



CDI News

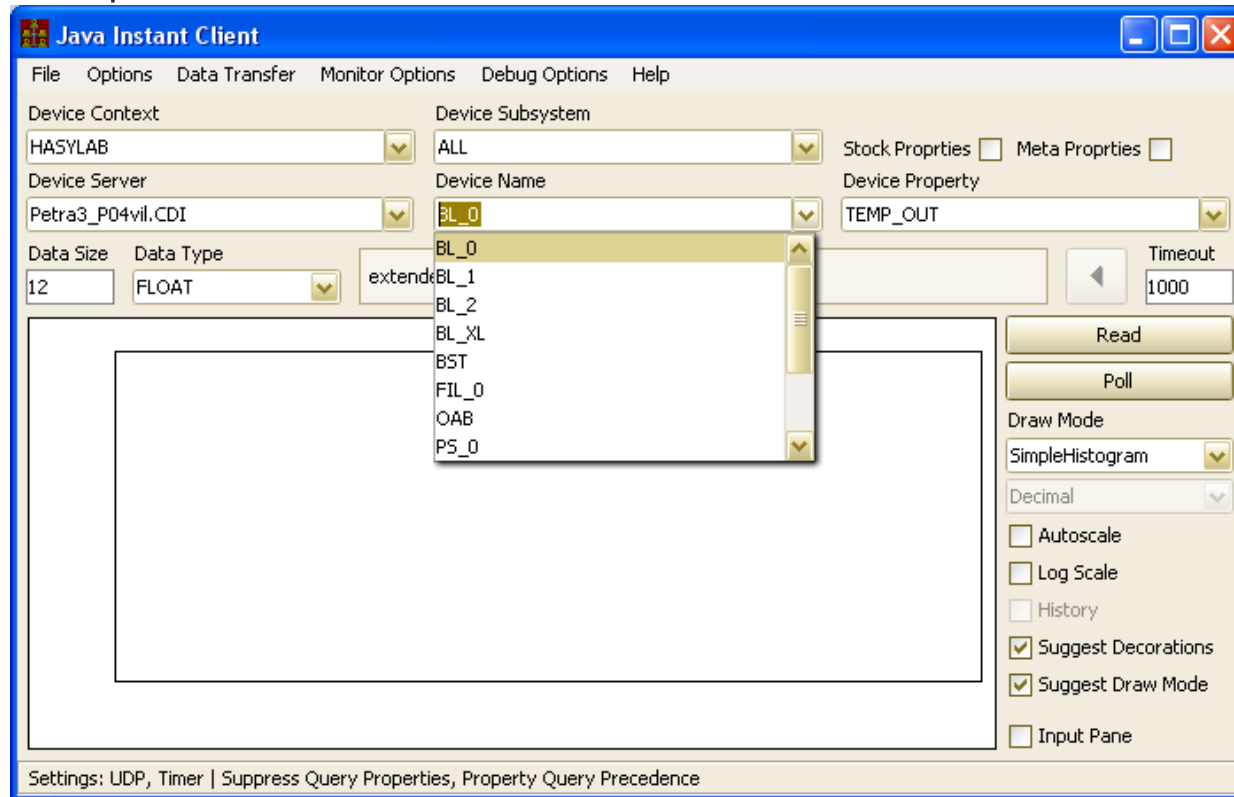
[New Features ...]

- CDI Version at 1.1.0 (supersedes 1.0.4)
- Feature addition:
 - Optional supplementary device server
 - ‘active’
 - can schedule data
 - a form of ‘reflected’ memory for ‘those who don’t want to develop their own device server’.
 - Issue:
 - some hardware is ‘remotely’ access via tcp over the ethernet (EtherCat, Siemens SPS).
 - Each CDI device access is a separate connection and transfer.
 - CDI can only ‘group’ calls with ‘atomic device lists’.

“RECV” + “BL_0.TEMP_OUT, BL_1.TEMP_OUT, BL_2.TEMP_OUT,…”

New Features ...

- Feature addition:
 - Optional supplementary ‘companion’ device server
 - The ‘extended’ CDI properties do this automatically



[CDI Companion Device Server]

- Need a 'cdiserv.csv' file with instructions.
 - Only the 'extended' properties are allowed
 - Note: these map to the relevant atomic list of CDI devices and bus property "RECV.CLBR"
 - "RECV.CLBR" is also available.
 - If any extended property is scheduled, so is "RECV.CLBR"
 - Note: the Alarm system requires registered devices (and not property Keywords).

CDI Companion Device Server

CDI Companion Device Server

Another and in most cases better way of scheduling data from a CDI Server is to set up a 'companion' device server. This is another (in-process) exported device server which essentially turns an otherwise passive CDI server into an actively updating device server capable of scheduling data to clients. If 'exclusive read' (see above) is in play, then clients in general will not be able to access data directly from the CDI server itself. The companion * device server, on the other hand, is in-process and setup to readout all relevant data at a configurable readback update rate, calibrate them, and make them available to outside client. This is in effect a way of reflecting the hardware memory (which may be at a remote location on the ethernet) to a local server. The data readback can also check assigned tolerances and schedule data to clients as necessary.

The companion server deals with 'calibrated' data and as such can re-export all of the 'extended properties' of the standard CDI server as well as its 'READ.CLBR' property. However, which of these is re-exported is a question of setting up the proper database.

This is an additional .csv File called 'cdiserv.csv', which if present will signal the CDI server to read its contents and export a companion server, which by default will have the same name as the CDI server with the final decoration ".SRV" appended. This name can be assigned either by making use of the environment variable (see above) CDI_DEVICESERVERNAME or the api call `cdiSetDeviceServerExportName()`.

```
Property, Device, Format, Schedule_Tolerance, Interval
P1, test1, Int32, 0, 100
```

In the above file, the column 'PROPERTY' is required and must be a valid extended property. Any or all of the known extended properties can be included (one per line), even if it refers to a single scalar readout.

Optional columns include

- 'DEVICE' should specify the associated extended property device used to access the information from the CDI server. If omitted, a '#0' will be used in its stead. For those scalar extended properties where the associated device has the same name as the property, then this become irrelevant.
- 'FORMAT' gives the desired data format to use to access the extended property from the CDI server. If omitted, 'CF_FLOAT' is used. This is the default registered data type for any extended property as the calibration rules will be applied, which often render the data readback into a floating point number. If it is known that the values should be read back as for example a 'short integer' then one should fix that here.
- 'SCHEDULE_TOLERANCE' gives the tolerance string to use for scheduling data to clients. If omitted, then the readback data will not be scheduled. Otherwise the data will be compared against the previous readback in making the scheduling decision (same logic as noted above).
- 'SIZE' gives the desired datasize of the extended property access. If omitted, then the full registered property size will be used.
- 'INTERVAL' gives the desired readback update interval in milliseconds. If omitted, a value of 1000 is used (corresponding to readback at 1 Hz).
- 'KEEP_HISTORY' is a boolean indicating whether the companion device server should maintain a local history of the extended property. If 'TRUE' this will apply the 'default' local history criteria of a short-term depth of 600 entries in a ring buffer, 1 month storage on the local disk, and readbacks at 1 Hz, storing to disk when a tolerance of 10% has been exceeded or a heartbeat of 15 minutes has passed. If more refined local history criteria are needed, then the use of a history.csv should be employed.

Calibration Rules

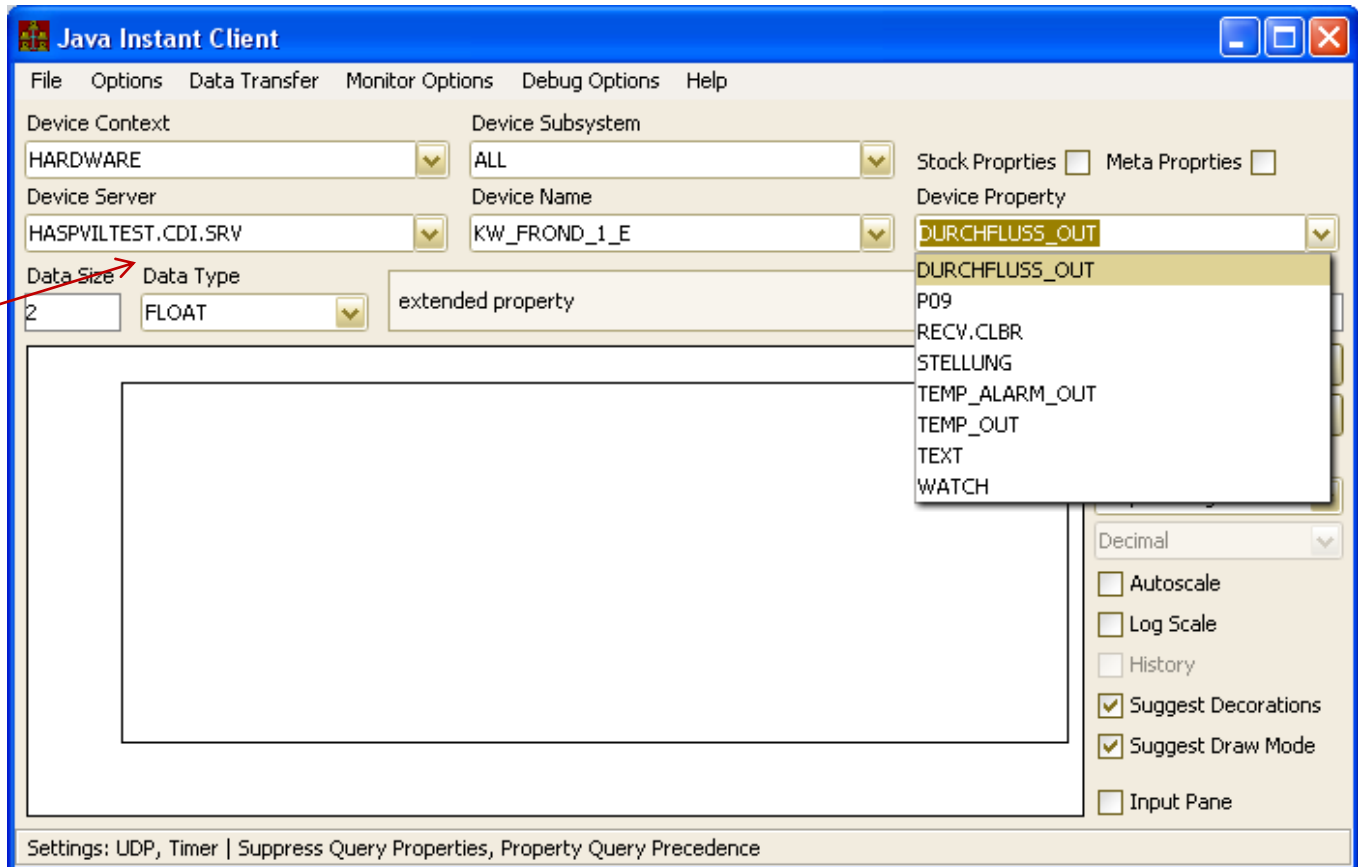
As noted in the above section on configuration details, you can supply calibration rules to both input data (data to send to the bus) and output data (data to receive on the bus). Data

CDI Companion Device Server

- Example (@ Hasylab)

default:
append '.SRV'
to CDI server
name.

can also set
via API or
environment.



[CDI Companion Device Server]

- Other considerations:
 - All external clients should use CDI **companion device server** !
 - Keep everyone else away from the **CDI server**
 - **ALL** CDI server bus properties are registered with **exclusive READ** (if companion device server registered) !
 - Add an **'ipnets.csv'** and/or **'users.csv'** to prevent any external clients reading accessing the CDI server.