

MULTI CHANNEL APPLICATIONS FOR CONTROL SYSTEM STUDIO (CSS)

K. Shroff, G. Carcassi, BNL, Upton, Long Island, New York, USA

R. Lange, HZB, Berlin, Germany

Abstract

With the development of the ChannelFinder[1][2] directory service and the PVManager[3] client library, a new set of applications have been developed for Control System Studio(CSS)[4]. These applications have simplified user interaction by requiring the user to provide them with only the criteria of the channels they are interested in, instead of the complete set of channels. They have improved performance with the use of PVManager to manage the collection of control system data.

Architecture

The figure 1. shows the architecture of the new CSS applications which use the ChannelFinder service along with PVManager and ChannelFinderClient library plugins. The applications use the ChannelFinderClient plugin, to query the service for a group of channels based on set of criteria which might include positional, physical or operations properties and/or tags associated with the channels, the resulting channels are then used by PVManager to create channel access (CA) connections to retrieve their values.

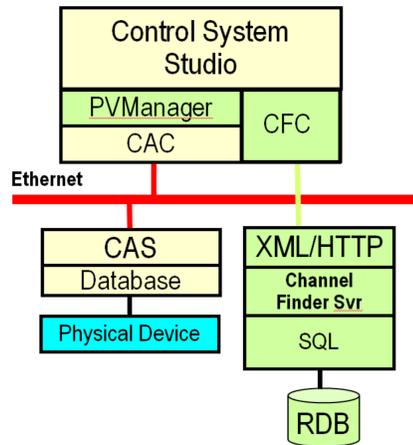


Figure 1 : Architecture .

CSS Applications

ChannelViewer

The ChannelViewer is a simple graphical application which can be used to query the channel finder. The input consists of a query can be based one or any combination of the criteria: Channel name, Property values, Tags. The figure 2. shows the result of a query for all the channels in the Storage ring associated with Beam Position Monitor (BPM) or Horizontal Correctors (HCOR) with the tag aphla.sys.SR. The resulting Channels can be sorted, grouped and tagged. They can be exported to any other CSS application as a set of process variables, thus the end user need not remember details of each process variable but can rather work with logical group determined by the query conditions.

Channel Name	Owner	handle	girder	cell	ordinal	devName	elemName	elemField	elemType	sEnd	length
SR-C01-BI-G02A...	d-update	READBACK	G2	C01	120	PL1G2C01A	PL1G2C01A	x	BPM	29.9886	0.0
SR-C01-BI-G02A...	d-update	READBACK	G2	C01	120	PL1G2C01A	PL1G2C01A	y	BPM	29.9886	0.0
SR-C01-BI-G02A...	d-update	READBACK	G2	C01	120	PL1G2C01A	PL1G2C01A		BPM	29.9886	0.0
SR-C01-BI-G02A...	d-update	SETPOINT	G2	C01	120	PL1G2C01A	PL1G2C01A		BPM	29.9886	0.0
SR-C01-MG-G02...	d-update	SETPOINT	G2	C01	120	PL1G2C01A	PL1G2C01A	x	HCOR	30.6673	0.2
SR-C01-MG-G02...	d-update	READBACK	G2	C01	125	CL1G2C01A	CL1G2C01A	x	HCOR	30.6673	0.2
SR-C01-MG-G02...	d-update	READBACK	G2	C01	133	CL2G2C01A	CL2G2C01A	x	HCOR	32.1047	0.2
SR-C01-MG-G02...	d-update	SETPOINT	G2	C01	133	CL2G2C01A	CL2G2C01A	x	HCOR	32.1047	0.2
SR-C01-BI-G02A...	d-update	READBACK	G2	C01	138	PL2G2C01A	PL2G2C01A	y	BPM	32.5523	0.0
SR-C01-BI-G02A...	d-update	READBACK	G2	C01	138	PL2G2C01A	PL2G2C01A	x	BPM	32.5523	0.0
SR-C01-BI-G02A...	d-update	SETPOINT	G2	C01	138	PL2G2C01A	PL2G2C01A		BPM	32.5523	0.0
SR-C01-BI-G02A...	d-update	SETPOINT	G2	C01	138	PL2G2C01A	PL2G2C01A	y	BPM	32.5523	0.0
SR-C01-BI-G02A...	d-update	READBACK	G2	C01	138	PL2G2C01A	PL2G2C01A		BPM	32.5523	0.0
SR-C01-BI-G02A...	d-update	READBACK	G2	C01	138	PL2G2C01A	PL2G2C01A		BPM	32.5523	0.0
SR-C01-MG-G04...	d-update	READBACK	G4	C01	150	SQMG4C01A	OMG4C01A	x	HCOR	36.7222	0.2
SR-C01-MG-G04...	d-update	SETPOINT	G4	C01	150	SQMG4C01A	OMG4C01A	x	HCOR	36.7222	0.2
SR-C01-BI-G04A...	d-update	SETPOINT	G4	C01	161	PM1G4C01A	PM1G4C01A		BPM	38.3018	0.0
SR-C01-BI-G04A...	d-update	SETPOINT	G4	C01	161	PM1G4C01A	PM1G4C01A		BPM	38.3018	0.0
SR-C01-BI-G04A...	d-update	READBACK	G4	C01	161	PM1G4C01A	PM1G4C01A	x	BPM	38.3018	0.0
SR-C01-BI-G04A...	d-update	READBACK	G4	C01	161	PM1G4C01A	PM1G4C01A		BPM	38.3018	0.0
SR-C01-BI-G04A...	d-update	READBACK	G4	C01	161	PM1G4C01A	PM1G4C01A	y	BPM	38.3018	0.0
SR-C01-BI-G04B...	d-update	SETPOINT	G4	C01	171	PM1G4C01B	PM1G4C01B		BPM	40.5345	0.0
SR-C01-BI-G04B...	d-update	SETPOINT	G4	C01	171	PM1G4C01B	PM1G4C01B		BPM	40.5345	0.0
SR-C01-BI-G04B...	d-update	READBACK	G4	C01	171	PM1G4C01B	PM1G4C01B		BPM	40.5345	0.0

Figure 2: The Channel Viewer.

MultiChannel Viewer

The MultiChannel Viewer application shown in Figure 3 is designed to plot the values of a group of channels which are ordered based on a user specified criteria i.e. s-position. The MultiChannel viewer accepts a query (similar to the one used in ChannelViewer) which represents a logical group of channels on the ChannelFinder server. The result of this query, a group of channels, is then plotted with their order along the x-axis being determined by the "Order By" property selected.

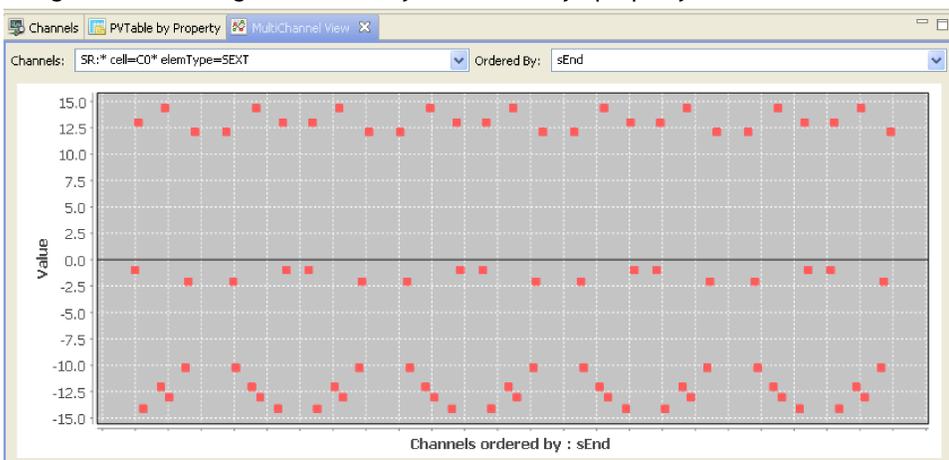


Figure 3a: MultiChannel Viewer displaying plotting all the sextupole values from cell C01-C09 ordered by sEnd.

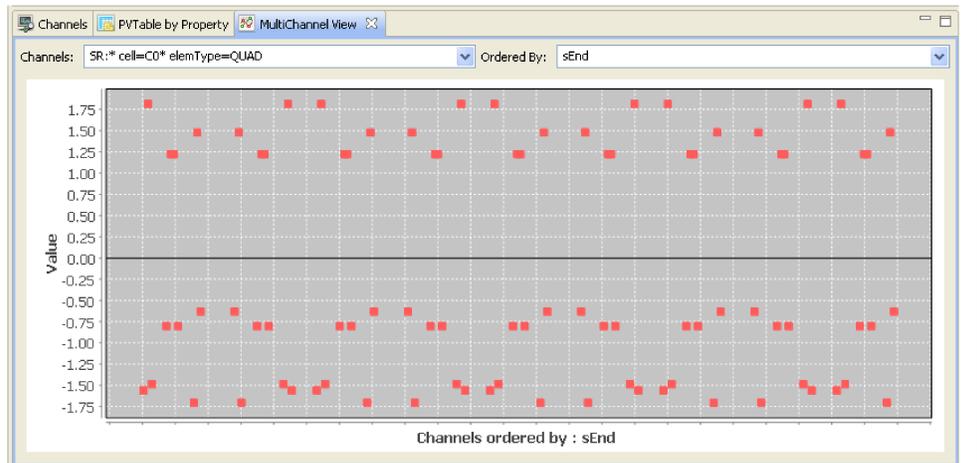


Figure 3b: MultiChannel Viewer displaying plotting all the quadrupole values from cell C01-C09 ordered by sEnd.

PVTable by Property

Figure 4 shows the PVTable by Property application. It allows user to organize the output of many channels in a table, based on the value of two properties, thus each cell of the table represents the current value of a single channel, the position of each channel is determined by the value of selected row and column properties. The list of channels and the properties used to create the rows and columns are obtained by querying the ChannelFinder.

elemName \ handle	READBACK	SETPOINT
PL1G2C01A	0.0	0.0
PL2G2C01A	0.0	0.0
PM1G4C01A	0.0	0.0
PM1G4C01B	0.0	0.0
QH1G6C01B	-0.6330350682439309	-0.633004
QH2G6C01B	1.4769240496189058	1.47765
QH3G6C01B	-1.7079907946459905	-1.70755
QL1G2C01A	-1.5621500957125198	-1.56216
QL2G2C01A	1.8123112550349432	1.81307
QL3G2C01A	-1.4895086244211488	-1.48928
QM1G4C01A	-0.8027569777272626	-0.803148
QM1G4C01B	-0.8031907401945524	-0.803148
QM2G4C01A	1.2219073476653697	1.2223
QM2G4C01B	1.2220868085732814	1.2223

Figure 4: PVTable by Property: showing the readback and setpoint channels associated with each element.

WaterFallPlot

The Waterfall plot shown in Figure 5. creates a time plot for all the values of a channel query or of a waveform.



Figure 5: WaterFall plot for all x field readback channels in storage ring.

Conclusion

Use of the ChannelFinder service and client eliminates the need for configuring and remembering each individual channel.

PVManager simplifies and improves performance of applications by handling various problems associated with use of large number of channel.

References

- [1] L. Dalesio, D. Dohan, R. Lange, A Proposal for introspection in EPICS, THP036, ICALEPS (2009).
- [2] R. Lange, ChannelFinder: A directory service; <http://channelfinder.sf.net>.
- [3] G.Carcassi, pvManager, EPICS Fall meeting (2010) ; <http://pvmanager.sf.net>
- [4] Control System Studio; <http://cs-studio.sf.net>