

Tine Code Generators



(But, you're still free to copy-
and-paste)

Server Setup Wizards

- ❑ I know my hardware/middle layer and the information it provides.
- ❑ I want to make this information available to the control system without really knowing what's going on "under the hood".
- ❑ I don't want to have to know much about the TINE APIs.

TINE Server

- ❑ FEC (Front End Controller)
 - Can have more than 1 FEC on a host computer
 - ❑ Need to ensure unique FEC names.
 - ❑ Need to ensure unique FEC ports (port offsets).
 - Manages Equipment Modules
 - ❑ Has a “local” identifying name (managed name)
 - ❑ Has a System-wide unique “Exported” Device Server name.
 - ❑ Have named Devices
 - ❑ Have named Properties
 - ❑ Belong to a Control System Context
 - ❑ Belong to a Subsystem

TINE Server Hierarchy

□ Computer

■ FEC1 (PortOffset 0)

□ EQM1

- Device1, ...

- Property (device specific properties)

- Property1, ...

- Device (property specific devices)

□ EQM2

- (etc.)

■ FEC2 (PortOffset 1)

□ (etc.)

□ Naming Scheme:

`/<context>/<server>/<device> [<property>]`

- Doocs: `<facility>/<device>/<location>/<property>`

- Tango: `<namespace>/<class>/<member>`

- Epics: flat namespace with ':' as delimiters

`(/<Context>/<server>)`

`(<device name>)`

`(<property>)`

TINE Server Hierarchy

(as seen by the client)

Advanced browsing

Access protocol: **TINE** Show stock properties

Device context	Device subsystem	Device server	Device name	Property name
DEFAULT	ALL	BEAMLINEINFO	Buffer-0	ArrayDepth
CSL	DIAG	DOBUNCHE	Buffer-1	ArrayPointer
Common.SEDAC	HIST	DOBeamLineMon	Buffer-2	BufferIndex_FastTau
Common.Daemon	MEX	DOIDC		BufferIndex_Tau
DESY2	RF	DoNeBu		Charge
DESY2.DAEMON	RPT	DoSRVideo		Chi
DESY3	SER	SRMONITOR		Chi2
DESY3.DAEMON	SUB	TestIDCDoris		Chi2Max
DORIS	TUNE			Current
DORIS.DAEMON	VAC			DVMIntTime
DORIS.SEDAC	TEST			DVMLoopTime
ELHEP				DVMStatus
HARDWARE				DVMStatusText
HASYLAB				DoIDCmBunch
HERA				DoIDCmBunch.HIST
HERA.DAEMON				I
HERA.SEDAC				InitGPIB
HERA.PVAK				KeepTiefe
LAB				Ladung
LAB.VAC				Lifetime
LAB.TEST				Loss
LINAC				LossArray
LINAC2				Loss.HIST
LINAC2.DAEMON				MaxDlCalc
LINAC3				MaxLossCalc

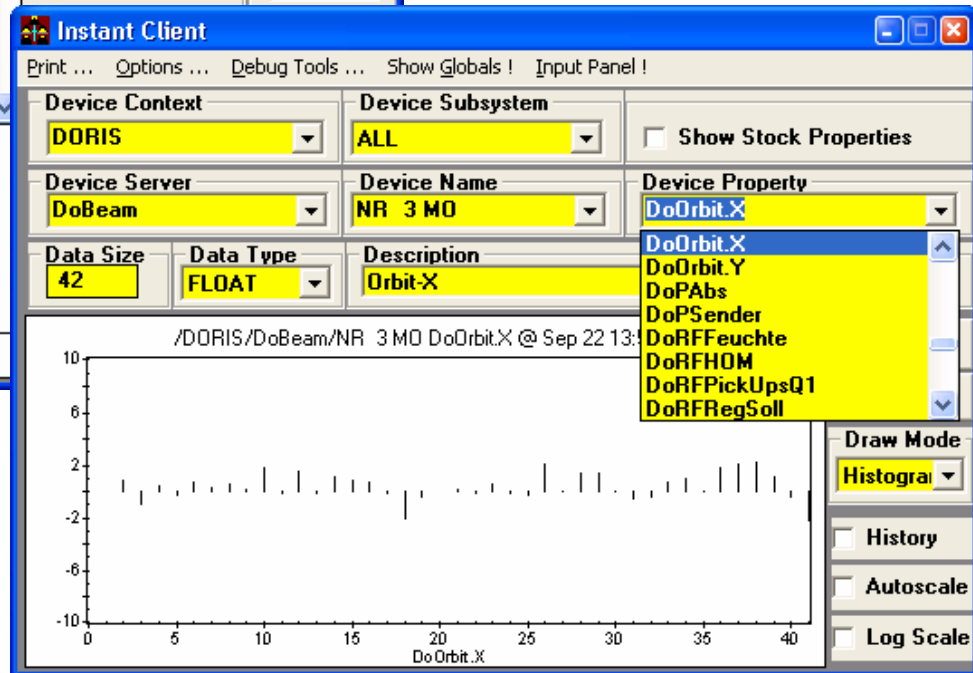
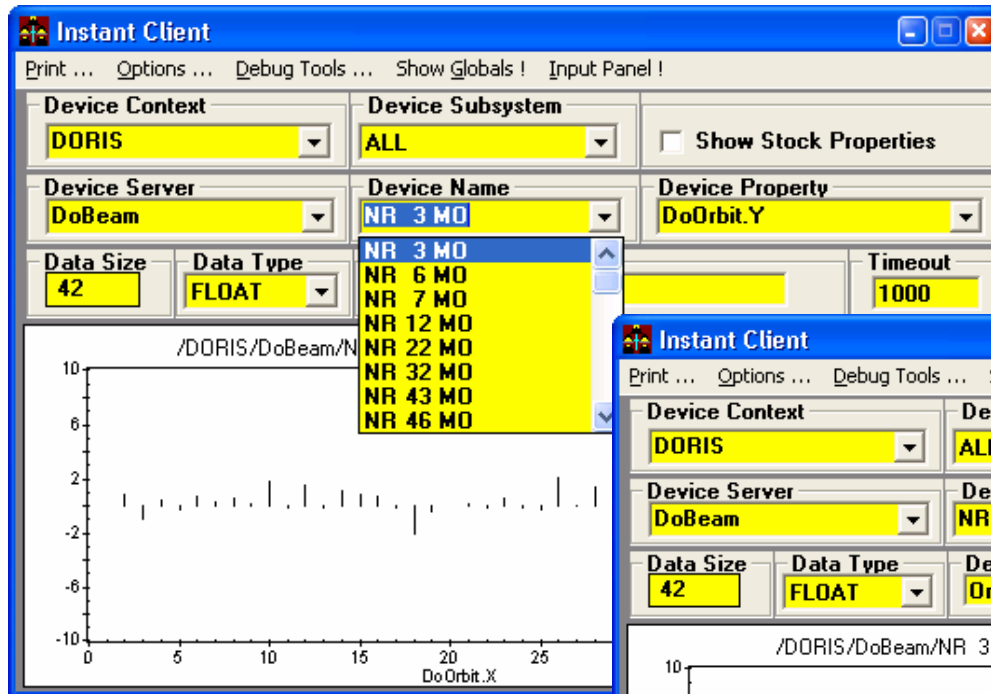
Access rate: 1000

Access mode: **POLL**

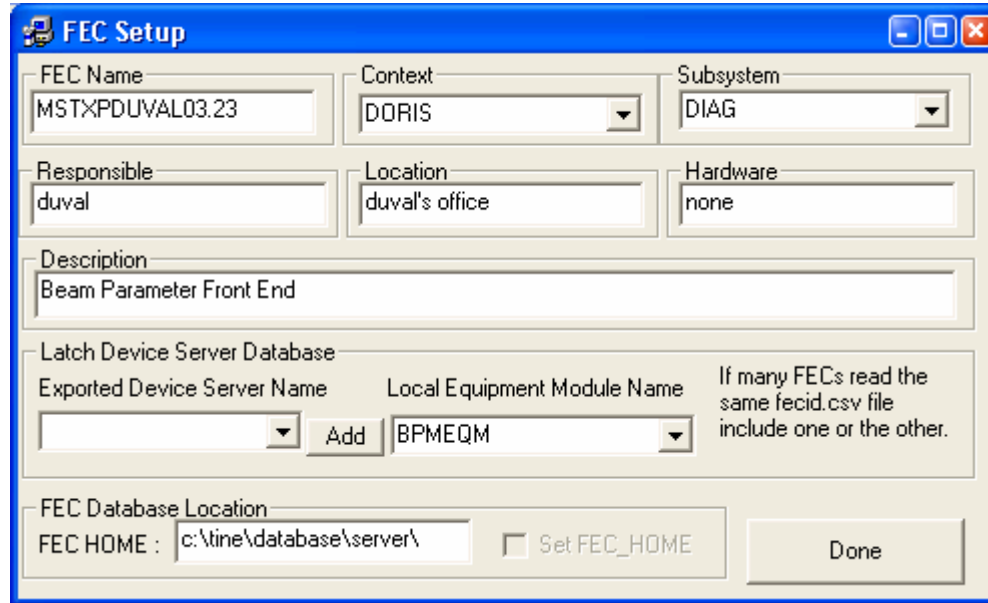
TINE/DORIS/DOIDC/Buffer-0/POLL,1000,-1

TINE Server Hierarchy

(as seen by the client)



Getting there with a Wizard



The screenshot shows the 'FEC Setup' window with the following fields and values:

FEC Name	Context	Subsystem
MSTXPDUVAL03.23	DORIS	DIAG
Responsible	Location	Hardware
duval	duval's office	none
Description		
Beam Parameter Front End		
Latch Device Server Database		
Exported Device Server Name	Local Equipment Module Name	If many FECs read the same fecid.csv file include one or the other.
<input type="text"/>	BPMEQM	
FEC Database Location		
FEC HOME :	<input type="text" value="c:\tine\database\server\"/> <input type="checkbox"/> Set FEC_HOME	Done

Generations configuration File only!

The FEC name is established here.

Getting There with a Wizard

The screenshot displays the **TINE Device Server Setup Wizard** window. The main window is divided into several sections:

- Guidance**: Shows the current step in the wizard.
- Device Server Information**: Fields for Export Name (DoBeqm), Local Name (BPMEQM), and Number of Devices (42).
- Property Information**: Fields for Property Name (Orbit.Y), Description (Vertical Orbit), Access (READ), Max Value (10), Min Value (-10), Data Format (NULL), Data Output (Single), Units (mm), Max Data Size (0), and Output Array Type (CHANNEL). There are also checkboxes for Device List, Alarm Watch, Keep History (checked), and Property is Process.
- History Panel**: Fields for Calling Device Name (#0), Archive Rate (secs) (10), Polling Rate (msec) (1000), Archive Heartbeat (secs) (18000), Ringbuffer Depth (600), Long term Depth (months) (1), and Tolerance (10%).
- Alarm Watch Panel**: Fields for Calling Device Name (#0), Alarm System ID (0), Severity (high) (10), Counts Threshold (3), and Value too high (0) with a warning level (0).
- Project Directory**: Field for Project Directory (H:\DoBeqm).
- Device List**: A list showing "[DoBeqm] Orbit.X <Horizontal Orbit>".
- Device Panel**: Radio buttons for Server (selected) and Property, and a field for Device 0.

Two dialog boxes are overlaid on the main window:

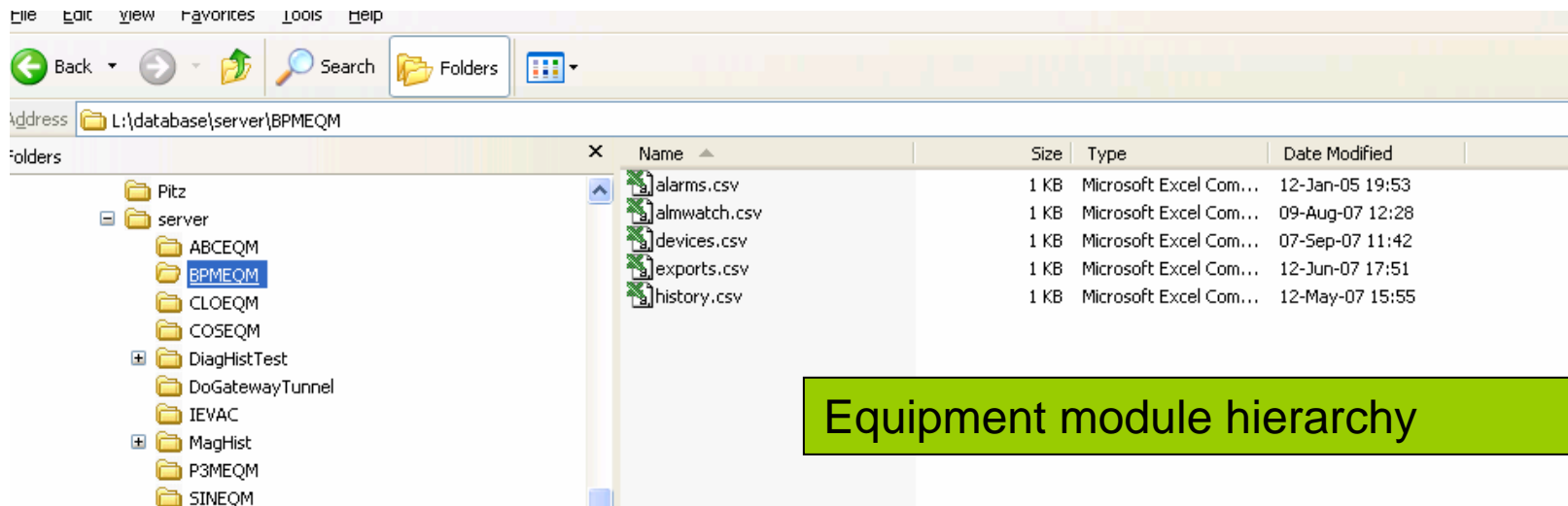
- Write to FEC_HOME repository?**: A question dialog asking "Do you want to update the database files under c:\tine\database\server\BPMEQM\ ?" with Yes and No buttons.
- Add New Property with these characteristics ?**: A confirmation dialog showing the property information being added: Export Name : DoBeqm, Local Name : BPMEQM, Property Name : Orbit.Y, Return Data : 42 float values (CHANNEL), Input Data : none, Property Access : READ, Maximum Value : 10, Minimum Value : -10, Data Units : mm, Description : Vertical Orbit, Keep History : Yes, Use Alarm Watch : No, Has Device list : No. It has OK and Cancel buttons.

What gets Generated?

Release 3.xx: csv configuration database:

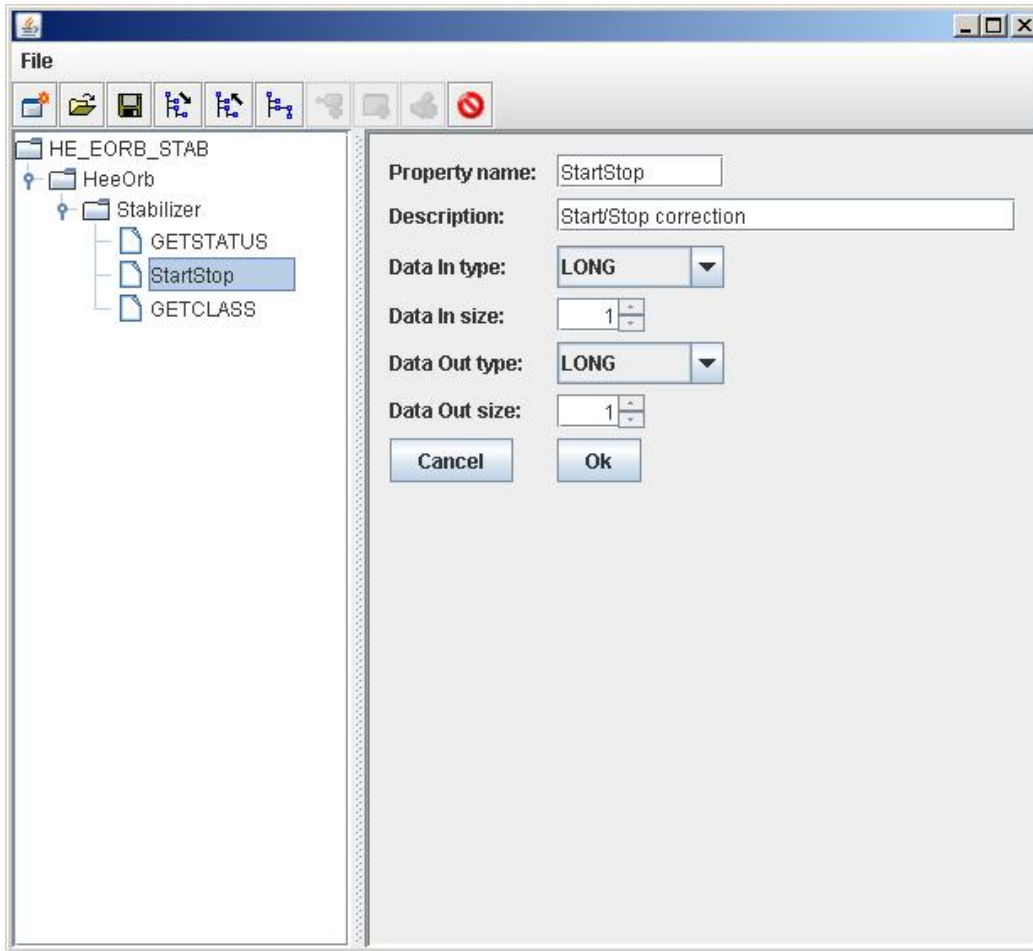
	A	B	C	D	E	F	G	H	I
1	Fec_Name	Context	Local_Nam	Port_Offse	SubSystem	Description	Location	Hardware	Responsible
2	MSTXPDUVAL03.16	TEST	WINEQM	18	TEST		duval's offic	none	duval
3	MSTXPDUVAL03.19	TEST	JEQM	19	TEST		duval's offic	none	duval
4	MSTXPDUVAL03.18	TEST	SINEQM	18	TEST		duval's offic	none	duval
5	MSTXPDUVAL03.17	TEST	SEQM	17	TEST		duval's offic	none	duval
6	MSTXPDUVAL03.23	DORIS	BPMEQM	23	DIAG	Beam Parameter Front End	duval's offic	none	duval
7									

fecid.csv



Equipment module hierarchy

Tree-style Server wizard (Release 4.0)



What gets Generated?

Release 4.xx: xml
configuration
database option:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <COMPUTER>
- <FEC>
  <NAME>MSTXPDUVAL03.23</NAME>
  <PORT_OFFSET>23</PORT_OFFSET>
- <EQM>
  <NAME>BPMEQM</NAME>
  <SERVER>DoBeam</SERVER>
  <CONTEXT>DORIS</CONTEXT>
  <SUBSYSTEM>DIAG</SUBSYSTEM>
  <DEVICE_SPACE>42</DEVICE_SPACE>
- <DEVICE>
  <NAME>NR 3 MO</NAME>
</DEVICE>
- <DEVICE>
  <NAME>NR 6 MO</NAME>
</DEVICE>
- <DEVICE>
  <NAME>NR 7 MO</NAME>
</DEVICE>
- <PROPERTY>
  <ID>1</ID>
  <NAME>OrbitX</NAME>
  <DESCRIPTION>[-10:10 mm]Horizontal Orbit</DESCRIPTION>
  <SIZE_OUT>42</SIZE_OUT>
  <DTYPE_OUT>float.CHANNEL</DTYPE_OUT>
  <ACCESS>READ</ACCESS>
  <REDIRECTION />
</PROPERTY>
- <PROPERTY>
  <ID>2</ID>
  <NAME>OrbitY</NAME>
  <DESCRIPTION>[-10:10 mm]Vertical Orbit</DESCRIPTION>
  <SIZE_OUT>42</SIZE_OUT>
  <DTYPE_OUT>float.CHANNEL</DTYPE_OUT>
  <ACCESS>READ</ACCESS>
  <REDIRECTION />
</PROPERTY>
</FOM>
```

Single xml file:
fec.xml

What gets Generated

```
void bpmeqm_exe(void)
{
    /* TODO: put your shutdown routines here */
}

int bpmeqm(char *devName, char *devProperty, DTYPE *dout, DTYPE *din, short access)
{
    int devnr, prpid, i, cc;

    /* TODO: If READ properties take input data, include code to examine the contents of din. */
    /* If different actions need to be taken at the start or end of a link, examine the */
    /* 'access' parameter against CA_FIRST or CA_LAST. */
    /* If allow format overloading (you return different data according to the request */
    /* format), then replace calls to putDataFromShort() etc with the desired code. */

    prpid = GetPropertyId(BPMEQM_TAG, devProperty);

    switch (prpid)
    {
        case PRP_ORBIT_X:
            if (access & CA_WRITE) return illegal_read_write;
            if (dout->dArrayLength > 0)
            {
                if (dout->dArrayLength > PRP_ORBIT_X_SIZE) return dimension_error;
                if ((cc = putValuesFromDouble(dout, &g_orbit_x, PRP_ORBIT_X_SIZE)) != 0) return cc;
            }
            return 0;
        case PRP_ORBIT_Y:
            if (access & CA_WRITE) return illegal_read_write;
            if (dout->dArrayLength > 0)
            {
                if (dout->dArrayLength > PRP_ORBIT_Y_SIZE) return dimension_error;
                if ((cc = putValuesFromDouble(dout, &g_orbit_y, PRP_ORBIT_Y_SIZE)) != 0) return cc;
            }
            return 0;
        ...
    }
}
```

Using with LabView, HPVee or a “Buffered Server”

- Use the wizard to generate the configuration database only.
- Labview: use the lvSrvInit.Vi
 - lvSrvPushXXX.Vi ,lvSrvWaitCmd.Vi, etc.
- HPVEE: use the Srv.ocx ActiveX control.
- Buffered Server: AttachServer

TO DO :

- ❑ Current Wizard can edit/append existing database but need to:
 - Preserve User Code
 - ❑ (a backup is always generated, but otherwise not integrated)
 - ❑ Introduce user-code tags.
 - ❑ Implement for C, C++
 - ❑ VB 6 will remain as is: But later support VB.NET, C#