

**mysticview**

**COLLABORATORS**

	<i>TITLE :</i> mysticview		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		April 14, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>mysticview</b>	<b>1</b>
1.1	mysticview.doc . . . . .	1
1.2	mysticview.library/MV_CreateA . . . . .	1
1.3	mysticview.library/MV_DrawOff . . . . .	3
1.4	mysticview.library/MV_DrawOn . . . . .	3
1.5	mysticview.library/MV_Delete . . . . .	4
1.6	mysticview.library/MV_GetAttrsA . . . . .	5
1.7	mysticview.library/MV_Refresh . . . . .	6
1.8	mysticview.library/MV_SetAttrsA . . . . .	6
1.9	mysticview.library/MV_SetViewRelative . . . . .	10
1.10	mysticview.library/MV_SetViewStart . . . . .	11

---

# Chapter 1

## mysticview

### 1.1 mysticview.doc

```
MV_CreateA()  
MV_DrawOff()  
MV_DrawOn()  
MV_Delete()  
MV_GetAttrsA()  
MV_Refresh()  
MV_SetAttrsA()  
MV_SetViewRelative()  
MV_SetViewStart()
```

### 1.2 mysticview.library/MV\_CreateA

```
NAME  
MV_CreateA - create a mysticview instance.  
MV_Create - varargs stub for MV_CreateA.
```

#### SYNOPSIS

```
mysticview = MV_CreateA(screen,rastport,taglist)  
d0          a0      a1      a2
```

```
APTR MV_CreateA(struct Screen *,struct RastPort *,struct TagItem *)
```

```
APTR MV_Create(struct Screen *,struct RastPort *,...,TAG_DONE)
```

#### FUNCTION

This function creates a viewer instance for a given screen and

---

rastport.

#### INPUTS

screen - pointer to the Screen for the viewer to appear on.  
 rastport - pointer to the RastPort for the viewer to appear on.  
 tags - pointer to an array of TagItems. most of these attributes are both available to this function and to  
 MV\_SetAttrsA()  
 . there are only few attributes that can only be passed to MV\_CreateA() for initialization:

#### TAGS

MVIEW\_Priority (ULONG) - task priority for the viewer's background task. valid range: -128...127. do not set this priority unless you have a good reason to and know exactly what you are doing, and better do not set this priority above your own task's priority.  
 default: the caller's task priority.

MVIEW\_RPSemaphore (struct SignalSemaphore \*) - pointer to an initialized SignalSemaphore, being obtained for each access to the view's underlying RastPort. locking the RastPort is mandatory if you want to share it between tasks, so you MUST supply this tag if you want to draw ANYTHING to the same RastPort as the view's background task. (v4) example:

```
struct SignalSemaphore rastlock;
InitSemaphore(&rastlock);

mview = MV_Create(screen, rastport,
  MVIEW_RPSemaphore, &rastlock, ...);

MV_DrawOn(mview);

...

/* your drawing here: */

ObtainSemaphore(&rastlock);

SetAPen(rastport, pen);
Move(rastport, ...);
Draw(rastport, ...);

ReleaseSemaphore(&rastlock);

...

MV_Delete(mview);
```

note: the view's background task tries to hold locks on the semaphore as short as possible, and it frees the lock as often as possible, even for very short time spans. try to be as fair as the view.

## RESULTS

mysticview - the mysticview instance created, or NULL if something went wrong.

## NOTES

only the tags that are exclusively available to MV\_CreateA() are listed here. refer to the documentation of MV\_SetAttrsA() for all other tags.

## SEE ALSO

MV\_Delete()  
,  
MV\_DrawOn()

### 1.3 mysticview.library/MV\_DrawOff

## NAME

MV\_DrawOff - disable asynchronous drawing.

## SYNOPSIS

```
MV_DrawOff(mysticview)
    a0
```

```
void MV_DrawOff(APTR)
```

## FUNCTION

This function turns off asynchronous drawing for the given mysticview instance.

## INPUTS

mysticview - pointer to a mysticview instance.

## RESULTS

none

## SEE ALSO

MV\_DrawOn()

### 1.4 mysticview.library/MV\_DrawOn

## NAME

MV\_DrawOn - enable asynchronous drawing.

## SYNOPSIS

```
success = MV_DrawOn(mysticview)
    d0          a0
```

```
BOOL MV_DrawOn(APTR)
```

#### FUNCTION

This function will establish a background task for the given mysticview instance. it will perform asynchronous drawing and instantly react to changes of attributes.

#### INPUTS

mysticview - pointer to a mysticview instance.

#### RESULTS

success - boolean to indicate whether background drawing could be established.

#### SEE ALSO

```
MV_DrawOff()  
,  
MV_SetAttrsA()
```

## 1.5 mysticview.library/MV\_Delete

#### NAME

MV\_Delete - destroy a mysticview instance.

#### SYNOPSIS

```
MV_Delete(mysticview)  
a0
```

```
void MV_Delete(APTR)
```

#### FUNCTION

This function will delete a mysticview instance, shut down related tasks and free all associated memory.

#### INPUTS

mysticview - pointer to a mysticview instance.

#### RESULTS

none

#### NOTES

You must delete a mysticview instance before you may close the underlying Screen or Window.

#### SEE ALSO

```
MV_CreateA()
```

---

## 1.6 mysticview.library/MV\_GetAttrsA

### NAME

MV\_GetAttrsA - query attributes.  
MV\_GetAttrs - varargs stub for MV\_GetAttrsA.

### SYNOPSIS

```
MV_GetAttrsA(mysticview, taglist)
              a0          a1

void MV_GetAttrsA(APTR, struct TagItem *)

void MV_GetAttrsA(APTR, ..., TAG_DONE)
```

### FUNCTION

Retrieve a list of attributes from a mysticview instance.

### INPUTS

mysticview - pointer to a mysticview instance.  
tags - pointer to an array of TagItems.

### TAGS

refer to the attributes' descriptions in  
MV\_SetAttrsA()

additional tags that may be queried only:

MVIEW\_PictureHeight (ULONG \*) - current visible  
height of the picture inside the RastPort.

MVIEW\_PictureWidth (ULONG \*) - current visible width  
of the picture inside the RastPort.

MVIEW\_PictureX (ULONG \*) - current left edge of the  
picture inside the RastPort.

MVIEW\_PictureY (ULONG \*) - current top edge of  
the picture inside the RastPort.

### RESULTS

none

### NOTES

the tags that are exclusively available to  
MV\_CreateA()  
cannot  
be queried with this function.

### SEE ALSO

```
MV_CreateA()
,
MV_SetAttrsA()
```



## 1.7 mysticview.library/MV\_Refresh

### NAME

MV\_Refresh - refresh the display.

### SYNOPSIS

```
MV_Refresh(mysticview)
    a0
```

```
void MV_Refresh(APTR)
```

### FUNCTION

This function will refresh the mysticview's current display, reflecting all changes that applied meanwhile, and redraw the picture. this function might be of use when you perform changes inside the picture's raw data, or when your RastPort's window is of the type WFLG\_SIMPLE\_REFRESH and you need to signal the viewer to redraw the picture, or when you do not use asynchronous drawing.

### INPUTS

mysticview - pointer to a mysticview instance.

### RESULTS

none

### SEE ALSO

```
MV_DrawOn()
,
MV_SetAttrsA()
```

## 1.8 mysticview.library/MV\_SetAttrsA

### NAME

MV\_SetAttrsA - set attributes.

MV\_SetAttrs - varargs stub for MV\_SetAttrsA.

### SYNOPSIS

```
MV_SetAttrsA(mysticview,taglist)
    a0          a1
```

```
void MV_SetAttrsA(APTR,struct TagItem *)
```

```
void MV_SetAttrsA(APTR,...,TAG_DONE)
```

### FUNCTION

submit a list of attributes to a mysticview instance. any changes will be applied instantly when background drawing has been activated via

```
MV_DrawOn()
```

.

## INPUTS

mysticview - pointer to a mysticview instance.  
tags - pointer to an array of TagItems.

## TAGS

MVIEW\_BackColor (ULONG) - RGB background color for areas not covered by the image. default: a pale, dark green

MVIEW\_DestHeight (ULONG) - destination height inside the RastPort. default: undefined.

MVIEW\_DestWidth (ULONG) - destination width inside the RastPort. default: undefined.

MVIEW\_DestX (ULONG) - destination left edge inside the RastPort. default: undefined.

MVIEW\_DestY (ULONG) - destination top edge inside the RastPort. default: undefined.

MVIEW\_DisplayMode (ULONG) - image scaling and aspect mode. currently defined:

MVDISPMODE\_FIT  
the image fits exactly into the visible area and may get distorted.

MVDISPMODE\_KEEPAPECT\_MIN  
respect the aspect ratios of both the screen and the picture. the image is fully visible at zoom factor 1.

MVDISPMODE\_KEEPAPECT\_MAX  
respect the aspect ratios of both the screen and the picture. either the image's width or height is fully visible at zoom factor 1.

MVDISPMODE\_ONEPIXEL  
ignore the image aspect. the screen's aspect will be respected, though.

MVDISPMODE\_IGNOREASPECT  
ignore both the screen's and image's aspects.  
default: MVDISPMODE\_KEEPAPECT\_MIN

MVIEW\_Dither (ULONG) - dither activation mode on displays with 256 colors (or less), or on HAM displays.

MVDITHERMODE\_ON - always dither

MVDITHERMODE\_OFF - never dither

MVDITHERMODE\_AUTO - dither only when necessary. refer to the 'prefs' textfile in the guigfx.library distribution, guigfx.library/DrawPictureA() and render.library/RGBArrayDiversityA() for details.

---

default: MVDITHERMODE\_AUTO

MVIEW\_DitherMode (ULONG) - error diffusion mode. refer to render/render.h for the available modes.  
default: DITHERMODE\_EDD.

MVIEW\_HSType (ULONG) - histogram type, according to the definitions in render/render.h. do not touch unless you know exactly what you are doing. only \_TURBO types are allowed. default: HSTYPE\_12BIT\_TURBO

MVIEW\_Picture (APTR) - pointer to a guigfx.library picture to be displayed. the image is NOT incorporated to the mysticview instance, it is only referenced. the image MUST NOT be deleted when it is the current picture of a mysticview instance. delete the viewer first, or set MVIEW\_PICTURE to NULL before. default: NULL

MVIEW\_Precision (ULONG) - color allocation precision. refer to graphics.library/ObtainBestPenA().  
default: PRECISION\_ICON

MVIEW\_PreviewMode (ULONG) - realtime refresh mode.

MVPREVMODE\_NONE  
no realtime refresh takes place.

MVPREVMODE\_GRID  
when the image scrolls, scales or rotates, or when the visible dimensions change, the image is first drawn as a grid, and then rendered with highest quality.

MVPREVMODE\_OPAQUE  
when the image scrolls, scales or rotates, or when the visible dimensions change, the entire image is first drawn as a quick and dirty preview, and then rendered with highest quality.

default: MVPREVMODE\_NONE

MVIEW\_ReadySignal (ULONG) - signal bit that will be submitted to your task when the current picture has been completely rendered in highest quality. default: -1 (no signal will be submitted)

MVIEW\_Rotation (ULONG)  
rotation of the image as a fixed floatpoint integer. the upper 16 bit determine the integral part of the number, the lower 16 bit determine the fraction. valid range is 0 (0\textdegree) to 1 (360\textdegree). defaults to 0.

MVIEW\_ScreenAspectX (ULONG)

MVIEW\_ScreenAspectY (ULONG)

enforce a screen aspect.

---

default: the screen's aspect.

MVIEW\_ShowCursor (BOOL)

display an image cursor. default: FALSE

MVIEW\_StaticPalette (BOOL)

use the same static palette for any picture instead of a dynamic one. this will result in a faster display at lower quality. skipping from one picture to another causes less flickering with a static palette.  
default: FALSE (dynamic palettes are used)

MVIEW\_Text (char \*) - a single line of text to be displayed

at the bottom of the picture's area inside the RastPort. the text will be copied to the mysticview instance and may be freed prior to the mysticview instance.  
default: NULL

MVIEW\_XPos (ULONG)

MVIEW\_YPos (ULONG)

horizontal/vertical position inside the image. this is a fixed-floatpoint integer. the upper 16 bit determine the integral part of the number, the lower 16 bit determine the fraction. valid range is from 0 to 1.  
default is 0.5 (0x00008000)

MVIEW\_Zoom (ULONG) - zoom factor. this is a fixed-floatpoint

integer. the upper 16 bit determine the integral part of the number, the lower 16 bit determine the fraction. valid range is from 0.1 to 10. default is 1 (0x00010000)

MVIEW\_DrawArrows (ULONG) - boolean to indicate that small

arrows are to be drawn to the view's borders, when a picture is not fully visible. default: FALSE (v4)

MVIEW\_ShowPip (ULONG) - boolean to indicate that a PiP

view of the current picture is to be displayed. this feature is considered experimental (v4). better do not use it. default: FALSE (v4)

MVIEW\_TextColor (ULONG) - 0x00rrggbb value for displaying

text (see MVIEW\_Text attribute). default: white.  
(v4)

MVIEW\_MarkColor (ULONG) - 0x00rrggbb value for displaying

highlighted display elements, such as arrows, grid lines and the PiP frame. default: bright green.  
(v4)

RESULTS

none

NOTES

all these attributes are also available to  
MV\_CreateA()

.

SEE ALSO

```
MV_CreateA()  
,  
MV_GetAttrsA()  
,  
MV_DrawOn()  
,  
MV_Refresh()
```

## 1.9 mysticview.library/MV\_SetViewRelative

NAME

MV\_SetViewRelative - perform relative movement of the picture

SYNOPSIS

```
MV_SetViewRelative(mysticview, newx, newy)  
                   a0           d0   d1
```

```
void MV_SetViewRelative(APTR, LONG, LONG)
```

FUNCTION

This function performs a relative movement of the mysticview's current picture.

in this example, we assume that you want to allow the user to drag the picture around with the mouse. proceed as follows:

- on a MOUSEBUTTON / SELECTDOWN event, call  
    MV\_SetViewStart()  
        with the mouse coordinates related to this event. call  
    ModifyIDCMP() to catch MOUSEMOVE events now.
- while dragging (thus, by receiving MOUSEMOVE events), pass  
    the mouse coordinates to the mysticview instance via  
    MV\_SetViewRelative(). this will position the visible part  
    of the picture. attributes like zoom, displaymode, rotation,  
    aspects, etc. are fully considered.
- when receiving a MOUSEBUTTON / SELECTUP event, call  
    ModifyIDCMP() and disable MOUSEMOVE events. (There is no  
    need for your application to be burdened with a continous  
    flood of mouse movement events.)

INPUTS

```
mysticview - pointer to a mysticview instance.  
newx       - new x coordinate  
newy       - new y coordinate
```

RESULTS

none

SEE ALSO

---

MV\_SetViewStart()

## 1.10 mysticview.library/MV\_SetViewStart

### NAME

MV\_SetViewStart - set starting point for picture movement

### SYNOPSIS

```
MV_SetViewStart(mysticview, startxpos, startypos)
                a0          d0          d1
```

```
void MV_SetViewStart (APTR, LONG, LONG)
```

### FUNCTION

This function sets that starting coordinate for a relative movement of the picture inside the RastPort. Refer to

```
MV_SetViewRelative()
for further details.
```

### INPUTS

```
mysticview - pointer to a mysticview instance.
startxpos  - initial x coordinate
startypos  - initial y coordinate
```

### RESULTS

none

### SEE ALSO

```
MV_SetViewRelative()
```

---