

# New Tango based tools

C. Scafuri on behalf of:

Fabio Asnicar

Giacomo Strangolino

Lucio Zambon

Simone Di Mitri

Giuseppe Penco

and many others...

---

## Brief review of tools for FERMi@elettra:

Synoptic panels

Matrix

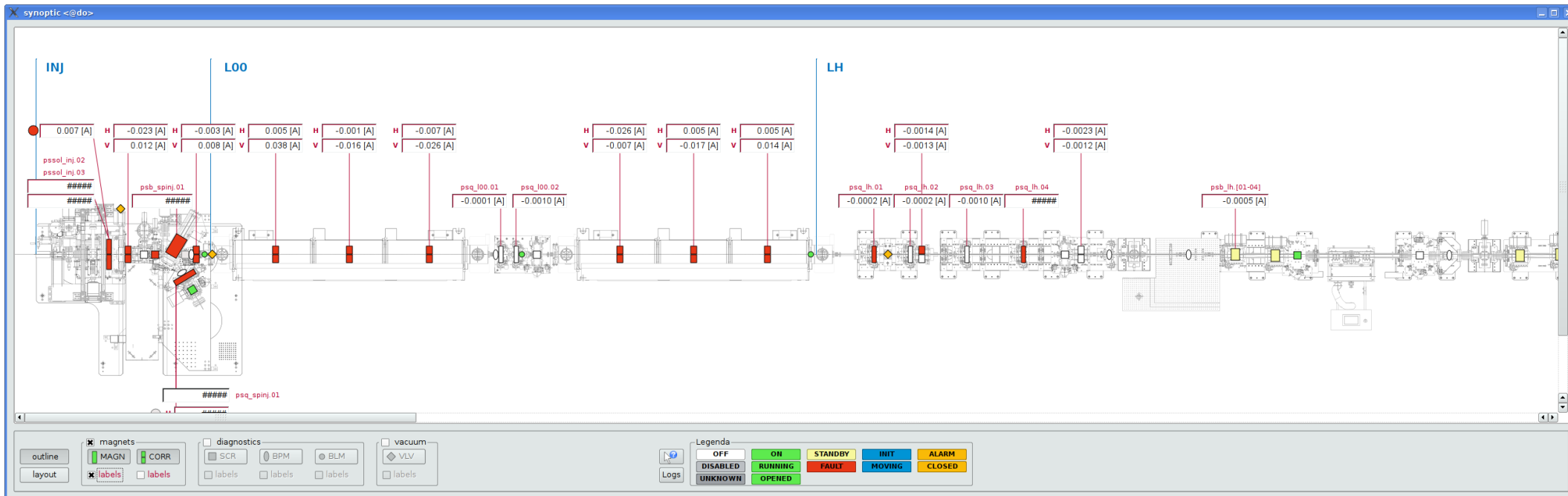
Sequencer

Qtango4 & Mango

matlab applications

Information about plant **geometry** and **status** ,  
users can **interact** with the machine

- **Geometry** derives from CAD
  - Static
  - High resolution
  - No new symbols to be learned
- **Status** is displayed by Tango aware widgets
- **Interaction** by means of Tango aware widgets



courtesy F. Asnicar

synoptic <@do>

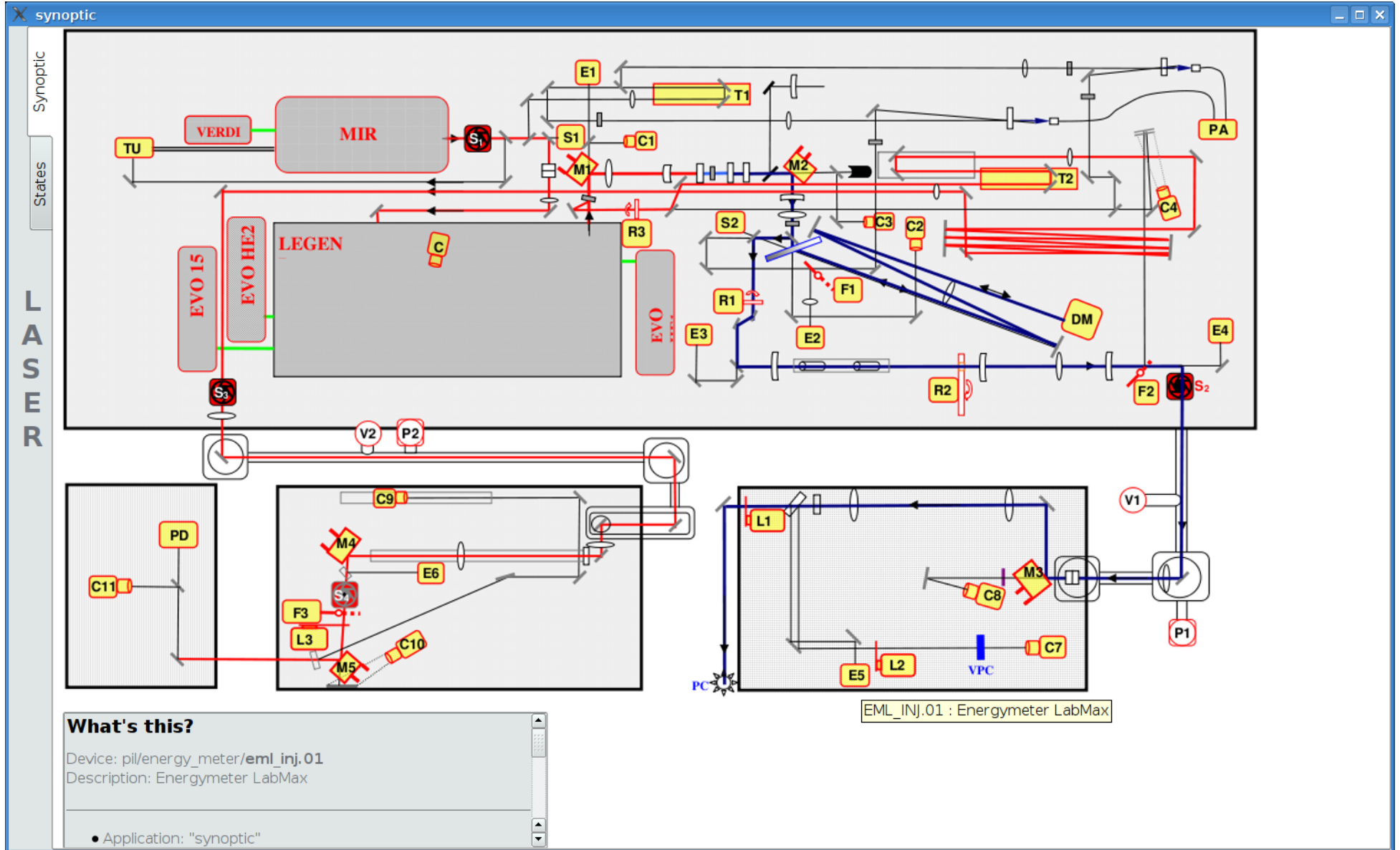
The diagram shows a synoptic view of the FERMI@elettra accelerator. It is divided into sections labeled INJ and LOO. Various components are shown with their current and voltage values:

- INJ Section:**
  - 0.007 [A] (H)
  - 0.023 [A] (H), 0.012 [A] (V)
  - 0.003 [A] (H), 0.008 [A] (V)
  - 0.005 [A] (H), 0.038 [A] (V)
  - 0.001 [A] (H), -0.016 [A] (V)
  - 0.007 [A] (H), -0.026 [A] (V)
- LOO Section:**
  - 0.026 [A] (H), -0.007 [A] (V)
  - 0.005 [A] (H), -0.017 [A] (V)
  - 0.005 [A] (H), 0.014 [A] (V)
- Other Components:**
  - psso\_inj.02, psso\_inj.03 (currents: #####)
  - psb\_spinj.01 (current: #####)
  - psq\_100.01 (-0.0001 [A]), psq\_100.02 (-0.0010 [A])
  - psq\_spinj.01 (current: #####)

**Legend:**

- magnets: MAGN, CORR
- diagnostics: SCR, BPM, BLM
- vacuum: VLV
- labels

**Legenda:** OFF, ON, DISABLED, RUNNING, UNKNOWN, OPENED



synoptic
⌵ ⌵ ⌵

LASER

		id	device	attribute	<input checked="" type="checkbox"/> all	min	value	max
States	C1	CCD_PLR01.01	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	0.142	<
			y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	-0.491	<
			sigma H	<input checked="" type="checkbox"/>	<input type="text"/>	<	9.345	<
			sigma V	<input checked="" type="checkbox"/>	<input type="text" value="0.0"/>	<	7.115	<
C2	CCD_PLR01.02	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	-0.088	<	
		y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	0.052	<	
C3	CCD_PLR01.03	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	-0.114	<	
		y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	0.188	<	
C4	CCD_PLR01.04	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	4.482	<	
		y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	0.863	<	
C7	CCD_PLR01.07	x [mm]	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
		y [mm]	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
		sigma H	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
		sigma V	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
C8	CCD_PLR01.08	x [mm]	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
		y [mm]	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
C9	CCD_LH.01	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	-0.073	<	
		y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	-0.004	<	
		sigma H	<input checked="" type="checkbox"/>	<input type="text"/>	<	1.798	<	
		sigma V	<input checked="" type="checkbox"/>	<input type="text"/>	<	1.391	<	
C10	CCD_LH.02	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	0.972	<	
		y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	-2.285	<	
C11	CCD_LH.03	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	-3.095	<	
		y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	1.389	<	

		id	device	attribute	<input checked="" type="checkbox"/> all	min	value	max
States	R1	ROTS_PLR01.01	angle [deg]	<input type="checkbox"/>	<input type="text"/>	<	#####	<
	R2	ROTS_PLR01.02	angle [deg]	<input type="checkbox"/>	<input type="text"/>	<	#####	<
	R3	ROTS_PLR01.03	angle [deg]	<input type="checkbox"/>	<input type="text"/>	<	#####	<
S1	SPCTM100_PLR01.01	center wave [nm]	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
		sigma	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
S2	SPCTM150_PLR01.01	center wave [nm]	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
		sigma	<input type="checkbox"/>	<input type="text"/>	<	#####	<	
E1	EME_PLR01.01 (A)	energy [μJ]	<input checked="" type="checkbox"/>	<input type="text"/>	<	0.000E+00	<	
E2	EME_PLR01.01 (B)	energy [μJ]	<input checked="" type="checkbox"/>	<input type="text"/>	<	5.461E+16	<	4e+16
E3	EME_PLR01.02 (A)	energy [μJ]	<input checked="" type="checkbox"/>	<input type="text"/>	<	1.725E-47	<	
E4	EME_PLR01.02 (B)	energy [μJ]	<input checked="" type="checkbox"/>	<input type="text"/>	<	3.220E-57	<	
E5	EME_INJ.01	energy [μJ]	<input checked="" type="checkbox"/>	<input type="text"/>	<	1.01	<	
E6	EME_LH.01	energy [μJ]	<input checked="" type="checkbox"/>	<input type="text"/>	<	0.53	<	
L1	LSH_INJ.03	position	<input checked="" type="checkbox"/>	<input type="text"/>	<	OPENED	<	
L2	LSH_INJ.04	position	<input checked="" type="checkbox"/>	<input type="text"/>	<	OPENED	<	
L3	LSH_LH.01	position	<input checked="" type="checkbox"/>	<input type="text"/>	<	OPENED	<	
V1	VGP_INJ.01	pressure [mbar]	<input checked="" type="checkbox"/>	<input type="text"/>	<	#####	<	
V2	VGP_LH.01	pressure [mbar]	<input checked="" type="checkbox"/>	<input type="text"/>	<	#####	<	
M3	KOM_INJ.01	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	0.000293	<	
		y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	-0.000195	<	
M4	KOM_LH.01	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	2.912344	<	
		y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	2.195000	<	
M5	KOM_LH.02	x [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	3.304531	<	
		y [mm]	<input checked="" type="checkbox"/>	<input type="text"/>	<	1.523438	<	

Clean all

SAVE

RESTORE

A system to give immediate information about the state of the Machine

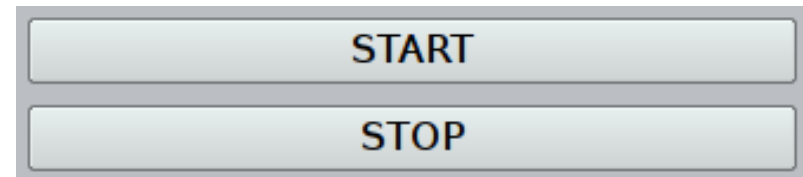


OK



We have a problem!

...and allow us to control it



courtesy F. Asnicar

matrix

Select section & sub-system

Functions	LASER	INJ	SPINJ	L00	LH	SPLH	L01	BC01	SPBC01	L02	L03	BC02	L04	TLS	DBD
Functions	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
MAGNETS	<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
DIAGNOSTICS	<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
VACUUM	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
RF	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
LASER	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

beam on SPLH

SPBC01

Select the scenario

beam on SPINJ

beam on BC01

GLOBAL

STATE

SYNOPTIC

show details

Ready to execute ScenariosBlock:beamAtSPINJBlock block

Reset

none

courtesy F. Asnicar

The screenshot shows the Matrix control interface with the following components and status:

- Select section & sub-system:** Functions, MAGNETS, DIAGNOSTICS, VACUUM, RF, LASER
- Top Bar:** Functions,  LASER,  INJ,  SPINJ,  L00,  LH,  SPLH,  L01,  BC01,  SPBC01,  L02,  L03,  BC02,  L04,  TLS,  DBD
- BLM:** blm\_gunlaser\_inj.01  ON
- BPM:**
  - rtbpm\_inj.01 No Link
  - rtbpm\_spinj.01 No Link
- CCD:**
  - yscrccd\_inj.01 OFF
  - yscrccd\_inj.02 OFF
  - yscrccd\_inj.03 OFF
  - yscrccd\_spinj.01 OFF
- MULTISCREEN:**
  - mscr\_inj.01  ON OFF
  - mscr\_inj.02  ON OFF
  - mscr\_inj.03  ON OFF
  - scr\_spinj.01  ON OFF
- View:** GLOBAL, STATE, SYNOPTIC
- show details

courtesy F. Asnicar

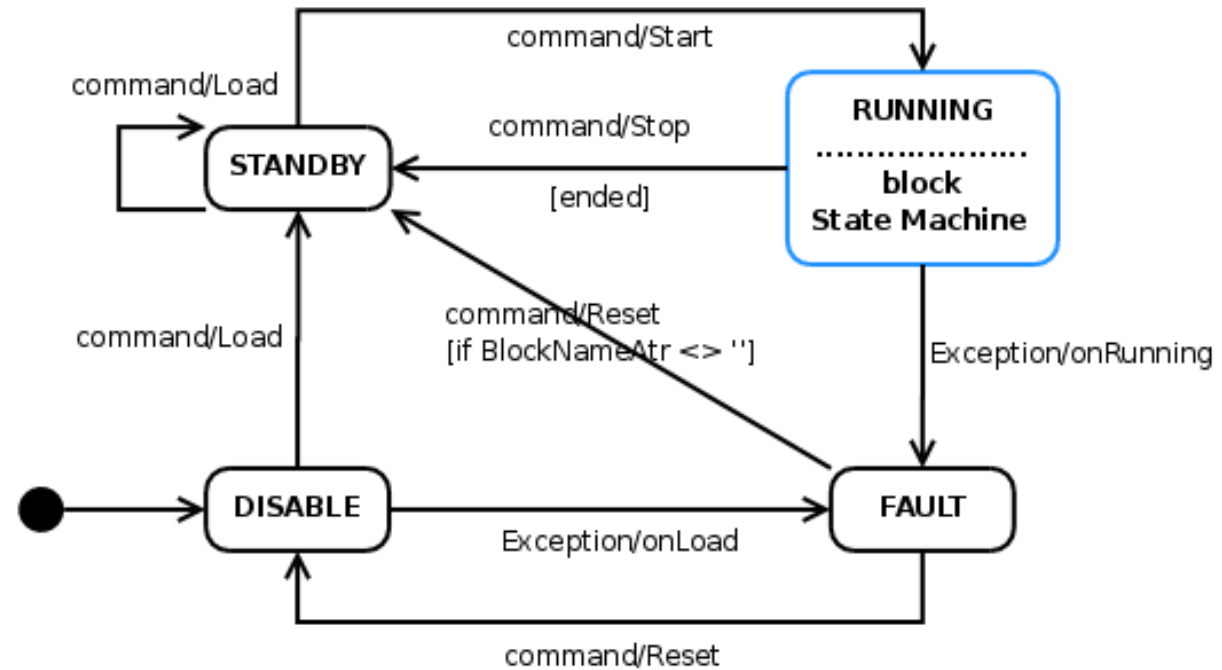
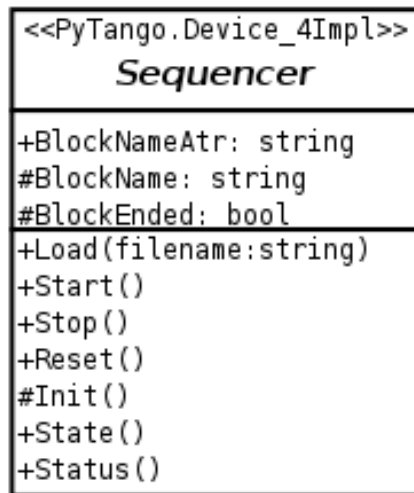
The screenshot displays the Matrix control interface for the FERMI synchrotron. The main window shows a detailed layout of the accelerator components, including injection lines, bending magnets, and various diagnostic and vacuum systems. The interface is organized into several sections:

- Top Panel:** A row of buttons for selecting different sections and sub-systems: Functions, LASER, INJ, SPINJ, L00, LH, SPLH, L01, BC01, SPBC01, L02, L03, BC02, L04, TLS, and DBD.
- Left Panel:** A vertical menu with buttons for selecting sections: Functions, MAGNETS, DIAGNOSTICS, VACUUM, RF, and LASER.
- Central Area:** A large diagram of the synchrotron layout. Components are color-coded and labeled. Labels include 'WM' (Wigglers), 'FC' (Focusing Chicanes), and several boxes containing '#####' representing diagnostic or vacuum points.
- Bottom Panel:** A control panel with various options and a legend.
  - View:** Buttons for GLOBAL, STATE, and SYNOPTIC views, along with a 'show details' checkbox.
  - Diagnostics:** A section with checkboxes for 'magnets' and 'diagnostics'. Under 'diagnostics', there are checkboxes for 'SCR', 'BPM', and 'BLM', each with a corresponding icon.
  - Vacuum:** A section with a checkbox for 'vacuum' and a 'VLV' icon.
  - Legend:** A table of status indicators:
 

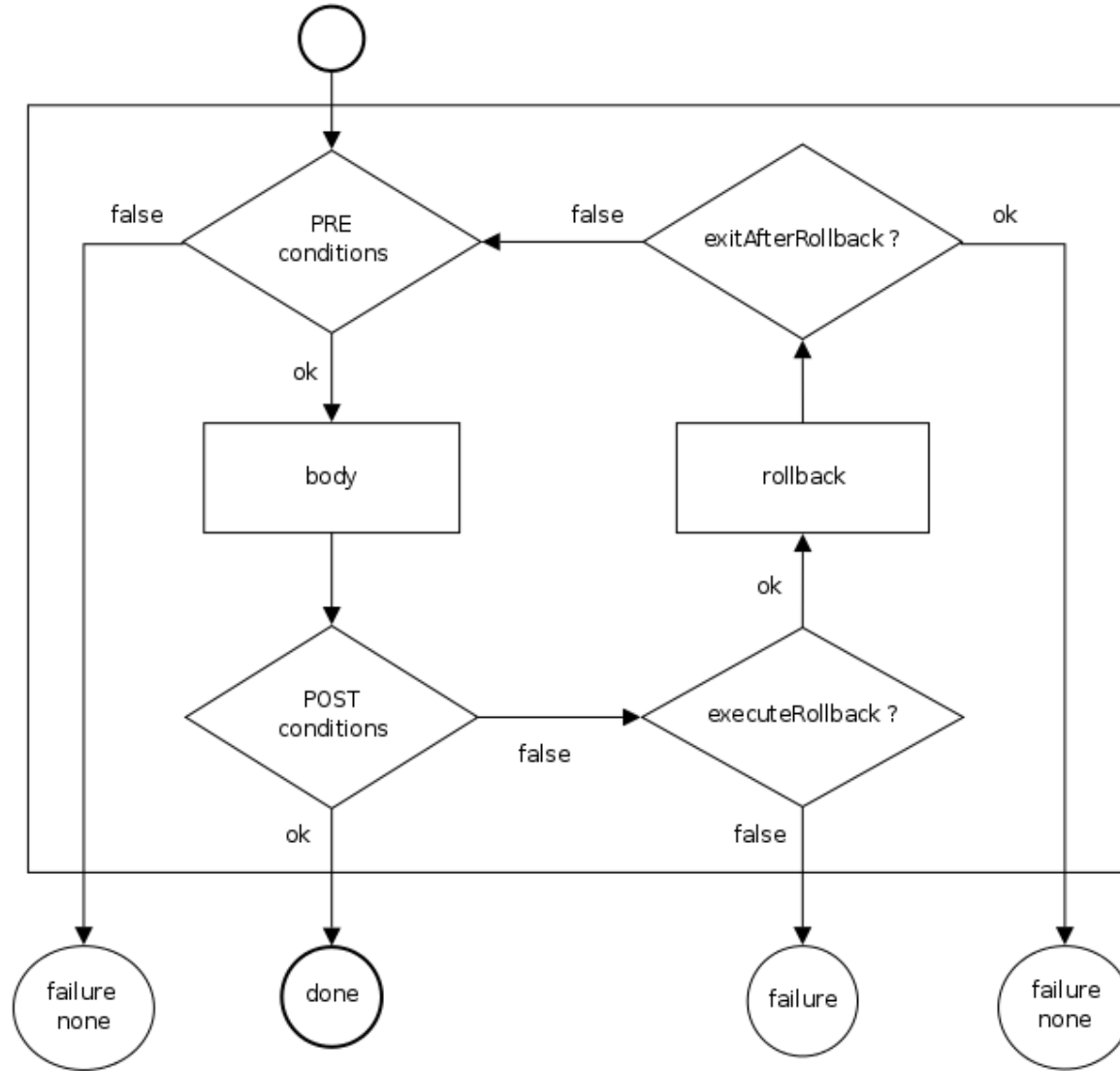
OFF	ON	STANDBY	INIT	ALARM
DISABLED	RUNNING	FAULT	MOVING	CLOSED
JNKNOW	OPENED			

courtesy F. Asnicar

Generic Tango device. Can *Load* a **block** of code, *Start* it and monitor the execution *State*



courtesy F. Asnicar

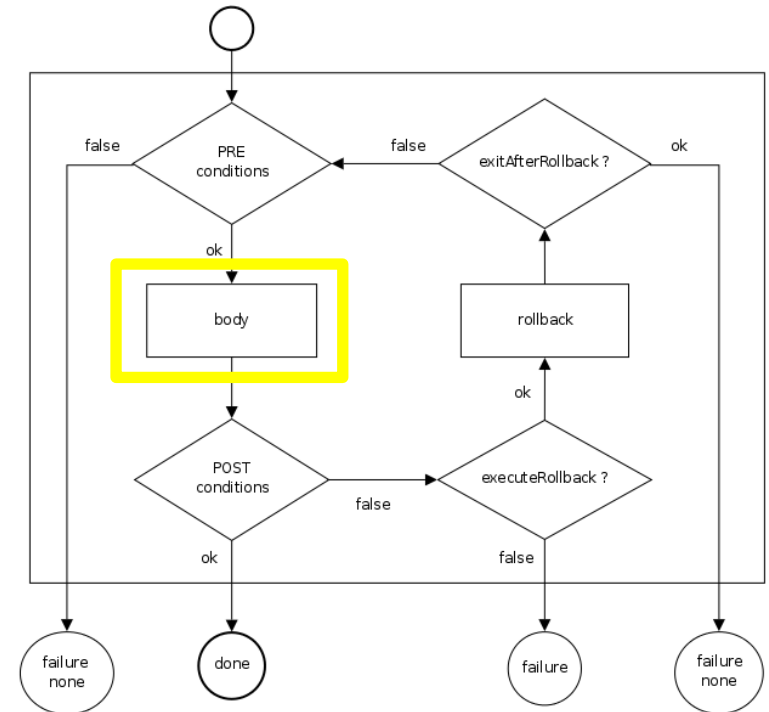


courtesy F. Asnicar

Specialization of block (for Matrix)  
 endless loop reads State of a list of devices and  
 calculates block State with inference rule:

okState {s ∈ Tango.State : s = desired state}

	If	∀s: s ∈ okState	⇒	<b>RUNNING</b>
else	If	∃ <b>FAULT</b>	⇒	<b>FAULT</b>
else	If	∃ <b>ALARM</b>	⇒	<b>ALARM</b>
else	If	∃ <b>STANDBY</b>	⇒	<b>STANDBY</b>
else	If	∃ <b>INIT</b>	⇒	<b>INIT</b>
else	If	∃ <b>UNKNOWN</b>	⇒	<b>UNKNOWN</b>
else			⇒	<b>FAULT</b>



courtesy F. Asnicar

Giacomo Strangolino  
*Elettra – Sincrotrone Trieste*

# QTango version 4

**A multi threaded framework to  
develop *Tango* applications**

**mailto: [giacomo.strangolino@elettra.trieste.it](mailto:giacomo.strangolino@elettra.trieste.it)**

# Core News

- **Reconnection** to the device at startup;
- *asynchronous* execution of *targets* (i.e. In the *DeviceThread*);
- introduction of a class called **ProxyInterface**. It provides a common interface to readers and writers, aimed at unifying the representation of some of their properties;
- getting **attribute properties** becomes easy: QTangoComProxyReader and Writer now have a method named `setDesiredAttributePropertyList()`;
- QTangoCore 4 is **more efficient**: the startup procedure of QTango applications is now much faster than before.

## Widgets News

**Font Scaling widgets**, for objects representing strings (labels, spin boxes, line edits...);

**drag and drop** support of tango points and widgets among different applications;

development of **new** widgets becomes **easier** and **faster**, thanks to the **QTangoWidgetCommon** interface

# QTangoWidgetCommon interface

- **view trend** functionality;
- source **copy** and **paste** capabilities;
- **show** and **hide** events management;
- tango point **information** and **meta information**;
- **drag** and **drop** capabilities, to activate tango connection by dragging a source from a widget into another;
- helper application functionality;
- E-Giga hdb integration through desktop services.

## Widgets news: TreaderWriter and Plots

- supports custom writer types (spin box, TLineEdit, TNumeric, combo box), alongside automatic configuration;
- zoom support on click event;
- custom writer position is possible, alongside automatic placement

**TrendPlot** renewed;

*E/TPlotLightMarker* renewed and highly configurable through a dialog;

**TRealtimePlot**, tailored for real time Tango commands;

## Widgets news: reading and writing *Spectrum* attributes

- **TSpectrumButton**: writes into a *spectrum* attribute fetching data from *SimpleDataProxy* widgets;
- **TwidgetGroup**: groups a set of readers and refreshes them with the values extracted from a *spectrum* attribute;
- Full *Qt4 designer* integration and configuration!

## Widgets news: TInfoTextBrowser

- A *TextBrowser* displaying tango point information, descriptions and connection status related to a source (*Ctrl+click* on a widget);
- future integration with a DBUS desktop-distributed application help system?

Giacomo Strangolino  
*Elettra – Sincrotrone Trieste*

# Mango

**an online designer to easily create QTango graphical  
interfaces**

**[mailto: giacomo.strangolino@elettra.trieste.it](mailto:giacomo.strangolino@elettra.trieste.it)**

# What is Mango

- Mango is a simple tool to generate online QTango panels, that are ready to run without need to be compiled;
- interfaces are made up of the simple QTango and Qt widgets;
- all the logic of the components and their interaction with each other reflects the only logic of the widgets themselves.

## What Mango *is not*

- it *is not* a tool to design complex control panels, with any other logic than the simple one offered by each widget through the exported Qt properties and the signal/slot connections;
- **Nevertheless**: future improvements of the interface and the introduction of complex logic can be started from the xml generated with Mango by importing it into a *Qt4 designer* project.

## Whom Mango is addressed to

- The GUI developer who wants to fastly create an interface to a tango device server whose commands and attributes fully support the QTango widgets, without needing any additional logic;
- the **Tango device programmer**, to design a *Tango server* fully compatible with the *QTango* library and immediately test his work in progress. He might even outline a GUI choosing the most suitable widgets;

## Whom Mango is addressed to (II)

- the **end user** of the graphical interface, who can outline a draw of the desired panel;
- the control room **operator**. He might draw a panel *on the fly*, to create an outline GUI gathering QTango objects from other applications, simply dragging a widget from an application and dropping it into any mango container.

## Whom Mango is addressed to (III)

- the library tester;
- the **hardware referent**, who commissions the desired control panel to the GUI developer. He may draw a draft of the application containing the relevant controls and widgets and discuss every aspect with the application programmer.

## Whom Mango is addressed to (IV)

- the **end user** of the graphical interface, who can outline a draw of the desired panel;
- the control room **operator**. He might draw a panel *on the fly*, to create an outline GUI gathering QTango objects from other applications, simply dragging a widget from an application and dropping it into any mango container.

## Features and usage

- Drag and drop widgets from an object factory;
- property editor to change the Qt properties;
- profile manager to save and load the projects;
- *Tango* point information text browser and *Tango* database browser integrated;
- signal/slot connections;
- Import from/export to *xml* files.
- In *execution mode*, the application is *equivalent* to any other Qt compiled executable.

## TODO

- Undo/redo framework;
- message dialogs to prevent the application to be closed while projects are unsaved;
- bug corrections (it's still *beta*)
- ...

# Design phase

The screenshot displays the Mango GUI design tool interface. The central workspace contains a GUI design for a device named "Pluteross". At the top, a label displays "74.8 [Pluteross]" next to a control panel with five buttons and an "APPLY" button. Below this is a tabbed interface with "Tab 1" selected, containing a widget labeled "test/device/1/double\_scalar" which features a circular gauge. A tooltip over the gauge reads "Last read (P): Wed Dec 30 12:39:13 2009". At the bottom of the workspace is a widget labeled "test/device/1/long\_spectrum\_ro" containing a line plot of a signal over time.

**Object Factory**

Object	Description
Label	A TLabel displayin...
Simple La...	A simple label dis...
Circular G...	A circular gauge ...
Led	A led to display a ...
PlotLight...	A plot to display a ...
RealtimeP...	A plot specialized ...
Table	A table to display ...
Push Butt...	A Push Button to ...
Apply Nu...	A widget to set a ...
Check Box	A check box to se...
Reader W...	A reader that, on ...
Spin Box	A spin box to set ...
Double S...	A spin box to set ...
Spectrum...	A push button to ...
Pixmap	A pixmap change...
Containers	Container widget...

**Tango DB Browser**

```

hokuto:20000
+ 1
+ a1
+ aa
+ alarm
+ archiving
+ arrays
+ b
+ barbera
+ beefix
+ blm
+ blmmaster
+ booster
+ branzino
+ class_attr
+ coccobill
+ csl
+ CStest
+ data
+ dbpmtemp
+ ddd
+ dev
+ devmapper
+ eewcal
+ eventester
+ f
+ fas
  
```

**Tango Point Information**

**What's this?**

Un doppio scalare del device primo(Value min: -248 max: 295)(Warning low: -90 high: 160)(Error low: -200 high: 235)

**test/device/1/double\_scalar**

*Un doppio scalare del device primo*

**Connection status:**

- *connected: last read on Wed*

**Object Properties**

Property

- PlotLightMarker
- TPlotLightMarker
  - source
  - sourcesList
  - period
  - helperApplicationEnabled
  - helperApplication
  - forceWidgetDisabled
  - enterLeaveEventsEnabled
  - enterEventDelay
  - copyActionEnabled
  - pasteActionEnabled
  - dropEnabled
  - hdbActionEnabled
  - fillFromHistoryEnabled
  - boundsFromAutoConfiguratio
  - colouredBackground
  - additionalInfo
- EPlotLightMarker

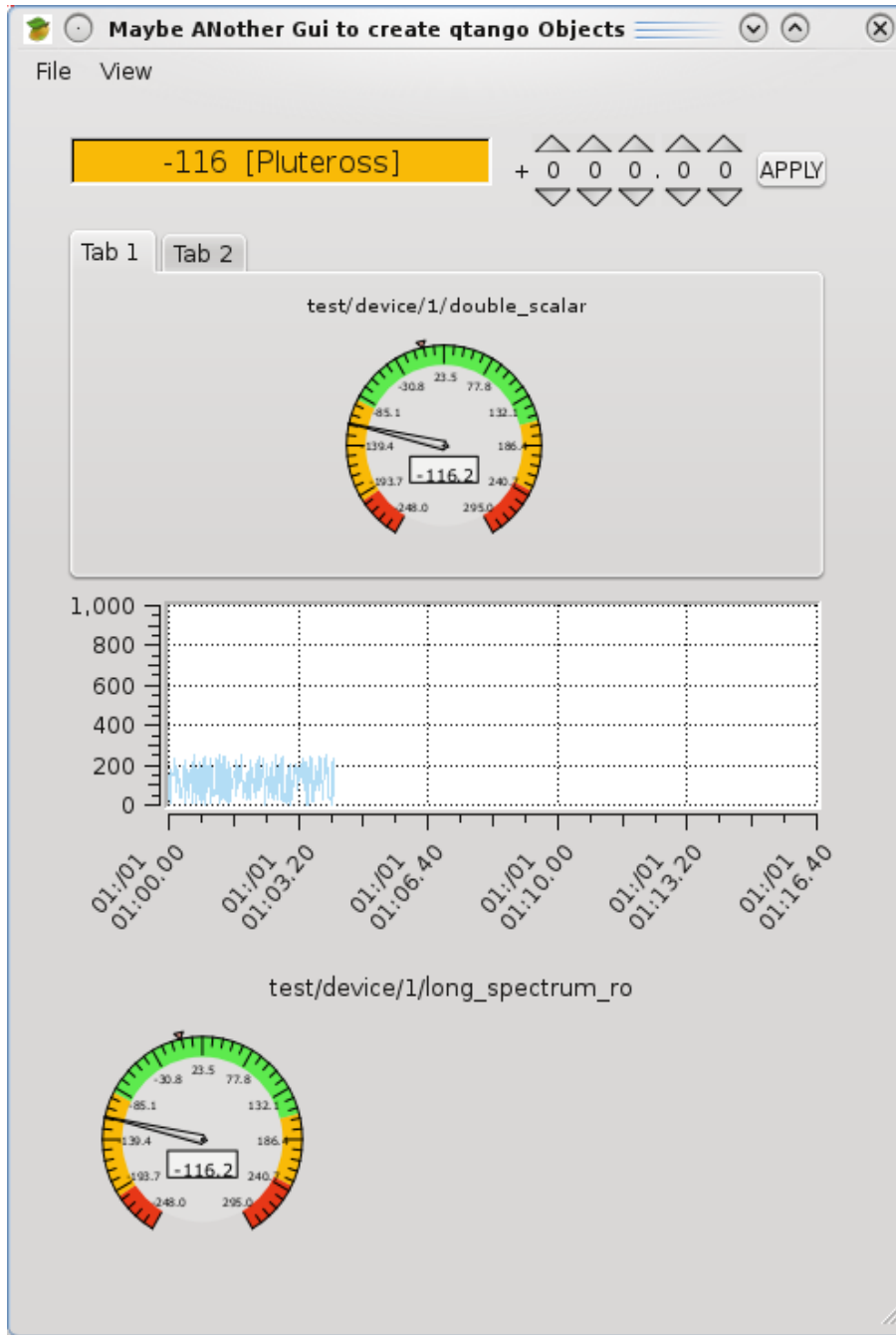
**Design Options**

Main Window Grid density: 20

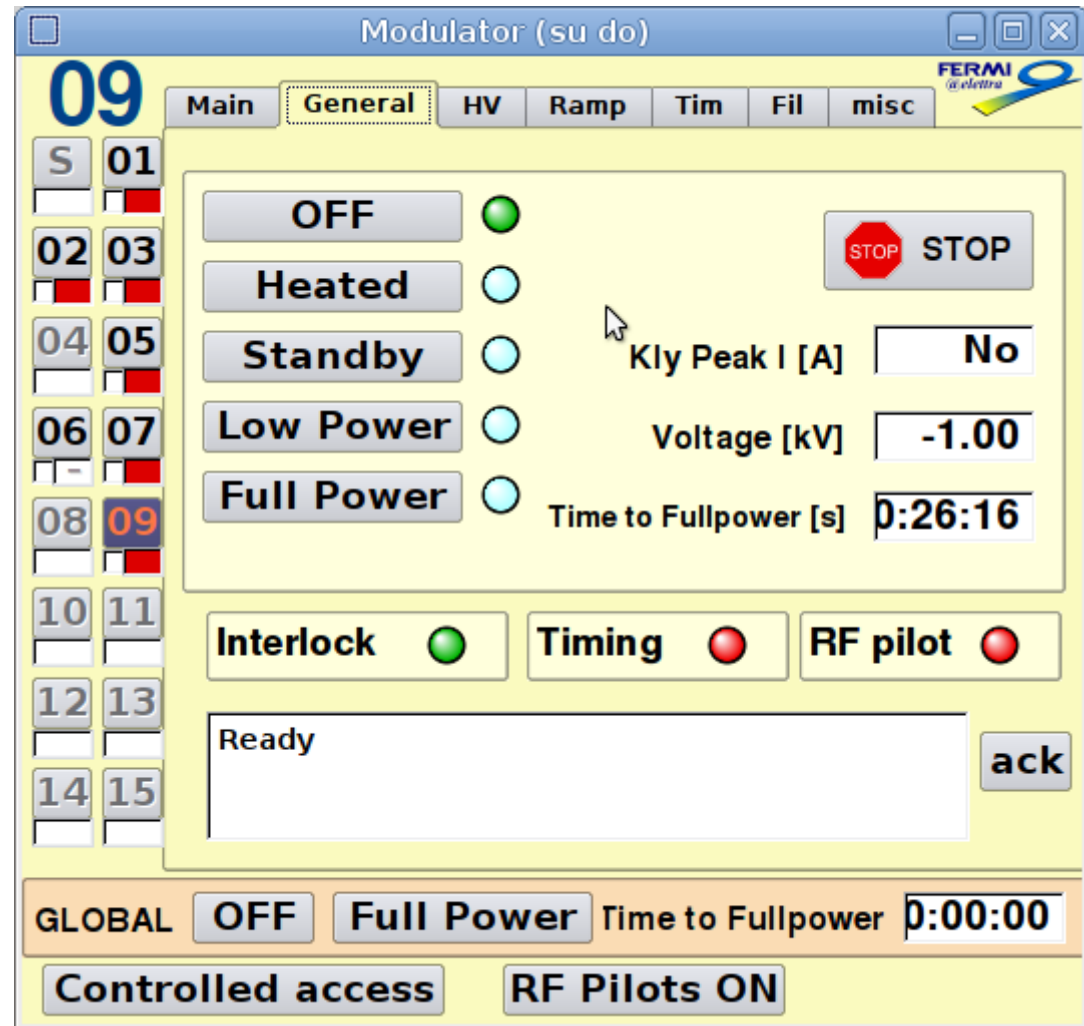
Ignore main window resizes

Delete selected Widget

# Run phase



Main panel for  
klystron modulator control



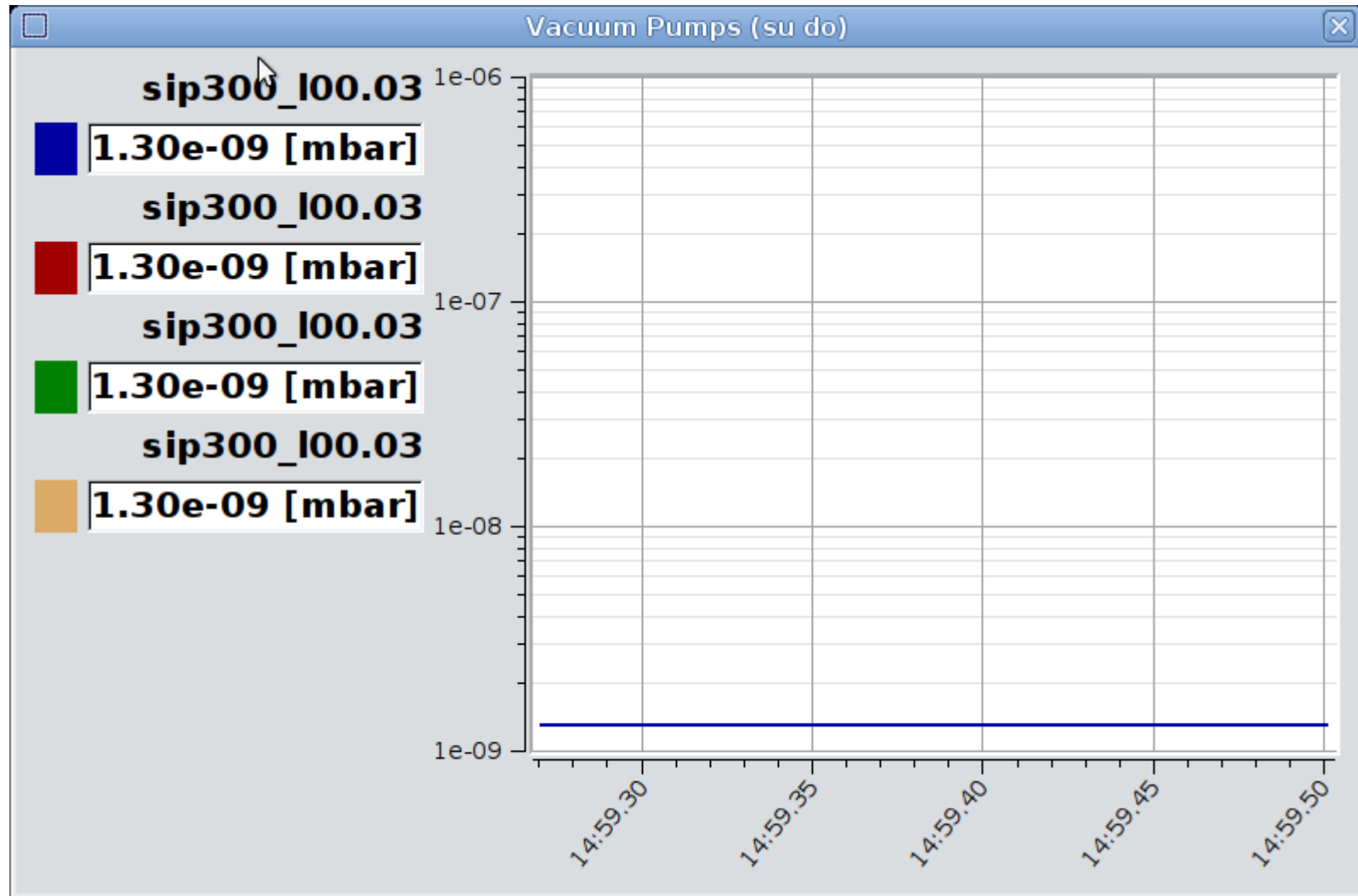
The screenshot shows a software interface titled "Modulator (su do)". It features a sidebar with 15 numbered buttons (01-15), with button 09 highlighted. The main panel has several tabs: "Main", "General" (selected), "HV", "Ramp", "Tim", "Fil", and "misc".

Under the "General" tab, there are five power mode buttons: "OFF", "Heated", "Standby", "Low Power", and "Full Power", each with a corresponding status indicator (green or cyan circle). A red "STOP" button is also present. To the right, there are three input fields: "Kly Peak I [A]" with the value "No", "Voltage [kV]" with the value "-1.00", and "Time to Fullpower [s]" with the value "0:26:16".

Below these are three status indicators: "Interlock" (green circle), "Timing" (red circle), and "RF pilot" (red circle). A text box displays "Ready" with an "ack" button to its right.

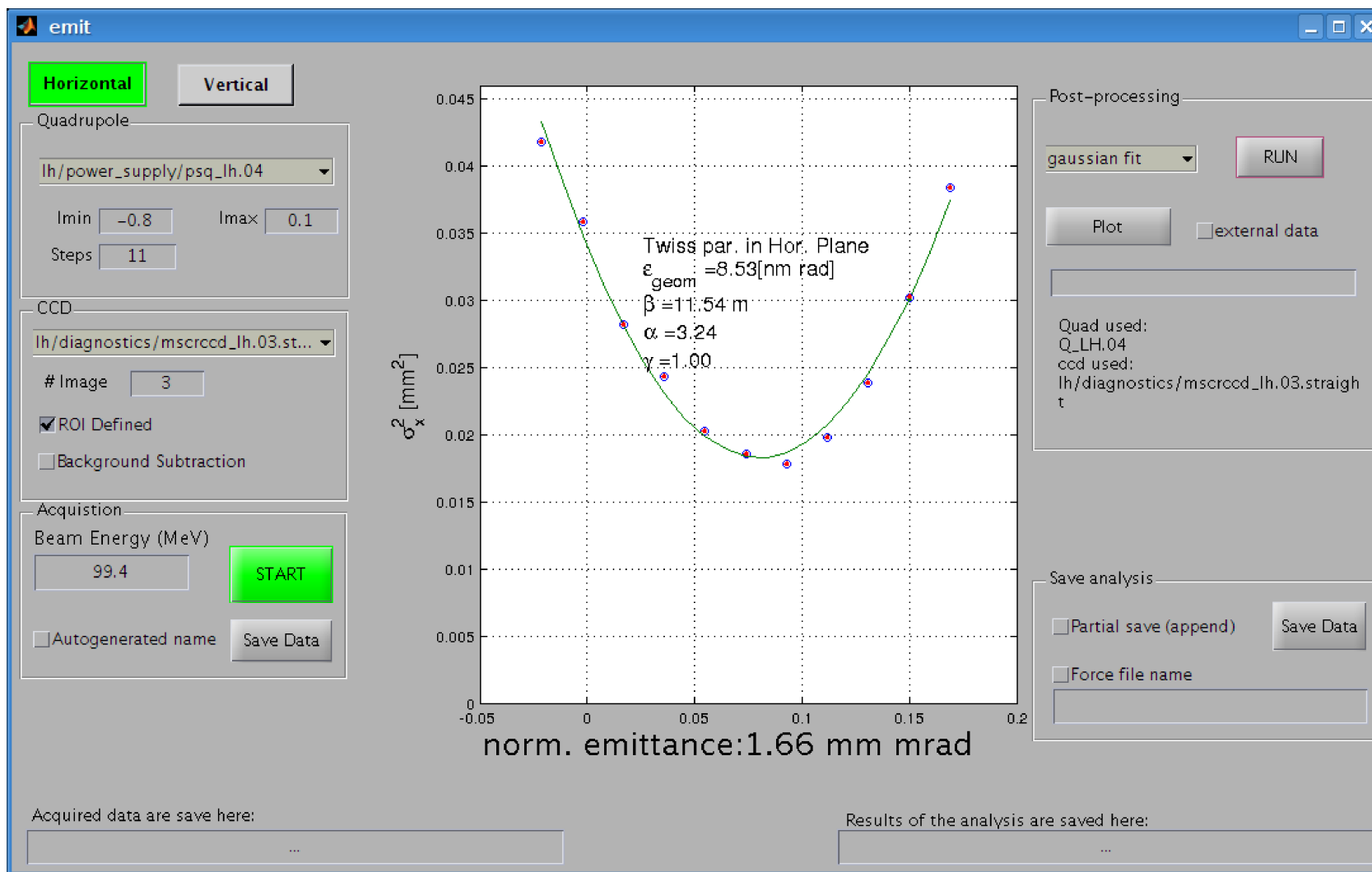
At the bottom, there is a "GLOBAL" section with "OFF" and "Full Power" buttons, and a "Time to Fullpower" field showing "0:00:00". Below this are two buttons: "Controlled access" and "RF Pilots ON".

courtesy L. Zambon



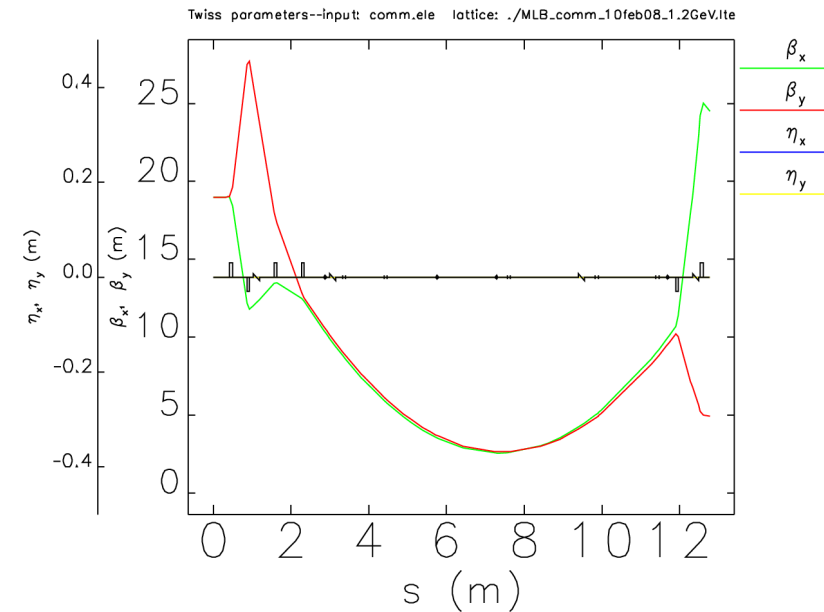
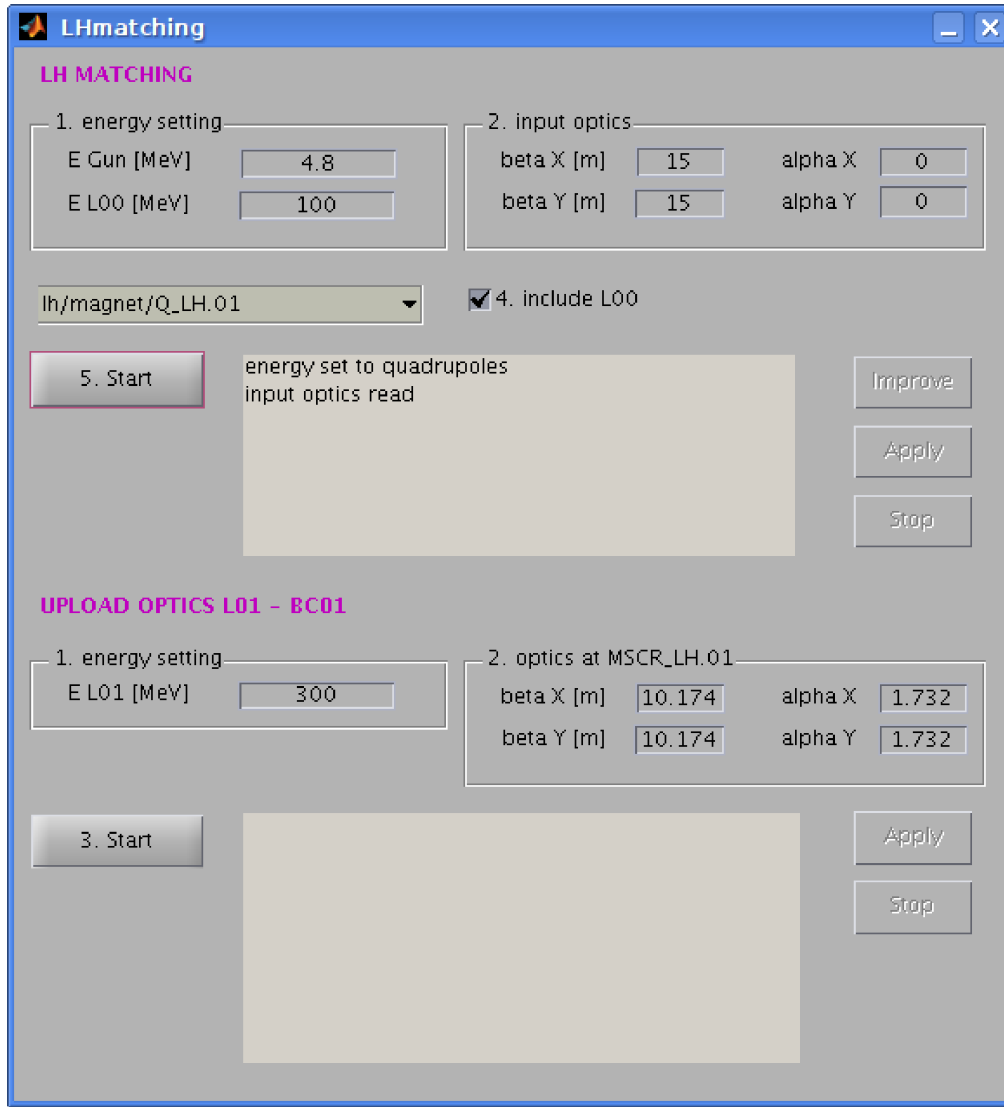
Main panel for RF plant vacuum monitoring

courtesy L. Zambon



Emittance measurement panel

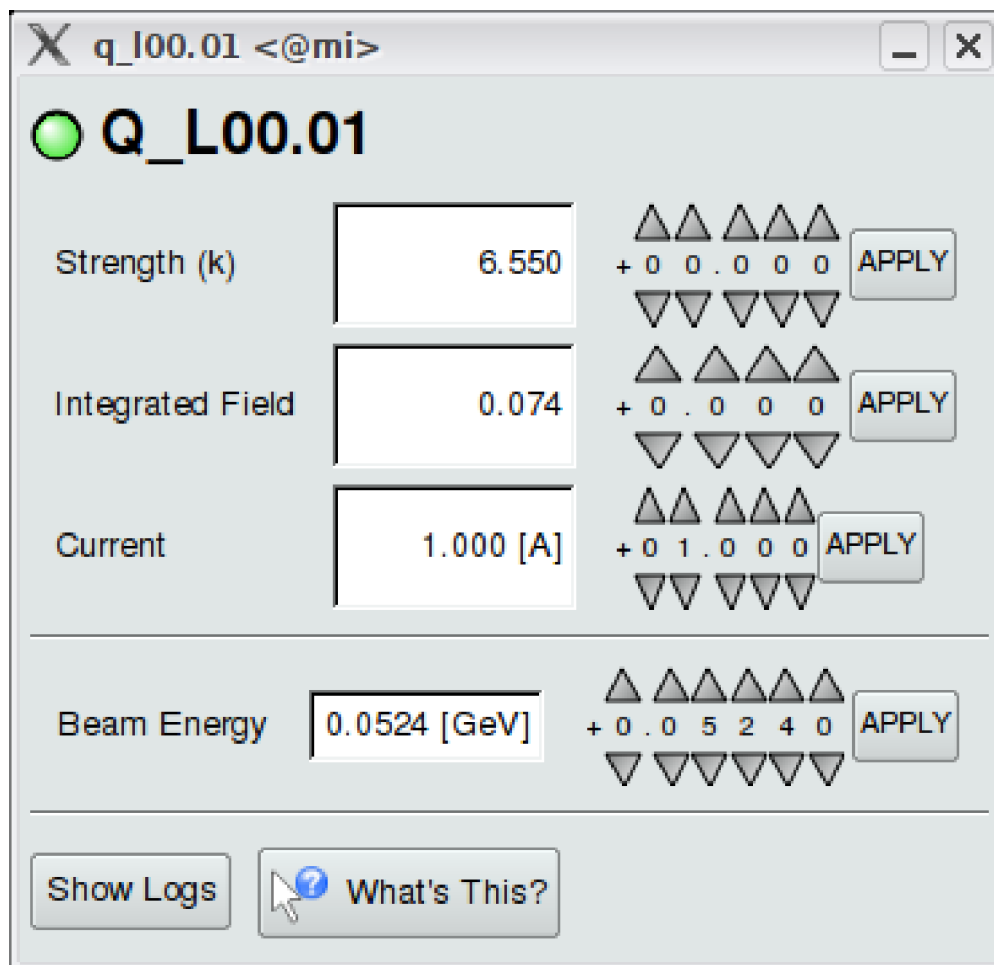
courtesy G. Penco



- Matching: impose desired optics
- start from measurements
  - run elegant to find & match optics
  - set magnet strengths

courtesy S. Di Mitri

- Use Matlab program for bookkeeping and coordinations
- Calculations done by elegant tracking code
- SDDS format data files to pass data from machine to elegant and back
  - PyTango programs do this + database form mapping names
- Tango “magnet” device converts from Amps to magnetic strenght



# That' s all!

Thank you for you attention.