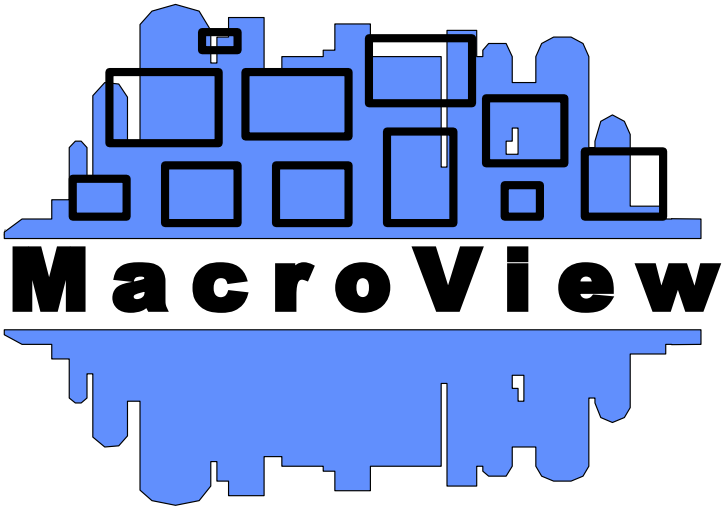


MacroView Release Notes

Version 3.12.2
20/04/2009

Table of Contents

Part I Introduction 2

Part II Version 3.10 2

1 Version 3.10.1 ... 2
2 Version 3.10.2 ... 2
3 Version 3.10.3 ... 3

Part III Version 3.11 3

1 Version 3.11.1 ... 3
Recent Message File Format - Version 1 ... 4
Recent Message File Format - Version 2 ... 4
2 Version 3.11.2 ... 5
3 Version 3.11.3 ... 6
4 Version 3.11.4 ... 6
5 Version 3.11.5 ... 7
6 Version 3.11.6 ... 7
7 Version 3.11.7 ... 9

Part IV Version 3.12 9

1 Version 3.12.1 ... 9
2 Version 3.12.2 ... 10
3 Version 3.12.3 ... 12
4 Version 3.12.4 ... 12
5 Version 3.12.5 ... 13

Part V Version 4.0 13

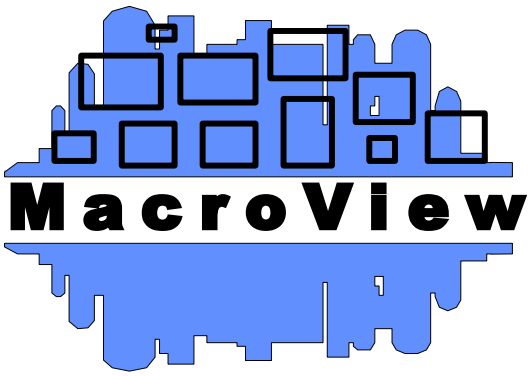
1 Version 4.0.0 ... 13
2 Version 4.0.1 ... 14
3 Version 4.0.2 ... 14
4 Version 4.0.3 ... 15

Part VI Notes 15

1 Windows Services ... 15
2 STATUS2R Attribute Format ... 15
3 Metascript MessageQueue Support ... 16

Index 17

1 Introduction



This document contains release notes for MacroView Releases from Sentient Computing starting from November 2004. The format of document is based around many short text descriptions of the version changes that have occurred since Version 3.10 of MacroView. As such, it is very dry reading but a useful change reference document. The release notes cover all the MacroView platforms supported by Sentient:

- Linux on x86 processors
- Windows XP/2003
- SCO Open Server
- Solaris 8, 9 and 10 on Sparc processors
- Tru64 Unix on Alpha processors

2 Version 3.10

2.1 Version 3.10.1

7th November 2004

The first formal release of MacroView on Microsoft Windows XP/2003 by Sentient Computing. The remainder of the release notes in this document use this release as a base line.

2.2 Version 3.10.2

Windows Packaging Modifications:

- Modified the demonstration configuration's Services.xml file to contain an alarms service definition by default.

mvmsgs 3.10:
7th November 2004

- Modified the debug output content to be less verbose and hence more useful in documenting message processing actions.

pingdrv 1.5:
3rd January 2005

- Modified the scheduling mechanism for ping results scans so that the entity scan rate defines the attempted scan rate when an entity value is in demand. This makes it more practical to use the ping driver functionality as the previous version would result in excessive ping traffic on the network when entity values were placed in demand.

2.3 Version 3.10.3

23rd January 2005

MacroView Windows Service Manager:

- Fixed a problem introduced in Version 3.10.2 where a new service could no longer be created in the service manager.
- Included Batch files (.bat) in the file filter for selecting Executables in the MacroView service manager.

3 Version 3.11

3.1 Version 3.11.1

xops3/ops3 2.3.3:
13 February 2005

- Fixed a problem where dynamic overlap support was not working when a variable set event caused a dynamic object to be updated.

mvmsgs 3.11:
29th March 2005

- Modified to use a new file format for the recent message file. See the notes in Version 2 of the Recent Message File Format definition for the reasons a new file format version was introduced.
- The new recent message file format has a backward compatible header structure so it can be used with older xops3/ops3 executables. It separates the latest into alarm message from the latest alarm message of any form.
- Older xops3/ops3 executables (before V2.4.0) will only display INTO alarm messages when used in conjuction with the default operation of mvmsgs V3.11 and higher.
- Added a new command line option -oldRecentMsg which instructs mvmsgs to use Version 1 of the Recent Message File Format definition *and* the associated older message processing functionality. This command line option is only made available for total backward compatibility. It has the disadvantage of a possibility that the xops3 alarm buzzer will not sound if an OUT OF alarm message is received immediately after an INTO alarm message is received. This only applies if xops3/ops3 alarm buzzer functionality is being used. The problem does not exist for console based entity buzzer functionality. As such, the -oldRecentMsg command line option should only be used when xops3/ops3 alarm buzzer functionality is *not* being used.

xops3/ops3 2.4.0:
29th March 2005

- This executable version resolves xops3/ops3 alarm buzzer functionality issues that were identified. The issues are only relevant for systems that rely on the local xops3/ops3 audible alarm functionality. An upgrade is not needed for systems that use the global console entity buzzer functionality.
- Modified to read both Version 1 and Version 2 of the Recent Message File Format.
- When working with a Version 2 recent messages file, the alarm line functionality has changed from previous versions such that:
  - Recent message data polls won't miss the occurrence of an into alarm message followed immediately by an out of alarm message. The into alarm messages will be detected in preference and result in the xops3/ops3 alarm buzzer sounding when an alarm line is configured for local buzzers.
  - The xops3/ops3 alarm buzzer will sound again if a new into alarm message is received with the identical message text to a previous occurrence.

xops3/ops3 2.4.1:
7th April 2005

- Modified the Clear message processing for the alarm line widget to fully zero the recent.msg file. This fixes an issue where clearing the alarm line and then restarting xops3 would show portions of

previous messages.
- Modified the alarm line widget to not sound the buzzer on start up if the last message was an out of message.

xops3/ops3 2.4.2:
mshell 1.14:
exproc 1.10:
setval 1.9:
10th April 2005
- Fixed a rounding problem when setting a PLC model DECIMAL formatted type attribute value. Previously a set request to 0.29 would incorrectly result in a set action of 0.28 in a particular configuration scenario.

3.1.1 Recent Message File Format - Version 1

In Version 1 of the mvmsgs recent message file format, the recent.msg file contained the following data:

Color<US>*Priority*<US>*Message*<US>*Superfind*<NUL>

Color	The color of the message as a MacroView color index.
<US>	The ASCII Unit Separator character (decimal 31).
Priority	The priority of the message (0-5).
Message	The message text.
Superfind	A superfind string used to locate a graphic page associated with the message.
<NUL>	The ASCII Null character (decimal 0).

Version of the recent message file format was used by mvmsg Version 3.10 and earlier.

3.1.2 Recent Message File Format - Version 2

In Version 2 of the mvmsgs recent message file format, the recent.msg file contains the following data:

IntoColor<US>*IntoPriority*<US>*IntoMessage*<US>*IntoSuperfind*<NUL>
FileFormatVersion<*US*>*IntoMessageID*<US>*Color*<US>*Priority*<US>*Message*<US>*Superfind*
<NUL>

IntoColor	The color of the last into alarm message as a MacroView color index.
<US>	The ASCII Unit Separator character (decimal 31).
IntoPriority	The priority of the last into alarm message (0-5).
IntoMessage	The message text from the last into alarm.
IntoSuperfind	A superfind string used to locate a graphic page associated with the last into alarm message.
<NUL>	The ASCII Null character (decimal 0).
FileFormatVersion	The recent message file format version number e.g. 2.
IntoMessageID	An integer identifier for the into alarm message defined in the first part of the file. This integer starts at a value of 1 when the message processor starts and continuously increments for each message processed.
Color	The color of the last message as a MacroView color index.
Priority	The priority of the last message (0-5).
Message	The last message text.
Superfind	A superfind string used to locate a graphic page associated with the last message.

Notes:
- The line separation used in the file format specification at the top of the page is for clarity only. The file format doesn't use line feed or carriage return characters for field separation.
- An "into alarm" message is interpreted as any alarm message that is not explicitly an "OUT OF" alarm message. Messages that don't define an INTO or OUT OF state are treated as INTO alarm messages.

- The separation of the recent message information into (a) the last into message and (b) the last message of any form was done to resolve an xops3/ops3 alarm buzzer problem. The ALARM LINE graphic object polls the recent.msg file (in the msgs directory) for changes. If multiple changes occur between polls and xops3/ops3 alarm buzzer functionality is being used, then the alarm buzzer may not sound. The Version 2 file format resolves this problem for xops3/ops3 executables both before and after V2.4.0. This is achieved by having a backward compatible header structure that only includes INTO alarm messages. OUT OF alarm messages are included in the second part of the file format, but will not be seen by xops3/ops3 versions before V2.4.0.

3.2 Version 3.11.2

18th April 2005

xops3/ops3 2.4.3:
18th April 2005

- Added the "WaitForModalWindows" and "DontWaitForModalWindows" message processing support for the "MetaFile" destination. The default operation for metafile windows is that metascripts initiate the display of a metafile window and then immediately return to executing the remainder of the script. Sending the "WaitForModalWindows" message to any "MetaFile" will change modal window handling for the entire operations session to halt the script until the modal window has been displayed and closed. Once the modal window has closed, the remainder of the initiating script is executed. This functionality was only added for backward compatibility where existing systems relied on the behavior and the source code could not easily modified. The preferred approach is to not rely on modal window display to pause scripts.

alarms 2.11:
27th April 2005

- Fixed a problem where analog value alarm checks (low, low-low, high, high-high) would not be enabled when particular constant numeric limits were configured. The problem occurred for particular numeric values and did not occur when comparing against another attribute such as PH or PL.

xops3/ops3 2.4.4:
29th May 2005

- Fixed a problem where the alarm buzzer incorrectly sounded in the following scenario:
  - an into alarm was received and the buzzer sounds as expected.
  - the alarm line cleared and the buzzer stops sounding
  - an unrelated out of alarm message was received. The original into alarm message appears and the buzzer sounded again. This problem no longer occurs in V2.4.4.

xops3/ops3 2.4.5:
13th June 2005

- Added the -iocount command line option. This reports the IO and entity count of the current configuration. This is handy to see how a configuration is growing and whether it will soon exceed the licensed IO or entity count.

alarms 2.12:
13th June 2005

- Added the -printonly command line option. This is intended for support and maintenance only as it turns off the transmission of alarm messages to the message queue and only reports the alarm messages as print systems to the standard output.
- Added the -scanonce command line option. This causes the alarms program to perform one alarm scan and then exit. Again this is intended solely for support and maintenance purposes.

3.3 Version 3.11.3

29th August 2005

Added support for the Digital Unix V4.x operating system on Alpha processors (known now as Tru64 Unix) to the Sentient source code base.

xops3/ops3 2.4.6:
29th August 2005

- Modified the minimum color table index allowed to be 16 instead of 17.
- Modified the metafile header processing logic to handle background colors defined in a color table included later in the metafile. Previously the metafile header couldn't be used to reference non-standard background colors.
- Added support for the STATUS2R Attribute Format functionality.

getval V2.1:
29th August 2005

- Added support for the STATUS2R Attribute Format functionality.
- Modified the default entity/attribute naming format to use an underscore as a separator. This matches the original getval default and the current setval default.

alarms V2.13:
mvdded V2.6:
exproc V1.11:
mv_histr V4.2.2:
setval V1.10:
mstracer V2.3:
webrep V1.8:
29th August 2005

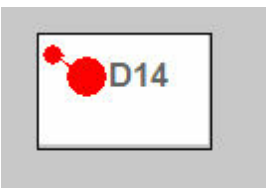
- Added support for the STATUS2R Attribute Format functionality.

3.4 Version 3.11.4

15th September 2005

xops3/ops3 2.4.7:
15th September 2005

- Fixed a dynamic update overlap problem that occurred in scenarios where 2 dynamically updating objects are placed on top of another larger dynamic object or objects. If we call the 2 smaller dynamic objects A1 and A2 and the larger underlying dynamic A3, the problem occurred when A1 updated its state. Note that A1 and A2 do not overlap, but both A1 and A2 overlap A3. Previously when A1 was updated, the underlying A3 would be refreshed and then A1 would be updated. A2 would not be refreshed causing it to disappear. The problem has been resolved in this revision. In the following graphic example, A1 would be circular graphics, A2 the text and A3 the rectangle below them both. Note the rectangle dynamically changes its color based on external inputs.



eng3:
- Added support for the STATUS2R Attribute Format in the type attribute format combo box.

ops3 2.4.8:
20th September 2005
- Fixed a windows specific file locking problem that occurred when large numbers of ops3 sessions concurrently displayed a chart of the same history file section. In this scenario, an ops3 session would occasionally exit with a fatal error. This version fixes that read lock problem.

mv_histr 4.2.3:
27th September 2005
- Modified the write locking implementation in the Windows version to match that used for read locking in ops3 V2.4.8.

EzEng:
- Added support for editing color change objects within the EzEng environment.

3.5 Version 3.11.5

24th October 2005

mshell 1.15:
ops3/xops3 2.4.9:
mvmsgs 3.12:
webrep 1.9:
24th October 2005
- The display precision is retained along with any numeric value within the metascript execution environment. It is the number of decimal places to display when converting a numeric value to a string. The display precision of the results of a calculation is typically defined by the maximum precision of the input values for the calculation, except for numeric power operations (^). The display precision increases with every power operation to allow for the precision needed to display the resulting value. This caused the display precision to grow continuously if a variable is involved in a long power operation loop. On Unix systems the application program would then fail when converting the numeric variable to a string and the numeric precision was very large. To resolve this problem, a limit of 32 decimal places is now used for the display precision of metascript variables.

xops3 2.4.10:
24th November 2005
- Added support for the "Viewer" destination in metascript SEND commands. The viewer destination only applies to the scenario where xops3 is running in a VNC session connected to a MacroView smart client viewer application. It allows the xops3 metascript to instruct the viewer to display report tabs and other such client side functionality. For example, the command SEND "Display(daily.rep)" TO "Viewer" is an instruction for the viewer to run and display the daily.rep report in a tabbed window. Note that at the time of writing (24th November 2005), the MacroView windows viewer functionality is not a released product.

3.6 Version 3.11.6

24th April 2006

ops3 2.4.11:
12th December 2005
- Fixed a display problem with ellipses on Windows system. The display problem related to the size of the ellipse and whether it was displayed.

ops3 2.4.12:
mshell 1.17:
13th December 2005

- Added the ability to send MacroView messages directly from metascript code into the message queue. Previously an external program such as the sendmsg utility would have to be used. See the Metascript MessageQueue Support section for details on the command syntax.

ops3 2.4.13:
13th December 2005

- Fixed a problem with the warning level message generation in the Metascript MessageQueue Support functionality.

mvmsgs 3.13:
12th February 2006

- Modified so that the metascript variables set before the message.ms script is run are reset to empty strings or zero values immediately after the script is run. This resolves issues associated when extra fields are added to the alarm specification database beyond those supplied as standard. A problem could potentially occur when processes other than the alarms process generate messages that don't contain the extra field definitions. Previously message.ms scripts could not tell if the extra field variable values were associated with the current message or just "residue" from the last alarms generated message. Now mvmsgs ensures that variables are reset to empty string or zero values after the completion of the message.ms script.

alarms 2.14:
12th February 2006

- Added the -almspec command line option. When present, extra alarm specification field values are added to every message generated by the process. The extra field values are almrec, ALMTYP, EXPRN, DEADBAND, GROUP and DELAY. The almrec field value is the record number of the alarm specification record that defines the alarm. The other field values are alarm specification fields that aren't included in the default alarm information embedded in each message. Note that mvmsgs provides these fields as lower case variables for processing within the message.ms script e.g. almrec, almtyp, exprn, deadband, group and delay.
- Removed the -almrec command line option from the version banner display. For backward compatibility specifying -almrec on the command line is processed as if it were a -almspec command. This retains backward compatibility since almrec information is included in that mode.
- In prior versions of alarm program, a %s entry could be placed in the alarm specification MESSAGE field and it would be replaced by the current alarm value when the alarm message is generated. The ability to include the previous value can now be included by including an additional %s entry.
- For example:
- A Change alarm with a configured MESSAGE field of "Voltage Value=%s Previously=%s"
- Can result in a generated alarm message of the form: "PV(TESTINT) CHANGED (Voltage Value=12 Previously=10)"
- The alarm specification and alarm message databases are now opened in read-only mode since they are never modified (directly) by the alarms program. This allows it to be used in scenarios where the database files are marked as readonly (as test configurations in source control systems) or owned by another user.

ops3 2.4.14:
4th April 2006

- A problem was encountered with the Navigator area alarms implementation on a Windows Server 2003. The problem related to the acknowledgment entity value for an alarm area not being set when an alarm was acknowledged. The acknowledgement entity is optionally configurable as part of the areas.dbf configuration database. It was identified that the ops3 behaviour for table variable lifetime management was different for the Windows and Unix implementations. This resulted in a table variable related to area alarm information going out of scope before it could be used in the Navigator alarm acknowledgement code. The windows implementation was modified to be consistent with the Unix implementation to resolve this issue.

mshell V1.18:
ops3 2.4.15:
4th July 2006
- Fixed a problem where INSERT row statements would fail after the field count requested was greater than 32. This problem was introduced in MacroView V3.11.5 as part of the precision check modifications.

mshell V1.19:
1st May 2006
- Added the -version command line option which reports the MacroView product version number and then exits.

mvmsgs V3.14:
13th March 2006
- Modified so that any LET actions on message variables in the message.ms script actually modify the values that are placed into the associated message database (alarms.dbf, sysalarms.dbf, guides.dbf and histmsgs.dbf). This feature provides flexibility in tailoring the message processing to site specific needs.

3.7 Version 3.11.7

alarms 2.15:
17th May 2006
- Fixed a memory allocation error that only occurred when extra alarm fields are used via the -almspec command line parameter.

ops3 2.4.16:
27th June 2006
- Fixed the SPRITE metafile object so that it works as required. This was a rarely used metafile object that implements a combination of color, shape, positional and scale change functionality based on the execution of metascripts. Historically, the introduction of the METASPRITE implementation actually broke the SPRITE implementation. A recent upgrade of an older MacroView site highlighted this issue which was resolved for the site upgrade.

4 Version 3.12

4.1 Version 3.12.1

29th June 2006

xops3/ops3 2.4.18:
mshell 1.20:
keyword 1.5:
webrep 1.10:
mvmsgs 3.15:
alarms 2.16:
mvdded 2.7:
exproc 1.12:
getval 2.3:
mv_histr 4.2.4:
setval 1.11:
5th September 2006
- Added support for the ENGUNIT field within the typeattr.dbf database. This allows engineering units to be defined on a per attribute basis. To enable this functionality modified the typeattr.dbf field structure to have a Character format ENGUNIT field that is 12 characters wide to your existing type attribute field structure. It must be added after the CONFIRMFLG field.

- Added support for the ENGUNITS() function to the metascript language. This function takes a *string* as an argument where the string defines an attribute of an entity. It can be of the form "ENTITY.ATTR" or "$.ATTR". It returns the engineering units for that particular attribute of the entity type as defined in the typeattr.dbf ENGUNIT field. If the type attribute engineering units field is blank then the entity ENGUNIT field value is returned.
- Extended the allowed size of the entities database ADDRESS field to be up to 100 characters. Increased address sizes allow deeper PLC addressing levels to be defined. To enable this functionality modify the entities database structure so that the ADDRESS field size is up to 100 characters wide. The field size should not be set to greater than 100 characters otherwise program failures will occur.
- Modified the metascript engine to return "3.12.1" for the MacroView VERSION().

Important Notes:
- The entire standard MacroView executable set will need to be upgraded to Version 3.12.1 or higher to be allowed to make the ENGUNIT structural change to the type attribute database. If this change is made to MacroView versions prior to 3.12.1 then the programs will exit and display an error messaging indicating that the type attribute database could not be opened.
- Third party executables that access MacroView data by linking to C libraries will need to be upgraded to work with the type attribute and entity database structural changes. Examples of such executables include in-house MacroView application executables built with the Fortran/C development kit and the Matrikon OPC server for MacroView. Third party executables only need to be upgraded if the database structure changes documented here are performed on a configuration.

4.2 Version 3.12.2

11th September 2006

xops3 2.4.19:
11th September 2006
- Fixed the implementation of the metascript SYS("WindowID") function call to correctly return the X window identifier on Unix/Linux systems. This functionality did work in much older MacroView versions and is used to implement screen capture scripts for Unix systems.

alarms 2.17:
28th November 2006
- Modified so that change alarms aren't raised when alarms is restarted and there is a change alarm already in the alarm messages database. Recent alarms executable versions would generate a CHANGE alarm on startup or restart if there was a change alarm entry in the alarms database.

xops3 2.4.20:
9th January 2006
- The implementation of relative offset attributes (e.g. DBFENT.KW[2]) was broken in the previous version as a result of a related functionality change. The relative offset attribute functionality was fixed in this version.

xops3 2.4.21:
22nd February 2007
- Fixed an issue with trend/chart displays where dynamically updating trends would have chart line drawing residue when a bitmap shift occurred.

xops3 2.4.22:
1st March 2007

- Fixed a problem where values from the wrong record could potentially be returned from view attributes if multiple views accessed the same underlying dBase file concurrently within the same xops3 session. The problem was introduced by changes in V2.4.20.

xops3 2.4.23:
mshell 1.21:
6th July 2007

- Type conversion warnings in metascripts now also display the line number and text of the metascript code that generated the warning.

alarms 2.18:
24th August 2007

- Modified the expression alarm implementation to not initiate an alarm state change if any of the entity.attribute values referenced in the expression have a -999 bad value state and the -nobadval command line option has been specified. This allows communication errors to not cause alarm state changes in expression alarms.
- Modified the alarm expression engine to be more lenient when performing arithmetic comparison operations between numeric and string value types. Arithmetic comparison operators include equals, less than, greater than, less than equals and greater than equals. Previously an attempt to compare a numeric and a string value would result in an expression error. This had caused alarm determination issues when the entity.attribute values being referenced were string values and a communication error caused a change to a -999 state.

xops3 2.4.24:
mshell 1.22:
1st January 2008

- Resolved a field error that occurred when the last field defined in a dBase file was a DATE field.

xops3 2.5.0:
mshell 1.23:
16th January 2008

- Resolved a daylight savings issue where if a metascript timestamp variable was initialized to a point in time which is in a different daylight savings mode to the current time, then an hour offset erroneously came into play. The issue related to a daylight savings flag in the mktime Unix/Linux system function call.

xops3 2.5.1:
mshell 1.24:
11th April 2008

- Added the ATTRADDR() function to the metascript language. The function takes a string that is expected to contain an ENTITY.ATTRIBUTE name. It returns the data source address of the attribute. If the attribute has no explicit address in the data source then an empty string is returned. This occurs in many sources as the attribute is associated with the entity and requires no more addressing information than that found in the entity's address. For PLC data sources, the attribute address will be the memory location in the PLC associated with the attribute e.g.
- Resolved a problem in the CREATE HISTORICAL VIEW command where it would fail with a parameter evaluation error if an invalid entity name was referenced in a column definition. Now the view is created with bad values in the columns that don't reference valid entities.

4.3 Version 3.12.3

20th May 2008

Ported MacroView to be able to run on Solaris 10 and 11 for Intel x86 processors. The updated executables and their revision numbers are listed in the table below.

Executable	Version Number
alarms	2.19
archfile	1.9
AutoRestart	3.5
dbfhist	2.3
dbftool	1.0.2
debugout	1.6
dxfdgm	2.11
exerver	1.2
exproc	1.13
getval	2.4
gwclient	1.15
gwserver	1.8
histraw	1.6
histtrim	1.2
imgstat	2.1
inctype	1.5
keyword	1.6
localimg	1.10
localsrc	2.8
mshell	1.25
mstracer	2.4
mvcat	1.9
mvdded	2.8
mv_histr	4.2.5
mvledit	1.7
mvlmd	1.6
mvmsgs	3.16
osinfo	1.6
pingdrv	1.6
plcstat	1.10
re_histr	1.9
sendmsg	1.6
setarea	1.8
setval	1.12
testterm	1.9
texthist	1.8
webrep	1.12
wlayout	1.0.4
xops3	2.5.2

4.4 Version 3.12.4

1st September 2008

Version 3.12.3 introduced Solaris x86 support. A fault was identified in the history files created and read by Solaris x86 Version 3.12.3 executables in that the files were incompatible with MacroView history files from other platforms such as Linux, Windows, SCO and Solaris Sparc. Version 3.12.4 executables no longer have this fault. Version 3.12.3 on Windows included a dependency on Version

3.5 of the .NET Framework in its setup.exe install file. The actual dependency is on Version 2.0 of the .NET Framework and the Version 3.12.4 setup.exe file has been modified to suit.

As a preparation for MacroView Version 4.0, internal buffer size changes were made in a core library that is re-used across many executables. This change necessitated a version number increment on all dependant executables.

Executable	Version Number
alarms	2.20
archfile	1.10
dbfhist	2.4
dxfdgm	2.12
exproc	1.14
getval	2.5
histraw	1.7
histtrim	1.3
imgstat	2.2
inctype	1.6
keyword	1.7
mshell	1.26
mstracer	2.5
mvdded	2.9
mv_histr	4.2.6
mvmsgs	3.17
plcstat	1.11
re_histr	1.10
sendmsg	1.7
setarea	1.9
setval	1.13
texthist	1.9
webrep	1.13
xops3	2.5.3

4.5 Version 3.12.5

7th September 2008

xops3 2.5.4:
7th September 2008

- Modified the licensing check process to check that both the TAGS ***and*** IOPOINTS licensed limits. The previous licensing mechanism would allow the operations program to proceed if either the TAGS or IOPOINTS limits were met. This modification brings the operations program's licensing implementation in line with the actual licensing policy.

5 Version 4.0

5.1 Version 4.0.0

20th September 2008

Version 4.0.0 is an internal release in preparation for a public release with support for 64 character entity names.

xops3/ops3 2.6.0:
histraw 1.8:

imgstat 2.3:
imgstat64: 2.2
localimg64: 2.2
16th September 2008

- Initial support for up to 64 character entity and attribute names. Note that the imgstat executable would successfully open with 64 character entity/attribute databases but not provide 64 character runtime functionality. The existing imgstat and localimg executables only support 8 character entity name functionality. To use the up to 64 character entity name support, the localimg64 and imgstat64 executables should be used to define and interact with the shared memory.

5.2 Version 4.0.1

16th January 2009

dbfhist 2.5:
12th October 2008
Initial support for up to 64 character entity and attribute names. Fixed an record buffer problem introduced in the original 4.0.0 internal development.

xops3/ops3 2.6.1:
19th November 2008
Resolved a complex and variable line display overlap problem that occurred when a DGT graphic was first loaded.

xops3/ops3 2.6.2:
20th November 2008
Modified the alarm bell sounding logic to not sound immediately if it's already partway through a bell sounding cycle.

5.3 Version 4.0.2

18th April 2009

xops3/ops3 2.6.3:
mshell 1.27:
9th February 2009

- Fixed an introduced problem where the use of the attribute definitions of the form ATTR(ENT) in historical view statements were not supported.

mvmsgs 3.18:
12th March 2009

- Changed the default settings so that there is no maximum alarm message count and the default histmsgs maximum count is 10000. Previously both these maximum values were 1000 record entries.

xops3/ops3 2.6.4:
mshell 1.28:
mvmsgs 3.19:
9th April 2009

- Resolved an issue where adding an hour to a timestamp across a daylight savings changeover would result in the same timestamp value. This caused the loop through functionality for a historical view at the daylight savings changeover behave incorrectly for some operating systems.

5.4 Version 4.0.3

20th April 2009

alarms 2.21:
exproc 1.15:
mstracter: 2.6:
mvdded 2.10:
re_histr 1.11:
setval: 1.14:
localimg 1.11:
mv_histr 4.3.0:

- Incremented the version number to highlight the additional support of up to 64 character entity and attribute functionality due to a recompile in this MacroView Version. Note that this supported had existed for these executables in the V4.0.x builds but the revision number had not been updated until MacroView Version 4.0.3.

6 Notes

6.1 Windows Services

The MacroView Windows Service implementation performs very simple MACRODIR environment variable replacement for the following service fields:

- Executable
- Executable Parameters
- Working Directory
- Log File

The MACRODIR environment variable replacement searches for $MACRODIR entries and replaces the text with the MACRODIR environment variable value. It doesn't support other environment variable names or specification formats.

6.2 STATUS2R Attribute Format

The STATUS2R attribute format is a new PLC model attribute format introduced in Version 3.11.3 of MacroView. It is a minor variation of the STATUS2 attribute format. The STATUS2 format interprets two consecutive data bits in PLC memory as identifying a four state attribute value e.g.

0 0 MANUAL
0 1 AUTO
1 0 CASCADE
1 1 DISABLED

The STATUS2R attribute format is similar in operation except that it's data is obtained from the same bit position in two consecutive registers. A little-endian approach is taken to ordering the bits i.e. the low order bit is read from the register address and the high order bit is taken from the following register. This feature has been added to ease the configuration burden in moving between PLC types which differ in how digital values are modeled in memory. For example, the Modbus protocol can be used to retrieve digital data where one digital value occupies a single register address. Other PLC drivers may not have that functionality and only allow digital data to be presented in the form of packed integer registers.

The STATUS2R attribute format implementation does not support attribute value sets. The following error message will be generated if an attempt is made to set a STATUS2R formatted value:

The STATUS2R attribute format does not support set requests.

Set request functionality is not supported because of the difficult in implementing write requests consistently across communication drivers - in particular existing drivers. Performing two protocol actions for a single STATUS2R set request would be problematic in failure scenarios.

6.3 Metascript MessageQueue Support

As of MacroView Version 3.11.6 metascripts in mshell and xops3/ops3 support the ability to send messages directly into the MacroView message queue. Previously this could only be done by running an external program such as the sendmsg command line utility. The ability to send messages from metascript code reduces the processor overhead involved in the action and also stops "DOS box" windows from appearing in the Windows version of MacroView. The syntax uses the existing metascript SEND TO command and is limited to basic message properties such as the priority, color and message text.

```
SEND "Priority(color, message)" TO "MessageQueue";
```

Where *Priority* is one of the following options:

- SystemAlarm
- Emergency
- Alert
- Warning
- LogOnly

The *color* is a MacroView color index and the *message* portion is any sequence of ASCII text. Color variables can be embedded by use of the embedded variable syntax for strings {}.
e.g. `SEND "LogOnly({Green}, Testing a log only message)" TO "MessageQueue";`

Index

- $ -

$MACRODIR 15

- A -

alarms 2

- D -

debug 2

- E -

environment variable 15

- M -

MACRODIR 15
mvmsgs 2

- P -

ping 2
pingdrv 2

- S -

scan rate 2
service 2
service manager 3