STUDY OF CONTROL SYSTEM STUDIO (CSS) AND DEVELOPMENT OF CONTROL PANEL FOR PF-AR VACUUM DISPLAY

Prachi Chitnis

RRCAT, Indore, INDIA

23 Feb, 2010

INDEX

SYNOPSIS

The research activity involved the study of the Control System Studio(CSS), which is the Integrated Development Environment (IDE)for development of full-fledged control system applications. It is an Eclipse Rich Client Platform based IDE, and is used to design and develop control systems based on EPICS, TANGO, TINE etc.

The work involved the study of various tools available in CSS, like Synoptic Display Studio (SDS), ADL Converter, Probe, PV Table etc.

The work also involved the development of control panels for PF-AR Vacuum. Two panels are created for the display. The first one is an exact replica of the previous panel that was designed in DM2K. The second one is the modified version with gradient color coding. The color gradient is coded in JavaScript.

Finally, a two day training-presentation for CSS was given to the KEKB controls group.

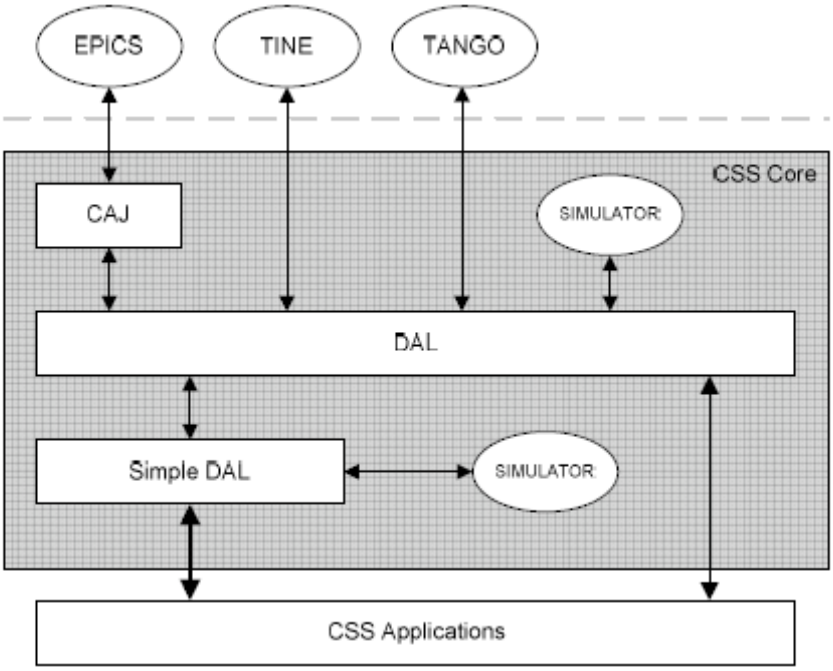
CONTROL SYSTEM STUDIO

The Control System Studio (CSS) is an Eclipse RCP based development platform and the fundament for many applications like EPICS, TANGO etc. As most of these applications deal with process variables and connections to control systems, the CSS Core provides the necessary APIs for a convenient start.

The integrated development environment of CSS provides facility for database development, alarm management system, display development and conversion, data trending, diagnostic tools etc.

The Data Access Layer (DAL) is the core of the connection APIs. It communicates to EPICS through CAJ (Channel Access Java), which is a pure Java implementation of the CA protocol. In the (near) future it will be possible to connect to TINE and TANGO control systems through DAL as well. A TINE integration is already available as Beta. DAL is an inherent part of the CSS Core but can also be used as a library in any other Java application. SimpleDAL is a connection layer built on top of DAL. It provides a slim, less complex API that allows for a much easier start for developers dealing with process variables in their applications. Using SimpleDAL implies a certain syntax for process variable addresses that enables applications to make use of the following features:

- Access different control systems (e.g. TINE and EPICS) in one application
- use characteristics, a concept for resource saving access to record fields
- query process variables in different types
- use simulated channels
- address system functions as process variable



There are various utilities available with the CSS such as:

- Display applications (e. g. The Synoptic Display Studio).
- Alarming tools and applications.
- Trend tools and applications.
- Diagnostic Tools
- Diagnostic tools and applications.
- Configuration tools and applications.
- Management tools and applications.
- Editors
- Utilities

Process variable address syntax

The general syntax of a process variable address is defined as follows. This uses the metasyntax based on the Extended Backus-Naur Form

[1] address ::= [protocol] id [type]

[2] protocol ::= ('dal-epics' | 'dal-tine' | 'dal-tango' | 'local') '://'

[3] id::= (letter | specialcharacter) +

[4] type ::= ', ' ('double' | 'int' | 'long' | 'string' | 'enum')

[5] letter::= 'A' | ... | 'Z' | 'a' | ... | 'z'

[6] specialcharacter::= ':' | '/' | '\' | '.' | '[' | ']'

[7] number ::= digitWithoutZero (digit)*

[8] digit ::= '0' | ... | '9'

[9] digitWithoutZero ::= '1' | ... | '9'

- There are 3 optional and 1 mandatory fragments that constitute a full process variable address (line 1).
- The protocol (line 2) is optional and defines the connection protocol. If a prefix is not specified, a default protocol is chosen according to the settings of the CSS-Core/Control-System preference page.
- The id (line 3) is mandatory. It must be a globally unique name identifying the information you want to address.
- The type (line 5) is optional, too. It can be used to specify the expected return type for channel values explicitly.

Syntax for EPICS

When EPICS channels are addressed, line 3 is as follows:

[3a] id::= recordname ['.' fieldname] [characteristic]

[10] recordname ::= (letter | specialcharacter)+

[11] fieldname ::= (letter)+

[12] characteristic ::= '[' (letter)+ ']'

- An EPICS process variable is always identified by its recordname (line 10) which
- is therefore mandatory.
- Optionally a fieldname can be provided to address a single field of a record (line 11). If no fieldname is provided the address defaults to the .VAL field.
- The characteristic (line 12) is optional as well. If defined it allows for accessing additional information of a record without establishing a new connection. All characteristics of the same record share the same connection. So in general it is a good idea to use characteristics whenever possible to save system resources. These can be
    - [Position] - position
    - [Description] – long description
    - [displayName] – short description
    - [propertyType] - type
    - [resolution] – number of bits used for ADC conversion of analog value when sampled
    - [minimum] – minimum allowed value
    - [maximum] – maximum allowed value
    - [graphMin] – minimum allowed value when displayed (e.g. in a chart)
    - [graphMax] – maximum allowed value when displayed (e.g. in a chart)
    - [format] – C print-f style format that is used to render the value
    - [units] – units of the value
    - [scaleType] – scale type for plotting (linear or logarithmic)
    - [warningMax] – upper warning limit
    - [warningMin] – lower warning limit
    - [alarmMax] – upper alarm limit
    - [alarmMin] – lower alarm limit
    - [sequenceLength] – sequence length
    - [enumValues] – enum value array (returns Object[])

- [enumDescriptions] – enum value descriptions (returns String[])
- [bitDescriptions] – bit descriptions (returns String[])
- [conditionWhenSet] – active bit significance
- [conditionWhenCleared] - inactive bit significance
- [bitMask] – bits relevance

There are various versions of CSS. The major ones include the DESY and SNS versions. CSS comes with a subset of the all associated tools depending upon the version. The following table shows a comparison between various versions.

	Basic EPICS	**SNS Office**	**SNS CCR**	**DESY SNS**
Download links	MS Windows Mac OS X x86 RH Linux x86	MS Windows Mac OS X x86 RH Linux x86	Full SNS Sources	DESY CSS site
Operator Panel Editor Prototype	✓	✓	✓	✓
Data Browser, 'Stripchart' live and archived data	✓	✓	✓	✓
Post Analyzer Basic fitting and correlation of Data Browser data	✓	✓	✓	✓
Probe Inspect live PVs	✓	✓	✓	✓
EPICS PV Tree Display a PV's input link hierarchy	✓	✓	✓	✓
PACE Table editor for 'critical' PVs	✓	✓	✓	-
RDB Table Generic editor for RDB data	✓	✓	✓	-
System Monitor Memory usage info	✓	✓	✓	✓
Therapist	✓	✓	✓	✓
Clock	✓	✓	✓	✓
Channel Access Client EPICS network library	✓	✓	✓	✓
Channel Archive Client Access to 'old' Channel Archiver	✓	✓	✓	✓
RDB Archive Client Access to BEAUTY	✓	✓	✓	✓
Other Archive Clients Access to AAPI, Data	-	-	-	

Logger, ...				
PV Utility Map Devices and IOCs to PVs	-	✓	✓	-
Rack Utility Locate devices in racks	-	✓	✓	-
PV Fields View EPICS record info for PVs	-	✓	✓	-
SNS ELog Support Interfaces various tools to SNS ELog	-	✓	✓	-
Message Log Viewer Viewer for the CSS Message log (in RDB, from JMS)	-	✓	✓	-
BEAST GUI User Interface of the Best Ever Alarm System Toolkit	-	-	✓	-
LDAP Authentication & Authorization Login support for alarm GUI and SDS	-	-	✓	-
Kerberos + LDAP Authentication & Authorization Login support ...	-	-	-	✓
Synoptic Display Operator Panel Editor	-	-	-	✓
Non-EPICS DAL Clients Network library for TINE, ...	-	-	-	✓
Screenshot Take screen shots, send to DESY ELog	-	-	-	✓
SNL Editor EPICS Sequencer IDE	-	-	-	✓
I/O Configurator Tool DESY Profibus Config Tool	-	-	-	✓
DCT DESY Database Creation Tool	-	-	-	✓
Namespace Tool DESY LDAP Access	-	-	-	✓
AMS GUI GUI for DESY alarm management system	-	-	-	✓

Generic 'Welcome' Generic CSS Introduction, Channel Access setup info	✓	-	-	-
SNS-specific 'Welcome' CSS Introduction with SNS Example PVs	-	✓	✓	-
DESY-specific 'Welcome' How to download optional plug-ins	-	-	-	✓
No Specific Settings Channel Access, Archive etc. not configured	✓	-	-	-
Settings for SNS Office Network Setup for SNS CA Gateway and Archives	-	✓	-	-
Settings for SNS CCR Setup for SNS CCR IOCs etc.	-	-	✓	-
Settings for DESY Setup for DESY Kerberos, LDAP, JMS etc.	-	-	-	✓

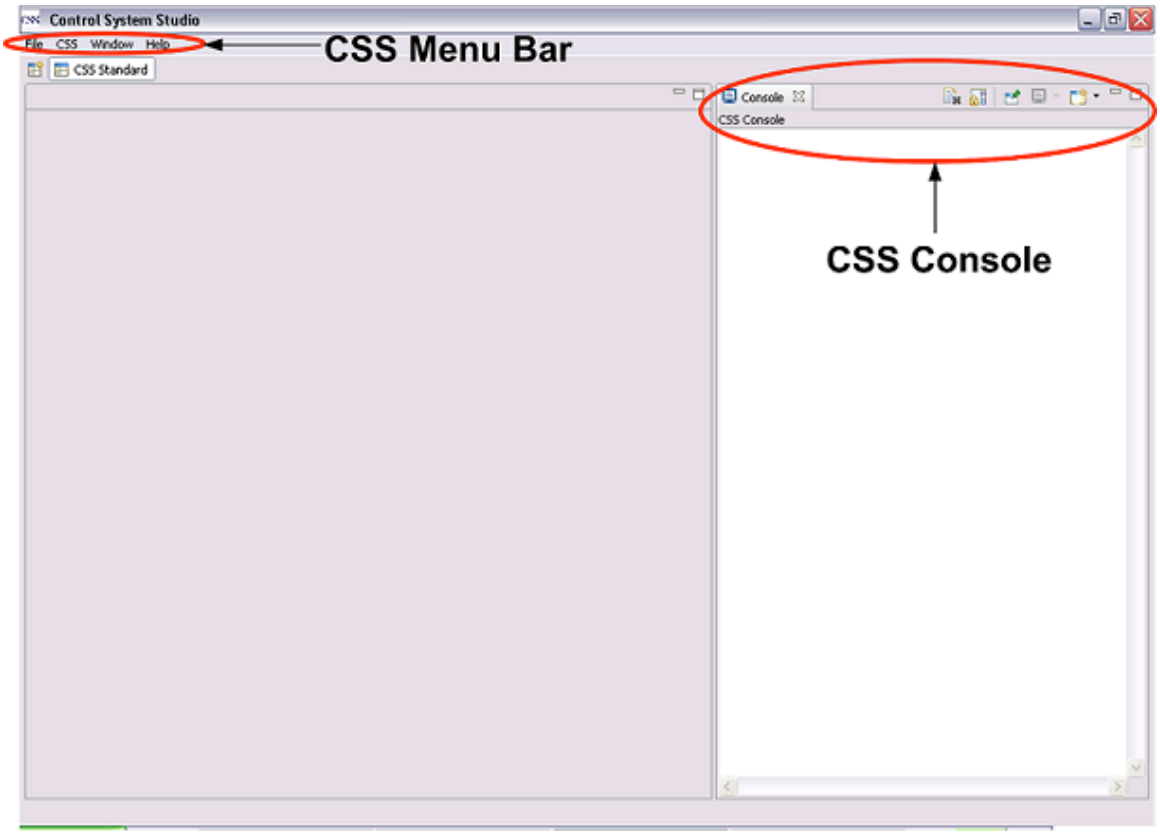
Legend: ✓ Included, ✓ Optional online update.

DOWNLOADING AND INSTALLATION

It is recommended that the standard version of CSS should be downloaded for the first use. It consists of the CSS platform and common control system applications.

1) **Pre-requisites** - Java version 1.6 or higher is required
2) **Download** - The link to this is http://css.desy.de/content/e413/index_eng.html Please choose the version according to your operating system.
3) **Extracting** – extract the downloaded zip file in a directory of your choice.
4) **Starting**
    a. Windows – Run the file css.exe under CSS <Top>.
    b. Linux – Set executable mode for the file 'css' under CSS <top>
5) **Login** - The XMPP login can be cancelled for initial use. This logins you as anonymous user into the XMPP server specified in the preferences. The XMPP server is used for remote management. One can also set up his own XMPP server.

When you start the CSS for the first time, it will look like this:



CSS Menu Bar

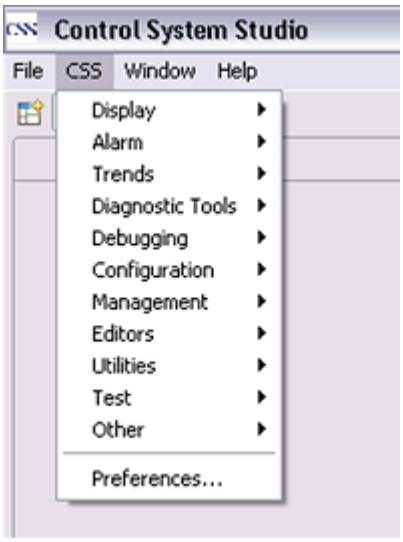
All features of the CSS are accessible through the CSS Menu Bar. It provides the following menu entries:

FILE

- New - Create a new folder, resource,
- Switch Workspace - Change the workspace of the CSS instance.
- Export preferences - Write current preferences of the CSS instance in a file.
- Open Workspace Navigator - Open the view 'Workspace'.
- Import - Import resource from CVS in the workspace.
- Exit - Shut down your CSS installation.

CSS

All available CSS applications are linked into the *CSS* menu. Depending on the CSS distribution this menu provides a subset of the following entries.



- Display - Display applications (e. g. The Synoptic Display Studio, PV table, ADL converter).
- Alarm - Alarming tools and applications
- Trends - Trend tools and applications (Data browser)
- Diagnostic Tools - Diagnostic tools and applications.(EPICS PV tree, probe)
- Configuration - Configuration tools and applications.(DCT, contacts)
- Management - Management tools and applications.
- Editors
- Utilities(clock, therapist, system monitor)
- Preferences - Open the central CSS configuration dialog that provides access to all CSS settings.

QUICKSTART - The Quickstart menu holds links to Synoptic Displays in the workspace. They can be added via context menu of a display.

WINDOW

- Open in New Window - Open a new CSS window.
- Open Perspective - Select and open a particular CSS perspective. In the CSS context, a perspective is a set of user interface elements that are positioned in a certain way.
- Show View - Open a particular CSS view. In the CSS context, a view is an element of the user interface that displays something and can be freely arranged by the user.

HELP

- Welcome
- About CSS - Open the "About" dialog that contains the CSS license agreement.
- Help Contents - Open the CSS help system. There you may find a more detailed description of the basic user interface concepts and all system settings.
- Key Assist - Open an overview of all shortcuts.
- Cheat Sheets - Shows available Cheat Sheets. Cheat Sheets are step by step documentations.
- Software Updates - Manage the local CSS installation by installing and updating features. -> find and install -> search for new features to install -> DESY CSS

CSS Console

The CSS contains its own console. This console display system messages of certain events and information about occurred errors.

Synoptic Display Studio (SDS)

SDS has its own perspective: Menu 'Window' → 'Open Perspective' → 'Display Development'. There are some example and training displays integrated that demonstrates the functionality of SDS: Menu 'CSS' → 'Install Synoptic Display Demo Displays'. The training displays are in the workspace folder 'SDS Demo Display/Training'.

For some displays a SoftIOC with special EPICS databases (Folder /SDS Demo Display/Training/EPICS_DBs_TrainingIOC) is necessary. A SoftIOC for MS windows can be found on the CSS Homepage. The databases are part of that distribution. For other OS you can copy the EPICS databases from the SDS directory mentioned above to your IOC load directory.

There are also some help pages available in CSS

LOADING CSS PLUGINS

CSS plug-ins are the various tools available with it. To add these into the CSS:

1. Go to CSS menu -> Help -> Software updates -> Find and Install
2. Select "Select for new features to install"
3. Select the check box for "Control system studio update site"
4. Click "Finish"
5. Expand the tree for "Control system studio update site"
6. Select the plug-in you want to install and click "select required" button.
7. Go "next" and accept license agreement.
8. Next and click "Finish"
9. The installed plug-in will now appear in the CSS menu under appropriate category

SETTING EPICS PREFERENCES

This setting is used to specify the EPICS IOC location and other attributes

1. Go to CSS Menu -> CSS -> Preferences -> CSS core -> Control system
2. Set the default control system as EPICS
3. Under CSS core, go to EPICS
4. Set the EPICS preferences here.

The Graphical User Interface (GUI) of the Control System Studio is based upon the following concepts:

Views
A view is a window-like UI element that typically provides a "view" on data. The Control System Studio and its applications provide various views. They are resizable and can be freely arranged within the CSS main window. Views may provide toolbars that you can use to configure the view. If you stack views on top of each other you can select the top view by clicking on its tab. Despite their name, views can be used to change the displayed data. Every change to an item of a view is saved immediately.

Editors
An editor is typically used to "edit" the data it displays. It shares many features with views. The main difference between views and editors is that an editor saves changes to its items only when you explicitly choose to save. There is only one place in the CSS main window where editors can appear. This place is called "editor area".

Perspectives
A perspective defines an arrangement of various views. Certain menu or toolbar entries can be associated to a perspective.

Synoptic Display Studio (SDS) is a graphical operator interface that represents the structure and current state of a plant. The structure is composed of basic elements so called widgets like labels, meters or bargraphs. To build up a part of a plant the widgets can be arranged in the edit mode of SDS on displays. In the execute or run mode the operators can control the processes via the displays.

Navigator

The navigator view shows the CSS projects in the workspace. A CSS project holds the configuration files for SDS displays and files of other CSS applications.

Editor / Palette

The Editor shows the displays in the edit mode and provides grid, align, ruler, etc. On the right side is the palette with all available widgets. There is a drag and drop support that connects a process variable from another CSS application automatically with a widget.

Widget Properties

For each selected widget in the editor the widget property view shows all available properties. It is possible to dynamise each property. That means that a property is connected to a process variable and changes with the value of the variable.

There are two kinds of property

1. Static- fixed during execution mode
2. Dynamic – changes during execution mode

STATIC PROPERTIES

Single left click on a property on the Property View sets a static property. Single left click on a property opens an editor. There are six general editors:

- Enter a text
- Enter a number
- Choose a color
- Choose a font
- Set a boolean
- Select a combo item

And three special editors:

- Add / change aliases
- Add / change actions
- Generate tooltip

LAYER MANAGEMENT :

To open the Layer Management Menu:

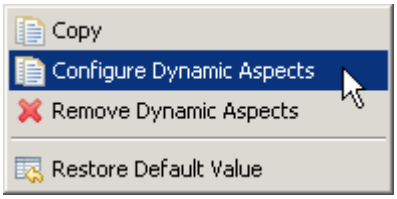
Window -> Show View -> Other… -> Synoptic Display Studio -> Layer Management

- Inside the layer management view, right click to add new layer, or move existing
- Visibility and order of a layer can be changed
- Visibility of a layer can be toggled dynamically

Right click a widget to change its layer

DYNAMISATION OF PROPERTIES

Properties that are dynamic are marked with a gear-wheel. To edit the dynamic behavior of a widget property right click on the appropriate one in the property view and select 'Configure Dynamic Aspects'.



The simplest way of dynamisation is to show the current value of a process variable in the display. The Rule is 'Direct Connection'. Channel name can be typed directly here, or an alias can be used instead of a PV name

Dynamic properties can be configured by rules (explained later)

Each widget has a Primary PV,

- It is used when you copy PV to clipboard
- Forwarded to the applications called by contribution menu
- Tip text

So it is necessary to define a primary PV

ALIAS

Alias is a macro that can be used instead of long PV name or syntax. If there is an alias set for a process variable it can be used in other properties of the widget to make typing easier. It is marked by the symbol '$' at the beginning and end of the String. Each widget can have several aliases (e.g. for displaying several plots on strip chart). Alias can be forwarded to a new display also.

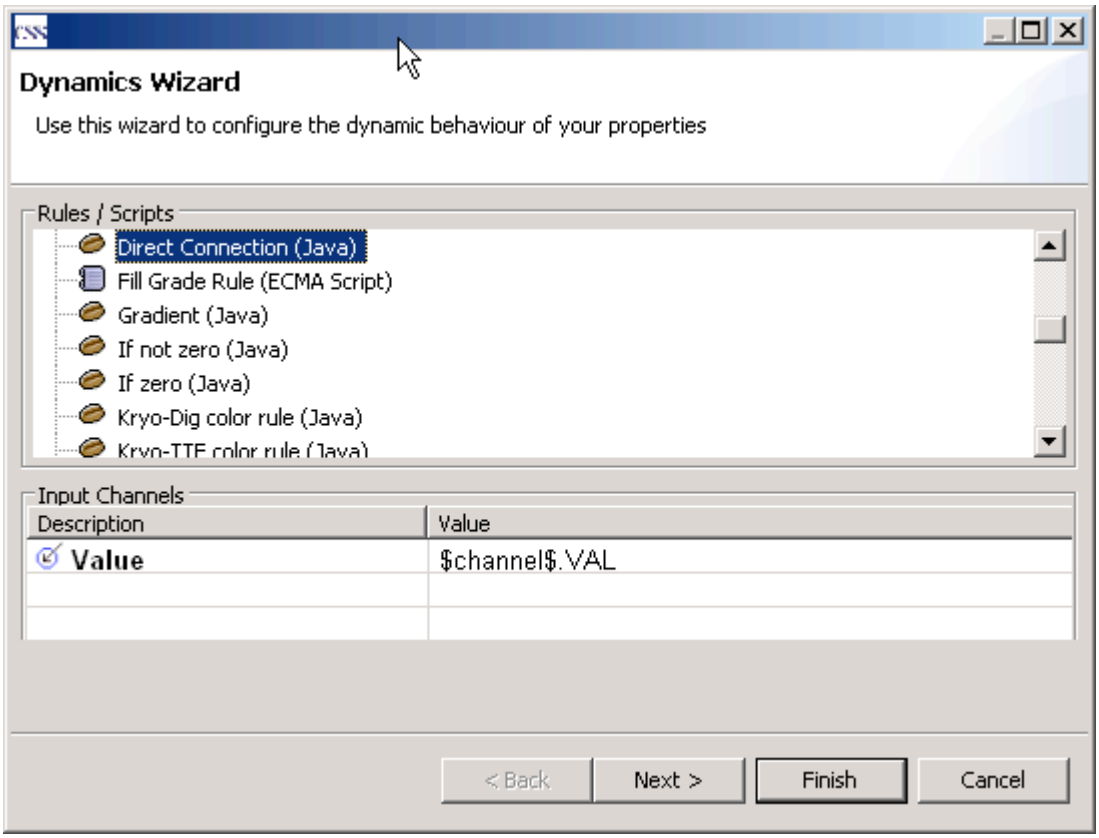
CONNECTION STATES

The background color changes according to the connection states (connection lost, connected, unknown, initial). The colors for these states can be customized. For this

1. Right click on background color properties
2. Configure dynamic aspects -> Next
3. Add or remove connection states by right click
4. Edit colors

RULES

Rules can be used to define the dynamic behavior of widget properties. There are two types of rules; Java rules and ECMA/JavaScript rules. It is recommended that JavaScript rules should be used for configuring dynamic properties instead of Java, as no compilation is needed in this case. Scripted rule files should be kept at 'SDS Script Rules' folder under the workspace



Rules

In the top part of the Window 'Dynamics Wizard' is a list of all available rules. There are two kinds of rules. The 'Java Rules' marked with are implemented as a java class. They are integrated in SDS and it is necessary to edit the source code of the plug-ins to add new rules. The 'ECMA Script Rule' or 'Java Script Rule' marked with are located in the CSS workspace in the folder 'SDS Script Rules'. To add a new rule you have to store the java script file in the folder and it is available for the properties.

Rule Filter

The selection of rules in the list depends on the type of property. The return type of the rule has to match the property type. In the first line of the rule the return type is set. It is possible to use all Java types as return types.

Parameter

The lower section holds the list of input and output channels that the rule defines. In the value field you can use the alias names that are defined in the widget or display instead of the process variable name. It is marked by the symbol '$' at the beginning and end of the String.

ACTION DATA

Action data setting defines the action which is performed when a widget is activated (i.e. say a button is clicked)

Actions can be of two types

- Sending value
- Open a display

Each widget can have arbitrary no. of action data

Action can be executed from the contribution menu as well. The contribution menu is invoked by right clicking a widget -> CSS -> <contribution menu>

CURSORS

Cursor can be changed as the mouse pointer moves over some widget

Widget properties have a field for changing cursor

There are two kind of cursors

1. System cursors
2. Other – action enabled, action disabled

DEFAULT DISPLAY

1. Default display is the default SDS file which opens through the contribution menu (open display)
2. To set the default display go to main menu -> CSS -> Preferences -> CSS Applications -> Display -> Default Display
3. Set the default file and the alias name

QUICKSTART

1. Quickstart menu is used to add shortcut to run the SDS files directly.
2. The SDS files added to quickstart menu will directly open in execute mode
3. To add files to quickstart menu, right click the SDS file in navigator pane -> add to quickstart

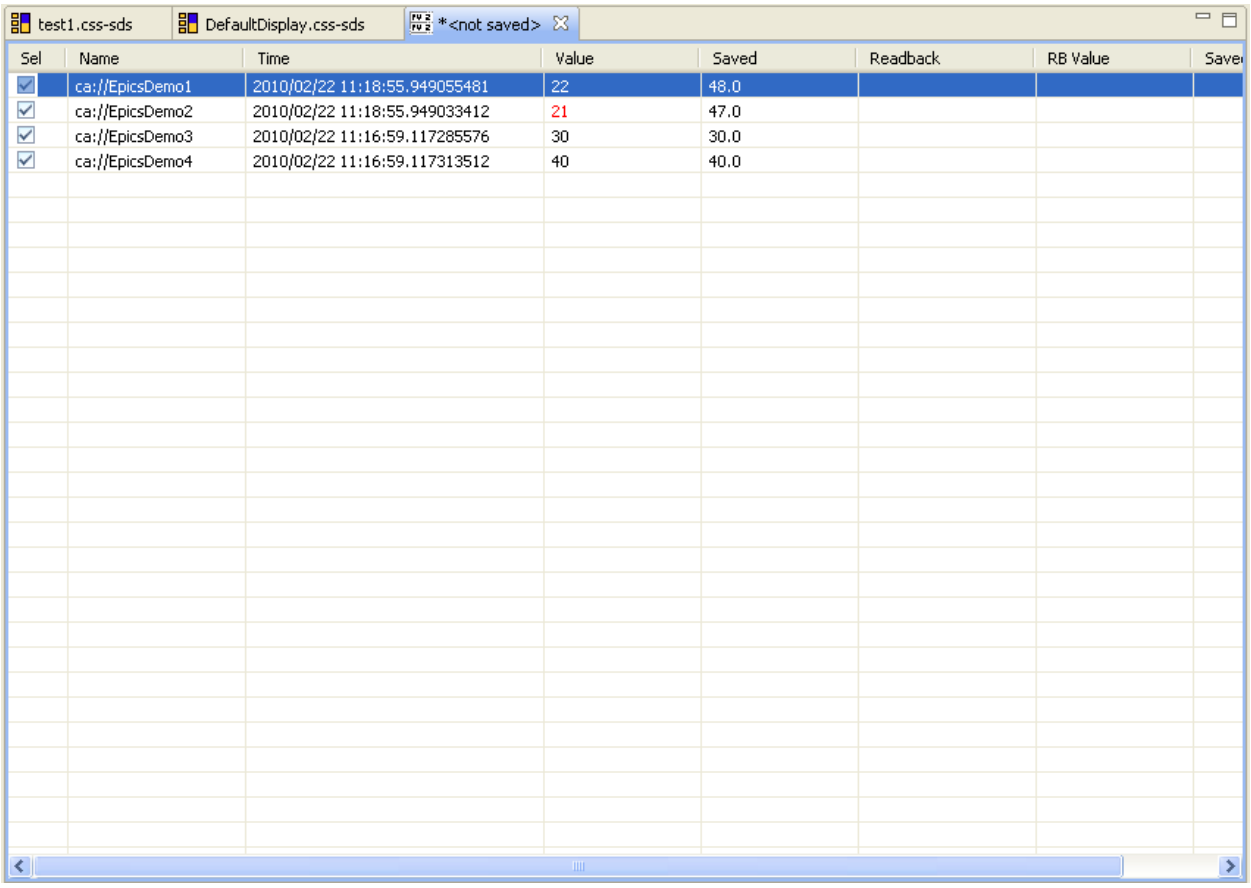
4. To invoke displays directly from quickstart, go to main menu -> quickstart -> (click file name)
5. To edit quickstart preferences go to main menu -> CSS -> Preferences -> CSS Applications -> Utilities -> Quickstart

The ADL Converter is a CSS Plug In that converts Control System Displays. This converts only from ADL-Format to the Synoptic Display Studio format from the Control System Studio.

1. To start ADL converter, go to main menu -> CSS -> Display -> ADL Converter
2. Click 'File' to upload MEDM/DM2K file (.adl file)
3. Select path of directory for the destination file (i.e. for the css-sds file)
4. Click convert
5. Multiple files can be converted at once
6. Add multiple files by above procedure, or add folder which contains multiple files (Subfolder;)

New Conversion

To convert other set of ADL-Files, clear the list by "Clear" button and choose the new ADL-Files and a new target workspace path. Then start the new conversion

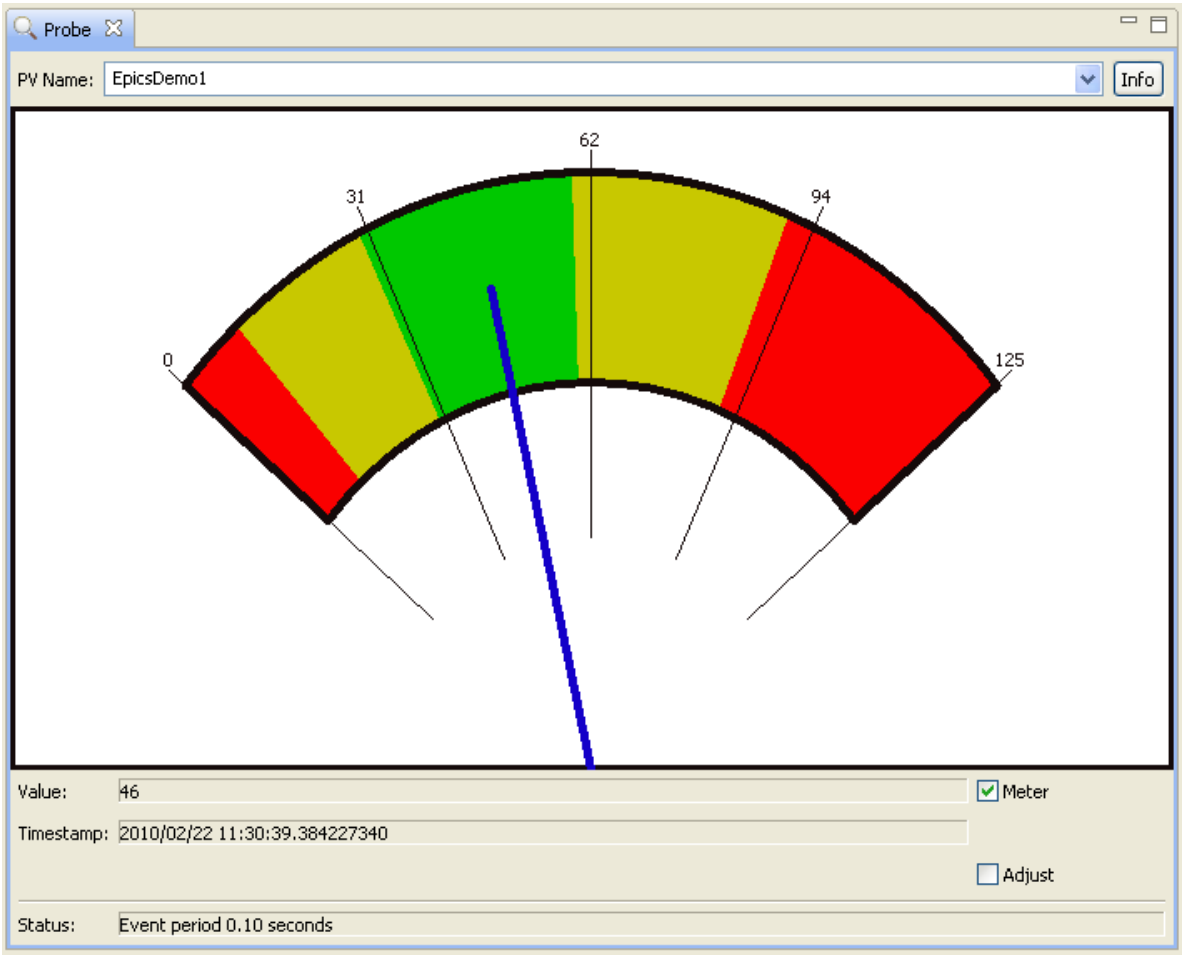


The PV Table provides a tabular view of PV names and their current value. One can start and stop live value updates. In addition, one can take a "snapshot" of current values, which gets saved when saving the PV table document. The PV table display indicates current values that differ from the "snapshot" values in red.

One can configure the update rate as well as the threshold for indicating differences between the current values and those from the snapshot.

To start with PV table

1. Go to main menu -> CSS -> Display -> PV Table
2. Right click the empty area in the table ->
3. Write the PV name to be monitored, go on adding names
4. A green LED symbol at the top icon bar is for starting the updates in PV table.
5. The update rate can be configured by right click on the PV table -> config



The Probe tool allows basic reading and writing of PVs.

Usage

1. Enter a name into the PV name text box and enter. The tool will display the current value of the given PV together with time stamp and status.
2. The 'Adjust' check box opens a dialog for writing a new value to the PV.
3. The status bar provides error messages. If all is OK, it displays a slowly averaged update rate of the PV.
4. It also displays alarm zones on the meter
5. The meter can be disabled also from the ‘Meter’ check box

Control panels for the PF-AR Vacuum are designed in Synoptic Display Studio. Two panels are created for the display. The first one is an exact replica of the previous panel that was designed in DM2K. The second one is the modified version with gradient color coding. The color gradient is coded in JavaScript.

The details of the file are :

1. C:\css-1.2.0-win32\ css-1.2.0-win32\css\workspace\prachi\PFARVacOld.css-sds

   Replica of the previous panel

2. C:\css-1.2.0-win32\ css-1.2.0-win32\css\workspace\prachi\PFARVacNew.css-sds

   The modified version

3. C:\css-1.2.0-win32\ css-1.2.0-win32\css\workspace\SDS Script Rules\continousColorb2r_dis.css-sdss

   Color gradient script

4. C:\css-1.2.0-win32\ css-1.2.0-win32\css\workspace\prachi\wide_rainbow.gif

   GIF image placed on the panel

The synoptic display files are also added at the quickstart menu.

REFERENCES

- Documentation from http://css.desy.de
- CSS Help and Training Examples