DVC

Roger Hågensen

COLLABORATORS

	TITLE : DVC		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Roger Hågensen	January 2, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

DVC

1.1 DVC Guide - Contents

```
                  System update util!    DVC 2.5 ©~Msi Software 1997    System  ↩
                     update util!
         Compiled using AmigaE v3.2e from Wouter van Oortmerssen.
----------------------- DVC = Direct Version Checking -----------------------
```

"Roger Hågensen <emsai@online.no>" "http://www.sn.no/~msi/index.htm"

1.2 DVC Guide - CLI

```
CLI Usage: DVC FILE/A INFO/S INFOGUI/S LIST/S MOVE/S PUB/K RX/K
```

File is the name of a file.

Info is a switch, by using it DVC will just print the name and version, of the given file to the CLI window, no installing/update/replace is done at all. This makes it easy to create script that make a name/version list of files.

Infogui is the same as Info, only if used instead of Info, it will display the result in a GUI. Info and Infogui can not be used at the same time, if both are used only Info will work.

List is a switch, by using it you tell DVC that the file is a DVC list. A DVC list is just that, a list of files with version numbers. Look at the example LibList.dvc and MakeLibList files included with this archive, MakeLibList can be changed so it creates lists of devices, handlers, datatypes, etc, etc. It is advised you use DVC’s INFO switch instead of c:version etc, since the version checking used for INFO and LIST is the same one. For more info go here !

Move is a switch, by using it you tell DVC that the selected file (original/source) will be removed after a sucessfull install/update/replace.

Pub (optional) is the name of a pubscreen DVC will show the GUI on! With the PUB option you can use DVC with DirWork/DirectoryOpus etc, in fact, that’s what I do, and I strongly suggest you do that also.

RX is used for browser/internet support (Plugins/DVC.rexx), and is only used with Type 2 lists. If you do not intend to use DVC with Internet, or have no knowledge of rexx etc then leave this one alone. For information on how to use DVC with Internet, look at the Internet part of this Guide. For more info on the RX option look at Plugins/DVC.rexx .

After you have set DVC up in your filemanager, you can use it to check the version of new system files like libraries/devices etc. And it will let you install/update/replace if you want to ;-)

No options will pop up the WB GUI, even when Workbench is not running it still works. But only the Check and Quit buttons can be used.

When started from CLI you can force DVC to stop and quit
when it is doing any looping/repeative work.
Like checking more than one file,
or searching for version info or when copying a file.
DVC can be stopped by pressing Ctrl-C,
or if DVC was started with run, the BREAK command can be used.

Any HALF files will be removed (all copied files are checked).

DVC can also be made resident.
When DVC is working on a list,
you may press a Quit button, to stop DVC working on the list.

DVC also have "sub.extension.support"
So files with names like "targa.datatype.68020",
is checked/installed as "targa.datatype" etc.
Anything after the second "." (including the "."),
is stripped away since it’s assumed to be a cpu extension etc.
Files with targa.datatype_020 or similar will not be stripped
since "_" is a valid character for filenames,
but "." is only valid in a filename if used ONCE.

1.3 Plugins/DVC.rexx

```
/* $VER: DVC.rexx 1.0 (27.2.97) Msi Software
 *
 * This script get the browser file & screen,
 * and get the correct browser address for use with sys:rexxc/rx
 * Then DVC is started with the RX option,
 * a string that holds the rexxaddress and browser command.
 * Currently this script support AWeb/Voyager/IBrowse,
 * but can easily be adapted for other browsers that use REXX.
 * Perhaps even for use with non-WWW programs like FTP etc.
 *
 * (The URL’s in DVC’s Type 2 lists may be HTTP, FTP, etc, etc.)
 */

/* Allow use of result strings */
options results

/* Get a list of the current ports */
ports=show(’P’)||’ ’

/* Get the arguments */
parse arg cmds
parse var cmds file screen

IF pos(’AWEB’,ports)>0 THEN
   DO
      /* Get the active AWeb port */
      parse var ports dummy ’AWEB.’ portnr .
      address value ’AWEB.’||portnr
      ’GET ACTIVEPORT’
      awebhost=result
      /* Get screenname if no screenname was given */
      if screen='' then do
         'GET SCREEN'
         screen=RESULT
      end
      /* Set the browser port and command for AWeb */
      browsercmd=awebhost' Open'
   END
ELSE
   IF pos('MINDWALKER',ports)>0 THEN
      /* Set the string for Voyager */
      browsercmd='MINDWALKER OpenURL'
   ELSE
      IF pos('VOYAGER',ports)>0 THEN
         /* Set the string for Voyager */
         browsercmd='VOYAGER OpenURL'
      ELSE
         IF pos('IBROWSE',ports)>0 THEN
            /* Set the string for IBrowse */
            browsercmd='IBROWSE GotoURL'
         ELSE
            /* Browser not running, or not supported */
            exit

/* As you can see, only 'BrowserPort BrowserCommand' is needed, */
/* DVC will later append the download url to the end of the browsercmd */
/* and send that string to sys:rexxc/rx */

/* Strip spaces and quotation marks in front and behind of strings, */
/* this is needed since the script (and DVC) later add "" marks. */
file=strip(file,'B',' ')
screen=strip(screen,'B',' ')
file=strip(file,'B','"')
screen=strip(screen,'B','"')

/* Start DVC*/
address command 'DVC 'file' LIST PUB="'screen'" RX="'browsercmd'"'

exit
```

1.4 DVC Guide - Workbench

```
                WB Usage: (asl.library needed to use file requester)

When used from WB it can use the ASL Filerequester,
AppIcon (Iconify) and AppMenu (Hide)
all three can make use of multiselection,
and can also do this while starting from WB (shift click icons).
For more information on this, read you WB manual.

Multiselection support selection of files, in different volumes/drawers.
But please note, that the filerequester does not.

There is 4 buttons:
Check, Hide, Iconify, About,  Quit.
```

Check will compare the selected file(s) with those installed,
and install/update/replace (it will ask you first).

Hide will hide DVC in the WB Tools menu,
and choosing any file(s) and then selecting DVC in the menu
the file(s) will be checked.
Just select DVC in the menu to get the GUI.

Iconify will put a icon on the WB called DVC,
you may choose file(s) by shift clicking on them and then
double click on the DVC icon,
or just drag and drop the file(s) on the DVC icon.
Just doubleclick the DVC icon to get the GUI.

About shows you some program info.

Quit will quit DVC.

When started from WB, DVC also support these tooltypes:

Hide, Iconify, Pub, List and Move.

Hide and Iconify will start DVC Hidden or Iconified.
Pub= followed by a Pubscreen name,
will make DVC open on that screen.

Move is a switch, by using it you tell DVC
that the selected file (original/source) will be removed
after a sucessfull install/update/replace.

List is a switch, by using it you tell DVC that the file is a DVC list.
A DVC list is just that, a list of files with version numbers.
Look at the example LibList.dvc and MakeLibList files included
with this archive, MakeLibList can be changed so it creates
lists of devices, handlers, datatypes, etc, etc.
It is advised you use DVC’s INFO switch instead of c:version etc,
since the version checking used for INFO and LIST is the same one.
For more info go
here
!

There is also a secret tooltype, it is also called List:
List= followed by a filename will tell DVC that the file is a DVC list.
You should only use this alternative List tooltype when it is needed,
since the normal List switch is the easiest one.

Please note that only one DVC LIST can be loaded,
so the list must either be the only icon clicked,
or the first icon (except the DVC icon).
If DVC goes into LIST mode then any other files are forgotten,
for now this is a take it or leave it thing.
It may change in the future, but as of now the loading of several lists
are not supported, and in Check/Hide/Iconify, lists are not supported.

When DVC is working on a list, or when multiselection is used you may press a Quit button, to stop DVC working on the list/files.

1.5 DVC Guide - Introduction

First of all, to install DVC copy the file DVC to where you want it,
if you want the icon, copy that along with DVC, that’s it :-)

DVC stand for "Database Version Checking",
and was the title of the old original (never released).
Now DVC is just called DVC ;-) Well, not exactly...
I have now decided that DVC will stand for,
"Direct Version Checking" :-)

The old DVC used several databases to compare versions
of installed files with new files.
But it used an arsenal of complex DOS scripts to do this,
the DVC command itself just compared version numbers.
So the user had to update the databases himself,
via scripts, and even I got lost several times :-)

But the new DVC do all this by it self.
The old DVC was never released so I simply trashed it ;-)
The new DVC uses the directories themself as databases,
after all, it is where the files are supposed to go,
so why not see if they are there and what version.
So thanks to the new DVC, no databases is needed.
And no need to tell it where to put files,
all you do is click on either the YES or NO button,
it can’t get any easier than that ;-)

DVC using databases/filelists was not included in DVC 1.x
But when DVC 2.0 was created,
a new LIST switch (CLI) and List tooltype (WB) was added.
These two switches let DVC use a list of files to check
if the user (you) have the latest version of a file.
Imagine a list of library (LIBS:) files,
DVC will get the name and version from the list.
Then it will check if you have this file,
if you have, then it will compare versions.
If you have an old version, DVC will tell you so.

This makes it very easy to check your files,
especially if someone like the author of LibGuide update
such a LibList.
But these DVC Lists may be used for other files also,
like datatypes/devices etc, etc, even other WB files,
like SetPatch, or WBPattern, etc, etc, etc.

There is more:
DVC also support direct download of new version of files from Internet,
and I don’t mean new versions of DVC.
But new versions of the files in a list,
YES! Now people can go to a site with a list and check if they have
the latest version, and if they don’t they will be able to download

the new version, for more info on the lists read
this
.

Hopefully more and more DVC list maintainers will appear,
and I hope people like the authors of ClassAct and MUI,
and DataType authors etc, will make use of DVC Type2 lists.
Thus making it easy and quick for users to check their installed
files, and download new ones if needed.

It is not just Type 2 lists that can be used with Internet,
Type 0 and Type 1 can also be used, depending on the situtation,
and what list is most usefull.
But in my oppinion, Type 2 lists will be more popular.
And you can have more than one list on a WWW page,
using all three types, or several lists of the same type.

A list can have anything from 1 filename to several thousand filenames.
And will work just as well with 1 filename as with 1000 filenames ;-)
But it is better if large lists are split up,
since large lists use more memory and take longer to to load.
And more than one list makes it easier to update the list,
and let you have lists for different types of files.

Type 2 lists have limited usage when not used with Internet,
it depends on the browser used, and how it is used.
Type 0 and Type 1 was created for normal usage,
while Type 2 was created just for Internet.
But all three can be used with or without Internet ;-)

DVC also have a email notify requester that is available,
when a new version of a file is found.
It will let you save a small file to Ram:
that you later can email to a list maintainer.
Or if you are connected to the net, you can mail it directly!
Type 0 list do not support this notify feature.

1.6 DVC Guide - About

How does it work!

It’s simple, VERY simple!
DVC looks at the file.extension,
and if it matches one of the know extensions.
It will then check the version of the file,
then it will look in the location of the INSTALLED file.
In other words it will check if you already have the given file,
and if you do it will compare versions.

Depending on the situation you will have the option to update
any to any new file, or install if you don’t have the file on your system.
Or replace if the file is the same or older,
and when DVC can’t get the version info.

If a file don’t have an extension or has an unknown extension, then DVC will just display the name and version.

The DVC list format (Type 0, 1 and 2),
Take a look at the LibList.dvc example.
The first line must always start with "DVC0", "DVC1" or "DVC2"
(depending on what type of list it is).
This is so DVC can make shure it is really a DVC list,
lines starting with a ";" or empty lines are ignored,
so they can be used to comment a list, or put blank lines
between diffrent parts of a list, thus making it easier to maintain.

From DVC 2.1, Type 1 & Type 2 lists also have a special comment "$EML:"
Look at
Plugins/TestList.dvc
to see such a comment.
This special ID is followed by an email address ;$EML: emsai@online.no
the $EML: tell DVC to notify the user on new versions of files,
and allow the user to save a small info file that later can be
mailed to the $EML: email address.
Making it easier for list maintainers and users.

DVC0 is identical to DVC1, but only contain filenames,
it is useless as a version database.
DVC use Type 0 lists to check if those files exist,
and if you do not have one of the files, it will tell you so.
It could be used to from a WWW site to notify the user
if he is missing files.

DVC Type 2 lists, are like Type 1.
But also have a Internet URL added after the version number.
For more info on Type 2 lists, see
Plugins/TestList.dvc
.
And
Internet
usage, plus
Plugins/DVC.rexx
.

That’s it! Oh yeah! DVC3-DVC9 are not used yet.
List types 3,4,5,6,7,8,9 are reserved for the future,
but as it stands now, I can’t think of anything else to add.

NOTE!
DVC lists are plattform independant, program independant, system independant.
All lines in a list ends with a LineFeed (ASCII 10),
and the lists are 8bit LATIN-1 (Standard on the Amiga)

But the filenames and version numbers in a DVC list,
may be for a special computer plattform/type.
Or even for a special version of a Operating System.
So those who make lists should make shure that the filenames
have the correct and standard/default filepaths.
And if using Type 2 lists, then the user should know BEFORE clicking
a link, what type of OS or Computer the lists expect.

From DVC v2.0 there is no danger with incorrect filenames etc,

since DVC will silently fail, and go to the next filename,
or simply just quit checking the rest of the list.
So the DVC program should not crash or do anything bad to you,
because some fool created a nasty list ;-)
This also makes it safe, even when lists are corrupt.

And no matter what is IN the lists, the security/safety of your machine,
is the same as when you use your browser. Not strange really,
since DVC use your browser when working with Type 2 list.
And the better/safer your browser is, the better/safer it is using DVC.

And if you trust your browser but not DVC or ME, then don’t use DVC.
The latest version of DVC can always be found on Aminet,
or at the Msi Software homepage.

1.7 DVC Guide - System filepaths

The extensions (DVC’s capability to install), is path dependant.
These are the extensions supported in this version of DVC,
and the expected paths DVC need to perform as was intended.

```
Normal paths:
OS 2.0+ - "#?.library"       - "LIBS:"
OS 2.0+ - "#?.device"        - "DEVS:"
OS 2.0+ - "#?-handler"       - "L:"
OS 2.0+ - "#?filesystem"     - "L:"
OS 2.0+ - "#?fs"             - "L:"
OS 2.0+ - "#?.rexx"          - "REXX:"
OS 2.1+ - "#?.country"       - "LOCALE:Countries/"
OS 2.1+ - "#?.language"      - "LOCALE:Languages/"
OS 3.0+ - "#?.class"         - "SYS:Classes/"
OS 3.0+ - "#?.datatype"      - "SYS:Classes/DataTypes/"
OS 3.0+ - "#?.gadget"        - "SYS:Classes/Gadgets/"
OS 3.0+ - "#?.image"         - "SYS:Classes/Images/"
OS 3.0+ - "#?.codec"         - "SYS:Classes/Codecs/"

Sub libraries:
OS 2.0+ - "xpk#?.library"    - "LIBS:Compressors/"
OS 2.0+ - "SPlay_#?.library" - "LIBS:smartplay/"
OS 2.0+ - "rtg#?.library"    - "LIBS:rtg/"

Sub paths:
OS 2.0+ - "#?.audio"         - "DEVS:AHI/"
OS 2.0+ - "#?.spobject"      - "LIBS:spobject/"
OS 2.0+ - "#?.svdriver"      - "LIBS:svdriver/"
OS 2.0+ - "#?.svobject"      - "LIBS:svobject/"
OS 2.0+ - "#?.svoperator"    - "LIBS:svoperator/"
OS 2.1+ - "#?.accent"        - "LOCALE:Accents/"

Developer paths:
OS 2.0+ - "#?.fd"            - "FD:"
OS 2.0+ - "#?.lib"           - "LIB:"

Custom paths:
```

```
OS 2.0+ - "#?.mui"           - "MUI:Libs/MUI/"
OS 2.0+ - "#?.mcc"           - "MUI:Libs/MUI/"
OS 2.0+ - "#?.mcp"           - "MUI:Libs/MUI/"
OS 2.0+ - "#?.ma"            - "MainActor:rexx/"
OS 2.0+ - "#?.pprx"          - "PPaint:Rexx/"
OS 2.0+ - "#?.gio"           - "photogenics:Data/GIO/"
OS 2.0+ - "#?.giose"         - "pgsse:Data/GIO/"

Custom libraries:
OS 2.0+ - "mui#?.library"    - "MUI:Libs/"
OS 2.0+ - "pgs#?.library"    - "photogenics:Data/libs/"
OS 2.0+ - "pgsse#?.library"  - "pgsse:Data/libs/"

Custom classes:
OS 2.0+ - "mui#?.class"      - "MUI:Libs/"

Custom images:
OS 2.0+ - "mui#?.image"      - "MUI:Images/"
```

(I need more paths, I you see a file or path missing let me know)
(I don’t have much software, so many paths/files are unknown to me)
(So please let me know, so I can add it later)
(I’m VERY interested in paths that have many files like MUI etc)
(and if any IMPORTANT AND COMMON paths are missing THEN TELL ME)
(there is only so much I can do alone, I need your help...)

Here is something that is nice to know!
If the file DVC check have a unknown extension or no extension at all,
then it will bring up a GUI with a "Choose Path" button.
Just click that button and choose a path in the ASL filerequester,
DVC will then use that path when Installing/Updating/Replacing.
If the file exists it will compare versions, nice huh? :-)

1.8 DVC Guide - History

```
 v1.0 - First release!

 v1.1 - Added 4 more extensions!
        Fixed an old bug in the version routines,
        now more files will have correct version checks.
        DVC now handles version strings with "v" infront of the
        version number, but this is the only non-standard version
        strings I will let DVC handle, if DVC fail, blame the
        programmer of that file, tell him to get a proper $VER: ;-)
        DVC is now able to handle xpk and smartplay sub libraries!

 v1.2 - No public release, became v2.0 instead!

 v2.0 - (6.March.1997) Added 7 more extensions!
        Improved version checking even more, I said I would not :-(
        Now files with REALLY bogus version info turns out good.
        Some of the xpk sub libraries have VERY bad versions strings.
        So DVC now handle strings that end right after the version number.
        You folks wouldn’t belive how rotten some version strings are ;-)
```

```
       Some programmers even use nonblank space (ALT SPACE) as a space (!)
       At the time of release, DVC has one of THE BEST version checks.
       But please remember that DVC work by NUMBERS, not by STRINGS.
       And thus DVC and a util like VersionWB may argue on the version,
       DVC nor VersionWB is perfect because there is so many bad infos.
       DVC rely a lot on compilers doing it right,
       several compilers don't do it right on libs/devices etc...
       WB support added, and DVC check if it is already running.
       If started from CLI, DVC can be stopped most of the time
       by pressing Ctrl-C, or if DVC was started with run
       the BREAK command can be used.
       In some situations Ctrl-C won't work,
       this is for safety reasons (avoid trashing of memory/files).
       The WB GUI is activated when starting with no CLI arguments.
       Plus a bunch of internal changes.
       DVC is now able to handle mui libraries/classes & sub libraries,
       plus handlers and filesystem files located in L:
       DVC now ask the user depending on the situation if he want to
       Install/Update/Replace, easier to notice what's going on ;-)
       Files with unknown or no extension is displayed with version info,
       so DVC can also be used as a Version util (date not shown).
       Guide documentation, file/drawer icons etc added to archive.
       List (0,1,2) support added. Info on how to use DVC with Internet.
       Example list, and MakeList script included.
       A second MakeList script is also included,
       along with a script for using DVC with Internet browsers,
       and a test list for browsers.
       DVC is also able to handle most FileSystem and Handler files in L:

v2.1 - No public release!

v2.2 - (21.May.1997)
       Added email notify requester and "$EML" support
       for Type 1 & Type 2 lists.
       Added SYS:Classes/Codecs.
       IBrowse info added to Guide.
       And some minor changes.

v2.3 - (30.August.1997)
       $EML: support for Type 1 and Type 2 lists have been extended,
       DVC now have an internal mailer "DVC Mail 1.0",
       If TCP: is running and connected to a ISP
       you can report on new versions directly to a list maintainer.
       Plus some internal improvements.
       Removed a possible namebug that probably have
       existed for a very long time (caused strange filenames).
       (Where's the bug reports people? Or is DVC perfect? :-)
       Implemented a CLI/WB Move option,
       the selected file (original/source) will be removed
       after a sucessfull install/update/replace.
       Added rtg sub libraries, .fd and .lib files.
       Added .rexx, .ma, mui .images, .pprx, and PPaint lib files.
       Added Photogenics .gio, .giose and pgs, pgsse lib files.
       Fixed a problem when trying to copy the same file,
       now DVC will notify the user and abort copy.
       Improved version routine (insane, it's over 4KB large :-(
       DVC will notify the user of files with bad version info,
       this is to alert her/him that DVC may not show correct info.
       A filecomment will be made when possible to show that
       the file has a bad version info.
       DVC has become modular, general functions moved out of
       the DVC source, program maintainance has become easier.
       It seems LoadSeg() function used in version routines,
       generate a memory hit when UnLoadSeg() is used
       on arp.library, you have been warned (c:version also have trouble)
       DVC is able to update/replace programs stored in the
       default command path (path command in startup-sequence).
       Added support for update/replace for system command files.

v2.4 - (4.October.1997)
       Seems like when I moved the version check stuff into a seperate
       module, some code from the old routines was left in DVC,
       mainly a check if a Ctrl-C was received while version checking,
       but the version module no longer support that,
       so DVC thought a Ctrl-C was received when a VERY BAD version info
       was found, not a surprise really, bad version info has caused
       me trouble before, and for progs like VersionWB etc.
       DVC should no longer stop list checks on very bad files.
       There is also some other minor changes, but can’t remember them :-)
       Also added a "Done!" requester, so you know when a list is finished!
       Added some error messages to DVC list parsing!
       The main page of the guide said v2.9 (!) Fixed it!
       Improved the Msi Version module, I hope I never have to see that
       damn code again, these version strings drive me crazy :-)
       Heh! DVC can handle files like the handler AmiCDFS, CacheCDFS etc.
       DVC will know make it possible to "Choose Path" if no extension,
       or if unknown extension, then it will pretend that the path
       was found by DVC, and do a normal install etc. on it :-)
       It will even put a suggested path in the requester, WOW! :-)

v2.5 - (5.October.1997)
       Damn I hate this, the day after a release I find a bug :-(
       Fixed a problem when installing files with unknown extensions.
       Sorry to those who have to download it again :-(
```

1.9 DVC Guide - Thanks

```
Heiko Schröder - <s93324@fh-telekom-leipzig.de>
                 The LibGuide maintainer, for requesting a WB interface.
                 And thus causing my brain to invent alot of other stuff.
                 Thus forcing me to bump the version from 1.2 to 2.0 ;-)
                 He also made me include "sub.extensions.support" ;-)
                 He also suggested the "new file" notify requester.
                 And also been a great beta tester, thanks Heiko.

Phil Beazley - <midnight@treknet.net>
               Reported the namebug, and kept on nagging me :-)
               until I fixed it! thanks Phil.
               He also suggested the MOVE option.

 (And a BIG thanks to everyone on the Msi Betatester mailinglist)
```

1.10 DVC Guide - Future

```
Let DVC handle more system files ;-)

So send me the name of a file DVC can't handle,
make shure you give me the real path and full filename.
I.e: name.extension, Full:File/Path/For/Drawer
For xpk sub libraries this would be:
xpkBLZW.library, LIBS:Compressors

I will then let the next DVC support the new files,
and put your name in this document for all to see ;-)


I would also like to make DVC more user friendly.
But the first step would be a easier to undrstand documentation,
anyone feel up to writing a "The DVC Beginners Introduction" ???
```

1.11 DVC Guide - Author

```
For bug reports/comments/ideas etc, write to...

   Roger Hågensen          Email: "Msi Software <emsai@online.no>"
   Msi Software              Www: "http://www.sn.no/~msi/index.htm"
   Kolstadflata 35B
   N-7078 SAUPSTAD
   Norway

With a subject like: "DVC 2.4"

Updates of this software, and other software
can be downloaded directly from the above WWW address.
```

1.12 DVC Guide - Distribution

```
DVC is Freely Distributable FreeWare,
it is illegal to make profit on DVC without my permission.

Only a small copying fee is permitted.

This also includes inclusion in ShareWare/SemiCommercial & Commercial,
and cover disk's/cover cd's and collection cd's.

Aminet and Fred Fish have my permission to distribute DVC as they wish,
since I trust them and wish others was more like them.
```

1.13 DVC Guide - Copyright

DVC and it’s documentation and archive may not be altered and
re-released by anyone else than the original programmer,
unless authorized to do so by the original programmer.

This software and releated names/documentation/logos,
are copyright Msi Software 1997.

Some names in this document are Trademarks of companies/persons,
and used as reference only, and not an attempt at infringement.

1.14 DVC Guide - Disclaimer

This software is provided AS IS,
and while it has been tested and is used by the programmer himself.
This do not guarantee reliability or that it will work at all,
so you use this software at YOUR OWN risk.

1.15 DVC Guide - Note

To programmers and other people to whom it may be of interest!

Only files that have proper version strings will have correct version
info, if you are a programmer, look at the version string of DVC ;-)
It has a name (with no spaces), and version.revision separated by a dot ".",
and the date enclosed in () in this format: day.month.year
day may be 1-31, and month may be 1-12, year may be 0-99.

Stay tuned for "ver.library" it will introduce VersionSpec2
and explain VersionSpec1.
This library also have all the version functions a programmer need,
and if anything is missing it will be included :-)

1.16 DVC Guide - Internet

If you use AWeb 1.6 or 2.1 for Internet, then set up DVC in AWeb like this:

```
Browser/Viewer settings
-----------------------
      Type: TEXT/X-DVC
Extensions: dvc
   Command: sys:rexxc/rx
 Arguments: Plugins/DVC.rexx %f %n
```

Then just copy DVC to the C: directory.
Now you can click on a DVC link on a page somwehere on the Internet,
and DVC will be started by your browser and will check if you have the
latest versions of the files in a list.
And if you have an old version, depending on the type of DVC script,

you may choose if you want to download the latest version from Internet.

For AWeb 2.0 you should set DVC up like this:

```
Mime types and external viewer settings
----------------------
      Type: TEXT/X-DVC
Extensions: dvc
   Command: sys:rexxc/rx
 Arguments: Plugins/DVC.rexx "%s" "%s"
```

If you use IBrowse 1.1 for Internet, then set up DVC in IBrowse like this:

```
Mime configuration
-----------------------
      Type: TEXT/X-DVC
Extensions: dvc
   Command: sys:rexxc/rx
 Arguments: Plugins/DVC.rexx %f %p
```

The DVC.rexx plugin script, can also be used with other IBrowse versions
and also with Voyager. But DVC has only been tested with AWeb.

How does it work?

DVC simply look in a DVC list ".dvc" and if it is of
Type 2
,
it contains HTTP/FTP url’s, that later will be sent to your browser
if you choose to download the new version of a file.

DVC only check the list and let your browser do the downloading,
but sometimes it may also tell your browser to go to a homepage.

So any trouble with the lists should be notified to those
who have created those lists.

1.17 DVC Type 2 list

```
DVC2 (Don’t change this line!)
;$VER: TestList.dvc 1.1 (12.4.97) Roger Hågensen
;$EML: Webmaster <webmaster@www.net>

;Load this file into your browser as a local file,
;or make a proper link to it from a HTML page.

;Below a fake version number is used to simulate/test
;if everything works,
;if it does then the browser will copy amigaguide.library
;from LIBS: to your default download directory.

LIBS:amigaguide.library 99.9 file://localhost/LIBS:amigaguide.library
```

```
;Below a fake version number is used to simulate/test
;if the NEW function works.
;If it does it will let the user save a small file
;to Ram: this file can then be sent to the webmaster
;so he can easily update his files/info.
;This is the same as for Type 1 lists.

LIBS:amigaguide.library 0.1 file://localhost/LIBS:amigaguide.library

;Please note that file:// and file://localhost works,
;and that mean any HTTP:// url either to a file/homepage etc,
;or a FTP:// url may also be used, even mailto: etc, could be used ;-)
;DVC simply pass the url (anything after the version number),
;to a browser with the help of sys:rexxc/rx
;DVC has a special browser option called RX,
;this allow a script/program or you to
;tell DVC what portname DVC should give to the sys:rexxc/rx util,
;and also what Load/Open command shoud be sent to the browser.
;This way virtually any browser/program can make use of DVC,
;one way or another.
```