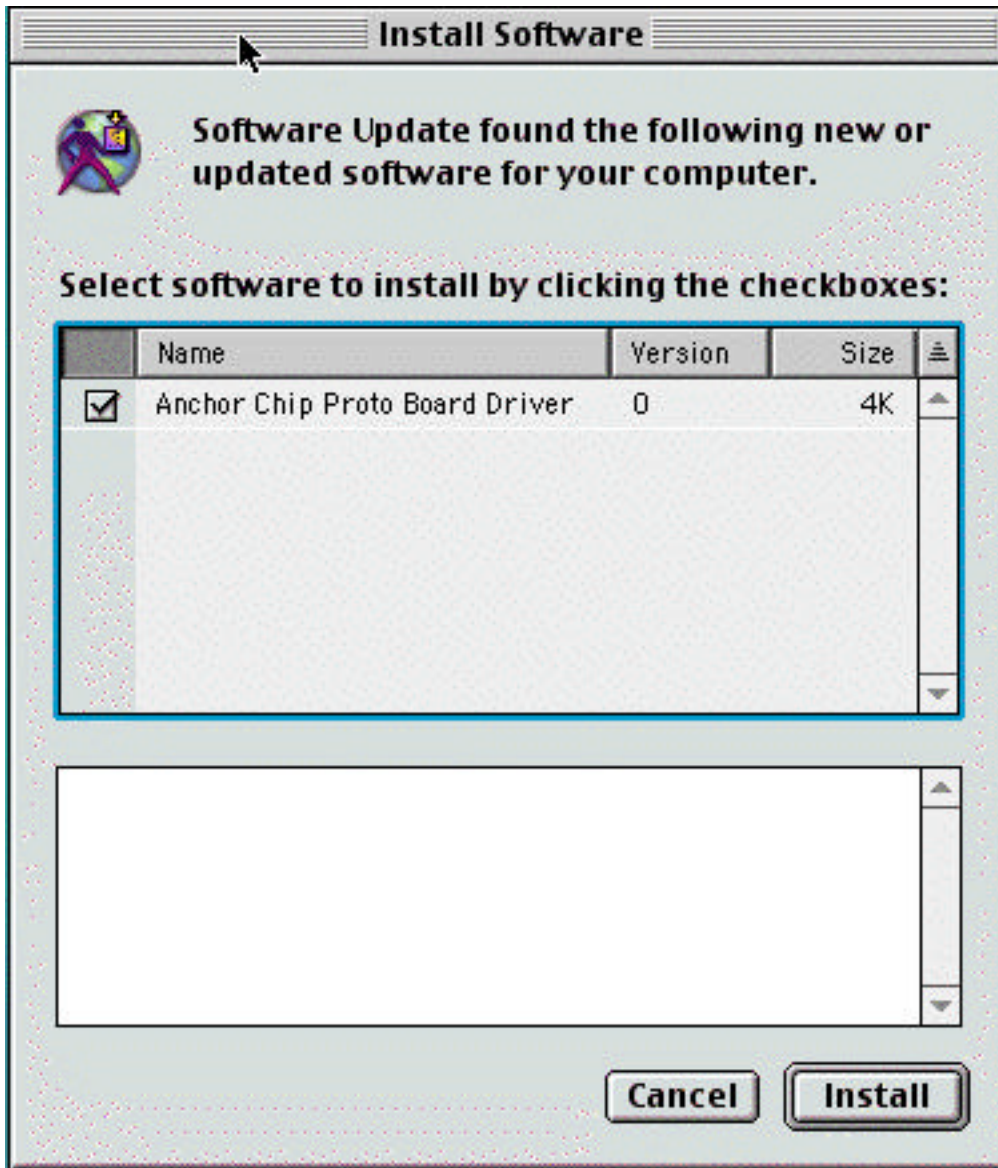
[Back to top](#)

USB Software Update Process Description

When the user attaches a USB device to the Macintosh, Mac OS USB searches the ROM and the Extension folder for an 'ndrv' driver file whose `DriverDescription` record matches the characteristics of the device. If no matching driver is found, the following alert is displayed under Mac OS 9.0 or greater:



Note the difference in wording between this alert and similar alerts provided under previous releases of Mac OS. Under Mac OS 9, the user is given the option to have the system search the Internet for a driver to support the device. If the user clicks the OK button, the system will access the Internet to find the Apple USB database web page. If the web page can be accessed, then Mac OS USB searches for an entry in the USB database matching the Vendor ID, Product ID, device class, and subclass. If a matching entry is found, the user is presented with the following alert:



In the above example, only one software package was found. It could be that there are several software packages available for a device. In such case, all matching selections are presented to the user. The user can click on any software package listing and read the associated description in the lower window pane in

order to choose the appropriate software.

The user has the option to download the selected software. If the user chooses to install the software, Software Update uses the URL specified in the USB database to download it. If the downloaded file is an Apple Installer Script, then the script is executed. If the downloaded file is any other file, then the file is placed into the developer-specified location, typically the Extension folder. Once the file is downloaded, Software Update tells USB to rescan for a USB driver to support the device. Note that while a USB driver module may get matched, there may be other processes that need to be activated, or installed, which may require the computer to be restarted.

The Apple USB Database is password-accessible from the [Apple USB Editor Web page](#). Developers must register with DTS to be able to access database records based on their USB Vendor ID. It is important that your device implement your vendor ID.

[Back to top](#)

Registering With DTS

The Software Update process begins by contacting [Worldwide Developer Technical Support](#) and submitting:

1. AppleConnect/Directory Services ID
2. Your name and company
3. The USB Vendor ID's for which you request access

After submitting this information, you can access the [Apple USB Editor Web page](#). The USB Editor Web page is a staging server where you enter the Software Update information needed to find the USB software for your device, and verify that the update process works as advertised. After the registration is approved, you can access the USB Editor Web Page to add, modify, or delete database entries associated with your vendor ID. For security reasons, we require that your Apple Connect ID be associated with the Vendor ID that you are requesting access for. If the ID does not match, we will require that you have the official vendor ID owner access the USB Editor database and provide the information on where to access your software.

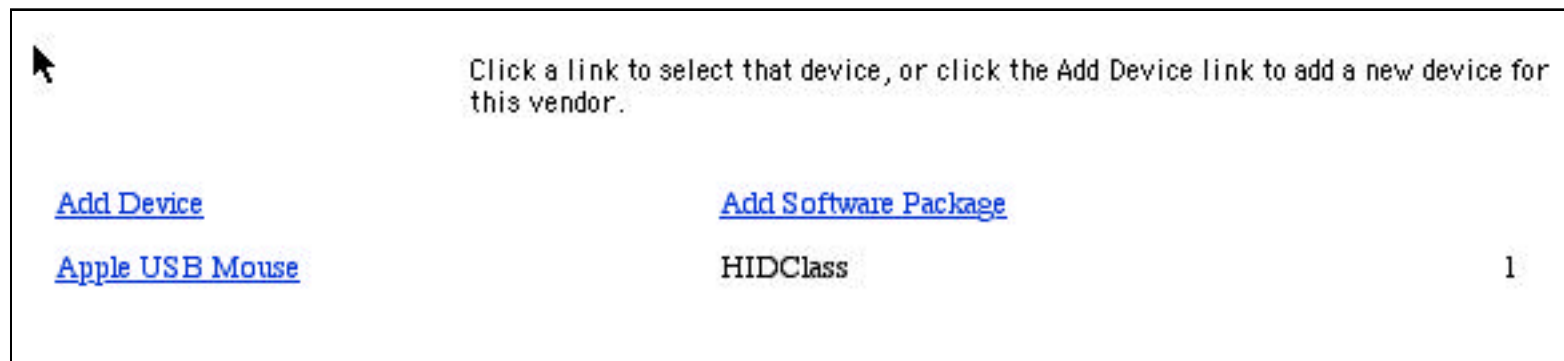
DTS understands that many developers are writing USB software for devices which have vendor ID's associated with a different developer. The current design of the USB Editor web page was based on the assumption that there would be only one developer accessing all records associated with each vendor ID. If multiple accounts accessed a vendor ID, they would have access to all records associated with the vendor ID. For this reason, we are limiting access for each vendor ID records to the developer assigned that ID. At some point in the future, we hope to have a better solution.

[Back to top](#)

Using the USB Editor Web Page

The information that you enter into the USB Editor Web page is sent to a staging server. Any information which you enter will only be available to you and others who have the Software Locator Switch application to have Software Update use the staging server instead of the production server. This allows you to test the Software Update process prior to making it live.

At the [Apple USB Editor Web page](#), enter your account name and password. You are presented with the database page for your vendor ID. The present Apple entry is shown below:



Use the "Add Device" link to define a new Device. Use the "Add Software Package" to define the software for the device. You can provide localized software that will be presented to the user based on the country code value associated with the active system software. Use the "Add Description" link to set a user-visible description for the software package, which Software Update displays to the user.

Add Device

The Add Device link takes you to a web page as shown below. Use this page specify the characteristics of your device, which the USB Software Locator will use to match a supported device:

Vendor:

Apple Computer

Last Changed on:

None

Name:

This should be the name of the USB device e.g. 'Apple USB External 1.2GB Drive'.

USB Class:**USB Subclass:**

This should be the USB device subclass as a decimal integer e.g. '128'.

Product ID:

This should be the vendor-specified USB product ID number e.g. '123'.

Protocol*:

This should be the USB protocol number used by the device, as a decimal integer e.g. '65535'.

Version*:

This should be the version number e.g. '1.2'.

The vendor field will be filled in by the system and should be the name of your company. You are required to enter the Device Name, USB Device Subclass, and Product ID. You must also set the USB Class pop-up menu to match the device class of your device. The Protocol and Version fields are annotated with an asterisk. These fields are optional. Click the "Save" button to have this information saved in the Apple database. Click the "Cancel" button if you don't want the information saved.

The Device Name is used in the "Add Software Package" page to associate the software package with a device. The Device Name is identical to the product name displayed to the user in the Software Update process.

If your device class is not displayed in the pop-up menu, please submit a note to [Developer Technical Support](#), and indicate the device class name and value.

[Back to top](#)

Add Software Package

Once you have defined the device and saved this information, return to the device web page and click the "Add Software Package" URL to take you to the following page:

Device:	
Apple USB Mouse	
Last Changed on:	
None	
Last Moved to Production on:	
None	
Move to Production:	
Y	
Enclosed file name:	
<input type="text"/>	This should be the name of the enclosed software package after unpacking e.g. 'AppleUSBDiskDriver'.
File Size (in kilobytes):	
<input type="text"/>	This should be the size in kilobytes of the downloadable software package when installed, as a decimal integer e.g. '512'.
Destination Folder:	
<input type="text"/>	

SystemROM1	
Software is required to use device:	
<input type="button" value="Y"/>	
Restart machine after install:	
<input type="button" value="Y"/>	
Language:	
Z : International English	
Download URL:	
<input type="text"/>	This should be the fully-qualified URL to obtain the downloadable package e.g. 'http://myhost/mydir/myfile.hqx'.
Product name:	
<input type="text"/>	This should be the human-readable description of the product e.g. 'Apple Mass Storage USB Driver'.
Product requires USB version:	
<input type="text"/>	This should be the version number of Apple's USB software [in Apple-standard major.minor(buildType)(buildNumber) format] required by the downloadable software package e.g. '1.1g5'.
Short Description:	
<input type="text"/>	This should be the short customer-visible name of the downloadable software package e.g. 'Apple USB Mass Storage Driver'.
Binhexed file name*:	
<input type="text"/>	This should be the binhexed package name available from the location URL e.g. 'myfile.hqx'.
Version*:	
<input type="text"/>	This should be the version number e.g. '1.2'.
<input type="button" value="Cancel"/>	<input type="button" value="Save"/>

The fields and pop-ups on this web page which have no explanation, are used as follows:

Device pop-up - select the device for which this software package is used. You can select only among those devices which have been previously defined.

Last Changed on Field- indicates the most recent modification date for this record.

Last Moved to Production on Field - indicates when this record was last moved to the production server.

Move To Production pop-up - indicates whether the updates/changes to the record will be moved to the production Software Update server. Set to "Y" to have the record moved to the Production Server. The changes are checked once a day. Allow 24 hours before your changes become live. While you are editing and testing the Software Update support for your device, set this pop-up to "N". When you are satisfied that the correct software package has been identified for the device, and will be installed properly by Software Update, set the pop-up to "Y".

Enclosed File Name Field - as indicated, this is the name of the unarchived, de-binxed file. If the file is an Installer Script with the file type and creator 'kajr', then Software Update will execute the Installer script. For the script to work properly, the associated tome file must be merged into the script. Refer to the section [Installing USB Shims and other related software](#) below, to understand how to activate a USB shim without forcing a reboot.

Destination Folder pop-up - if your software package consists only of a USB module, then set the Destination Folder pop-up to the Extensions Folder where the module will be installed. If the software package is an Installer script, then set the destination folder to the `ItemsDeletedAtBootFolder`. After the installation is complete, the file will be deleted after the system is rebooted.

Language pop-up - set this pop-up to the language that the software package supports. You can provide localized versions of your software. Each software package will be matched by this pop-up to the country code for the target system software. If there are no software packages with a match for the country code, and if there is an International English package, then this choice is presented to the user.

Download URL Field - provide the complete URL that Software Update will use to download your software package. The software package will be maintained on your support web server. You will be responsible for ensuring that the URL remains valid.

The "Binhexed File Name" and the "Version" fields are annotated with asterisks. These are optional fields.

There can be multiple software packages associated with a device. Each software package is presented to the user in the Software Update window. As the user clicks on each selection, the associated product description is presented to the user.

[Back to top](#)

Add Description

You can provide a description of your software package using the "Add Description" link. There will be an Add Description link for each software package that you provide. At a minimum, you should provide a description using International English:

Driver:
Apple USB Mouse Driver

System Script:
smRoman : 0

Long Description:

This should be the customer-visible description of the downloadable software package e.g. features, caveats, etc. HTML is encouraged.

Cancel Save

[Back to top](#)

Installing USB Shims and other related software

For many software packages, there are multiple files to be installed. There are two options in this case. Both options require the development of an Apple Installer script. One option is to implement the script so that the files are installed to their appropriate location then the system is rebooted. For software packages that implement a USB shim, there is a new call defined in version 1.3.5, `USBAddShimFromDisk`, which will activate a USB shim on the fly. Unfortunately, this call is broken under Mac OS USB 1.3.5, and will be fixed under Mac OS USB 1.4 and greater. The following is a description of how the `USBAddShimFromDisk` call can be implemented.

A USB shim can be installed as part of the USB Software Update process. Under Mac OS USB 1.4, it is possible for the USB shim to be activated on the fly so that the CPU may not need to be rebooted. To make this possible, there are several points to consider.

- The `USBAddShimFromDisk` call is used to activate the USB shim by a post-installation action atom in an Apple Installer script file.
- Only the Apple Installer script is supported by the Software Update mechanism.
- In the case that the Apple Installer script is downloaded, the associated "tome" file must be merged into the script file. A tool is available in the sample code package that merges the tome file into the Installer script file.
- The Apple Installer is not part of the download process. Software Update uses the Installer Engine, which is included in the Application Support folder, to perform the installation actions of the script.

For those unfamiliar with either the Apple Installer or a tome file, the tome file contains the compressed files and resources which will be installed by the Installer Engine. The Installer script contain a number of resources, called "atoms" that describe:

- the rules to check whether the installation can be performed,
- what files and resources will be installed or removed,
- where the files and resources can be found, and
- additional code resources to be executed in support of the installation process.

The source for the Installer script is a resource source file. Typically, the MPW Rez tool is used to build the Installer Script; however, it's also possible to use the Metrowerks CodeWarrior IDE for this purpose. An example USB module and USB shim with Installer script source and post-installation action atom

source is provided in the Mac OS USB DDK.

The above describes the use of an Installer `action atom` to instantiate the USB shim. Alternatively, the action atom code could be written to perform other tasks. Refer to the [Apple Installer SDK](#) for more information on the implementation of action atoms. The installation action is performed by Software Update. At this time, there is no support for other installers by Software Update.

Apple Installer Script Tips

The following are some notes to help you in creating an Apple Installer script to work with Software Update:

- The source code for an Apple Installer script is a resource script file, which can be compiled into a resource file using either MPW or Metrowerks CodeWarrior.
- You will need to use the Macintosh Programmer's Workshop (MPW) build environment for two steps of the Installer Script build process - 1. to use the ScriptCheck tool, and 2. to use the MergeScriptMPWTool to produce the single Installer Script file that Software Update knows how to implement.
- Use the FileCrusher Tool to create a tome file, the file containing the compressed files and resources, which the Installer script will install.
- Use the MergeScriptMPWTool to merge the tome file into the Installer Script file.

[Back to top](#)

Testing Your Live Update

Apple has set up two servers to handle the USB software update process, a production server and a staging server. When you define or update a software package entry, you should ensure that the "Move To Production" pop-up menu is set to "N" until you are satisfied that the update process works as expected. In order to have Software Update access the staging server, use the Software Locator Switch application to direct Software Update to either server for USB software.

The Software Locator Switch application is included as part of the Mac OS USB DDK 1.4 and later. Once you are satisfied that the software is properly downloaded from the staging server, set the "Move To Production" pop-up menu to "Y". Within 24 hours, the changes will be picked up by the production server.

[Back to top](#)

Tips

You may find the following information useful when planning your software update:

- We encourage you to make the download available from your web server, as opposed to using File Transfer Protocol (FTP) server. You can specify `<http://www.mycompany.com/myusbdriver.hqx>` for the download address as an alternative to specifying the download from an FTP server. Setting the download from an HTTP server gets around the problem of trying to access an FTP server from within a firewall.
- If you must use an FTP server, consider supporting PASV mode. Many customers will have machines that reside within a firewall where only passive connections to FTP servers are allowed.
- HTML files that are going to be displayed in Software Update should have the following two items.
 - A META tag such as `<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=x-euc-jp">`, where the charset specifies the encoding where the HTML was created.
 - The body of the HTML should be enclosed by a `` where font name specifies the font to be used to display the HTML.

Software Update will convert the HTML from UTF-8 to the encoding specified in the META tag. If a charset is not specified, Software Update will use the default encoding for the running System. The list of Internet names for MacOS encodings is located at:

<http://developer.apple.com/techpubs/mac/TextEncodingCMgr/TECRefBook-151.html>.

- For example, the Internet name for Mac Roman is "x-mac-roman".

[Back to top](#)

Summary

Mac OS 9.0 and Software Update provides a new mechanism for USB developers to support their devices. Users can easily download a USB driver to support their devices if they have an Internet connection and the USB driver software is registered with the USB Editor Web Page.

Further References

- [Apple Installer SDK](#)
- [Latest Mac OS USB DDK](#)

[Back to top](#)

Downloadables



[Acrobat version of this Note \(K\)](#)



[Mac OS USB API Reference Guide.pdf](#)

To contact us, please use the [Contact Us](#) page.
Updated: 22-November-1999

[Technotes](#) | [Contents](#)
[Previous Technote](#) | [Next Technote](#)