

EDM Exercises

These exercises may go into more detail than you wish to go if you are not likely to have to use EDM in your job, so don't worry if you don't get to the end or you decide to skip some of the tasks. In particular there is probably no point in replicating all of the complex graphic in exercise 2, just pick one or two symbols to work on.

In most cases, edm loads and saves files to/from the current directory. However for color definitions and the default schema will it go to the \$EDMFILES directory.

A subdirectory called edm containing the file edm.db should already exist in your home directory; use this directory to save your edm files for these exercises.

Change into that directory and start edm as follows:

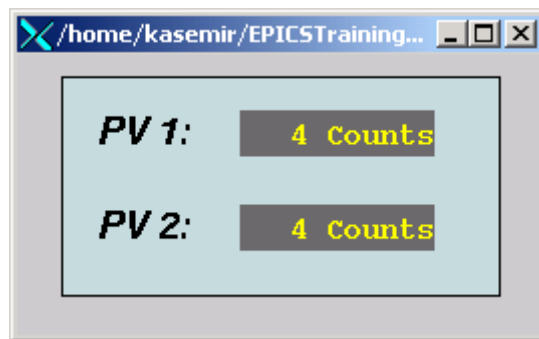
```
cd edm
edm -m user=$USER &
```

You will also need to start an IOC to serve the Process Variables (PVs) that your displays will be connecting to. Open a separate terminal window for the IOC, and in that window run these commands:

```
cd edm
softIoc -m user=$USER -d edm.db -s
```

Exercise 1: Editing

We will create a display similar to this:



From the edm main window, create a new display using File/New.

Save the empty display as "example1" (from now on, save every once in a while just in case ...)

Create:

Two Labels "PV 1" and "PV 2":

Left-button-rubberband the approximate area. Select Graphics/Static Text. Enter "PV 1" as the text. Select a font such as helvetica 18. Press OK.

Repeat for "PV 2".

Two Monitor/Textupdate objects. For PV names use “\$(user):aiExample” and “\$(user):calcExample”.

One Graphics/Rectangle, make it “filled”.

Note: To change the stacking order, select objects, then use the middle-button menu to raise or lower them.

Use Select, move, resize, align, ...until the display looks a bit like the example shown above (which is in execute mode).

These may help you to finish quicker:

Display Properties: snap-to-grid, ortho move & line draw

Copy/paste

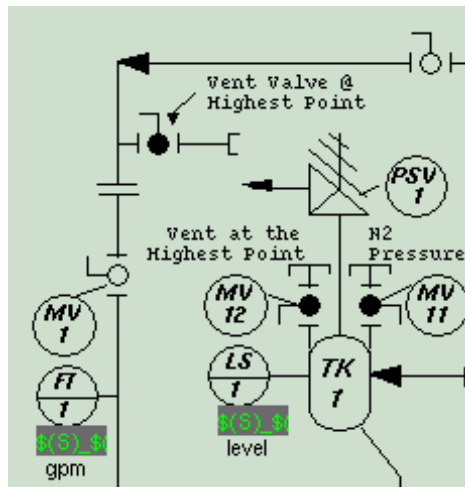
Switch to “execute” mode by de-selecting all objects, click the middle mouse button, and choose *execute* from the menu

Exercise 2: More Editing

Create a new display, save it as example2.

Unless you already remember everything, launch Help/Line Objects.

Use Graphics objects (circles, text and mostly lines) to create some of the elements you see in this screen-shot:



Hint: Arrowheads are “filled” lines

Select several objects at once, change color or font or ... via the “Edit/Display Properties” option.

Exercise 3: PVs

We will see how different PVs can be represented in different ways.

Execute edm as before and open example1. Save the display file as “example3”.

In your display there is a text update for the record “\$(user):calcExample”.

Add a Monitor/Meter that uses the same PV – execute.

Add a text entry control to the .CALC field of the record:

Create a Control/Textentry, set “\$(user):calcExample.CALC” as the PV name.

Add a Control/Menu Button to the .SCAN field of the same record (e.g. PV name “\$(user):calcExample.SCAN”) - execute.

Add a text update that displays a calculated PV, e.g. “CALC\sum(\$(user):calcExample, 2)” - execute.

Exercise 4: Colors, Macros

We will see how macros get passed to related displays.

Open example3 and save it as “example4”. This will be our main display that launches off two related displays.

Add a label (Graphic/Static text) that shows “User \$(user)”.

Add two Control/Related Display buttons:

Set “File” to “relatedDsp”, macros to “param=1”, label to “Rel. 1”

Button #2: File: “relatedDsp”, macros: “param=2”, Label: “Rel. 2”

Add a text entry control, and set its Control PV to “\$(user):aoExample”. This PV name will also be used in a color rule inside the related display.

Add an exit button. Check the “Exit Program” option.

Create a new display to be used as the related display, save it as “relatedDsp”:

Create a static text object with Text Value set to “Related Display, param=\$(param)”

Create a rectangle, choose a dynamic color for line color, use the PV name “\$(user):aoExample”

Create an exit button. Do *not* check the “Exit Program” option.

Save “relatedDsp” and close it.

Execute “example4”

Click the Related Display button, the associated display should appear and the static text object should display the symbol value

Change the value of the PV from the text control, the rectangle color should be

determined by the color rule

Click the Exit Button on each related display.

Click the Exit Button on “example4”, the main screen.

Does this exit edm?

Exercise 5: Symbols

Create a new display, save it as “switch”.

Follow the “Creating symbol” help item to create the states of a simple switch:

broken, open, closed

Details:

Create an invisible rectangle

Draw invariant symbol components

Copy image and paste two copies to the display

Draw state dependent components

Arrange images in a rows/columns ordering, first state is upper-left, last is lower-right

If any objects have been grouped, un-group them now

Click middle button, choose Auto make symbol

Save symbol file as switch & close the display.

Create new display “example5”

Add text entry to control e.g. “\$(user):aoExample”

Add a symbol instance (Monitor/Symbol)

Use symbol file recently created, use same PV as referenced in text entry object. Select 3 items, configure each as follows:

Item 1: $0 \leq \text{PV value} < 1$

Item 2: $1 \leq \text{PV value} < 2$

Item 3: $2 \leq \text{PV value} < 3$

Execute example5

Exercise 6: Command Button

Open any of the examples.

Add a shell command button to start StripTool.

Execute.

Note how you can drag & drop (middle button) PV names from an edm object to StripTool's channel name field.

Exercise 7: Display Schemes

Create a new display using File/New

From the middle mouse button menu, select "Display Properties".

In here, you can change the background color and select default fonts and colors for newly created display elements. Select a background color, click OK.

You would save the "Display Scheme" in case you want to create a standard setup (background color etc.) for part or all of your displays. In fact EDM automatically loads the file names "\$EDMFILES/default.scheme" on startup, so by saving a scheme under that name, you determine the start-up defaults.

One quirk/feature is that EDM wants to save & load all display schemes in a global EDM configuration directory (\$EDMFILES). That makes sense when you want to share display schemes with others, especially for the "default.scheme". For this training however, keep all your files in your home/edm directory.

Invoke the middle-button menu again, select Save Display Scheme, navigate to your home/edm directory, and create "training.scheme" (yes, the Motif-type file dialog is strange).

From the main window, choose File/Exit. There will be a warning because you didn't save the display. That's OK: We don't care about the display, we only wanted to create a "Scheme"!

Restart edm, load one of your example displays, select "Load Display Scheme..." and load your "training.scheme" file. What changes when you do that?