

“jddd” a Java DOOCS Data Display for the XFEL

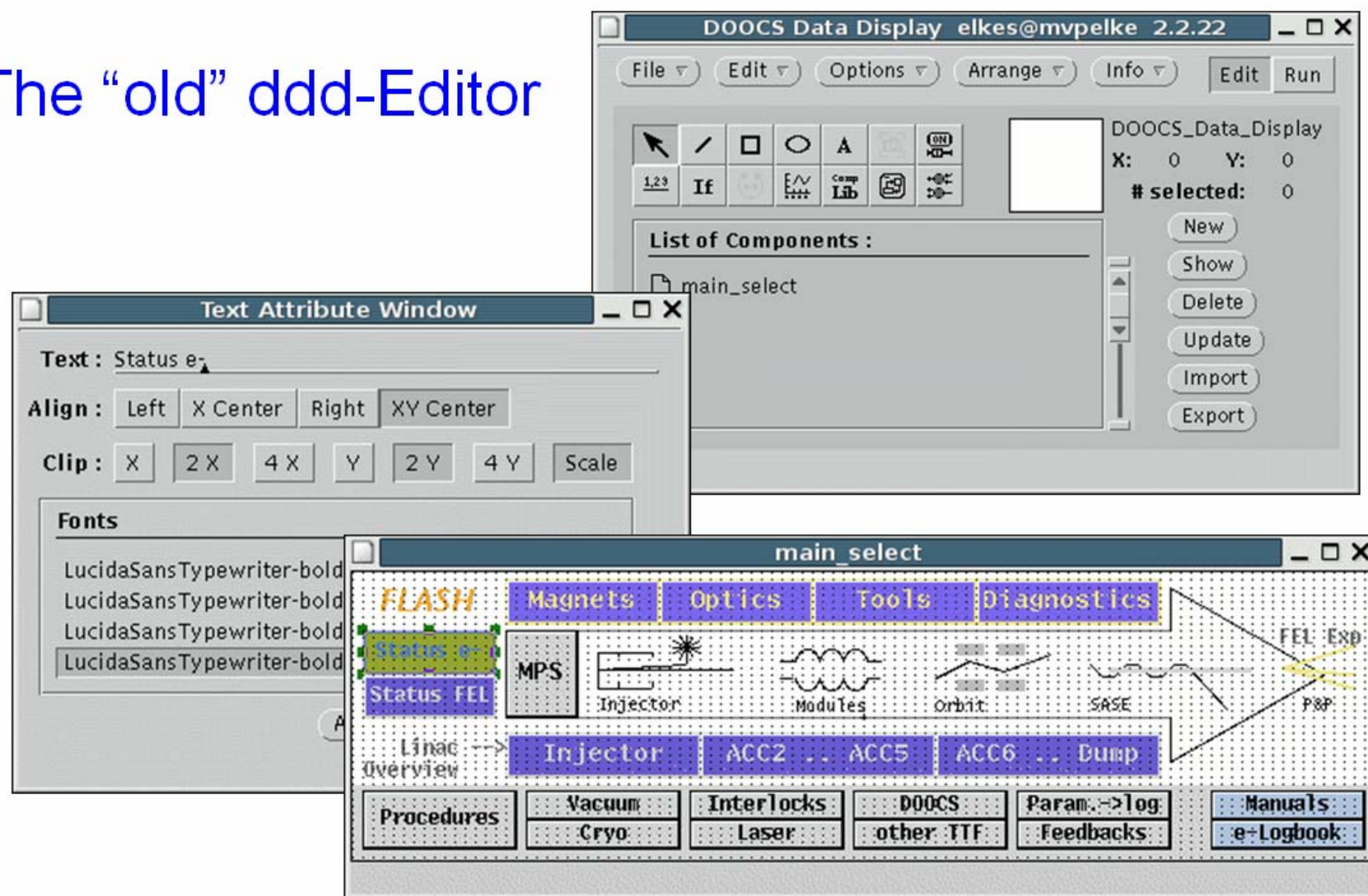
Editor for a Graphical User Interface of DOOCS & TINE
and Runtime Engine

Elke Sombrowski

- Motivation
- The “old” ddd (DOOCS Data Display)
- Considerations about what we need in future
- The new jddd (Java DOOCS Data Display)
 - The jddd editor
 - Reusable components in jddd
 - Special features of jddd
 - How to start jddd
- Experience & Outlook

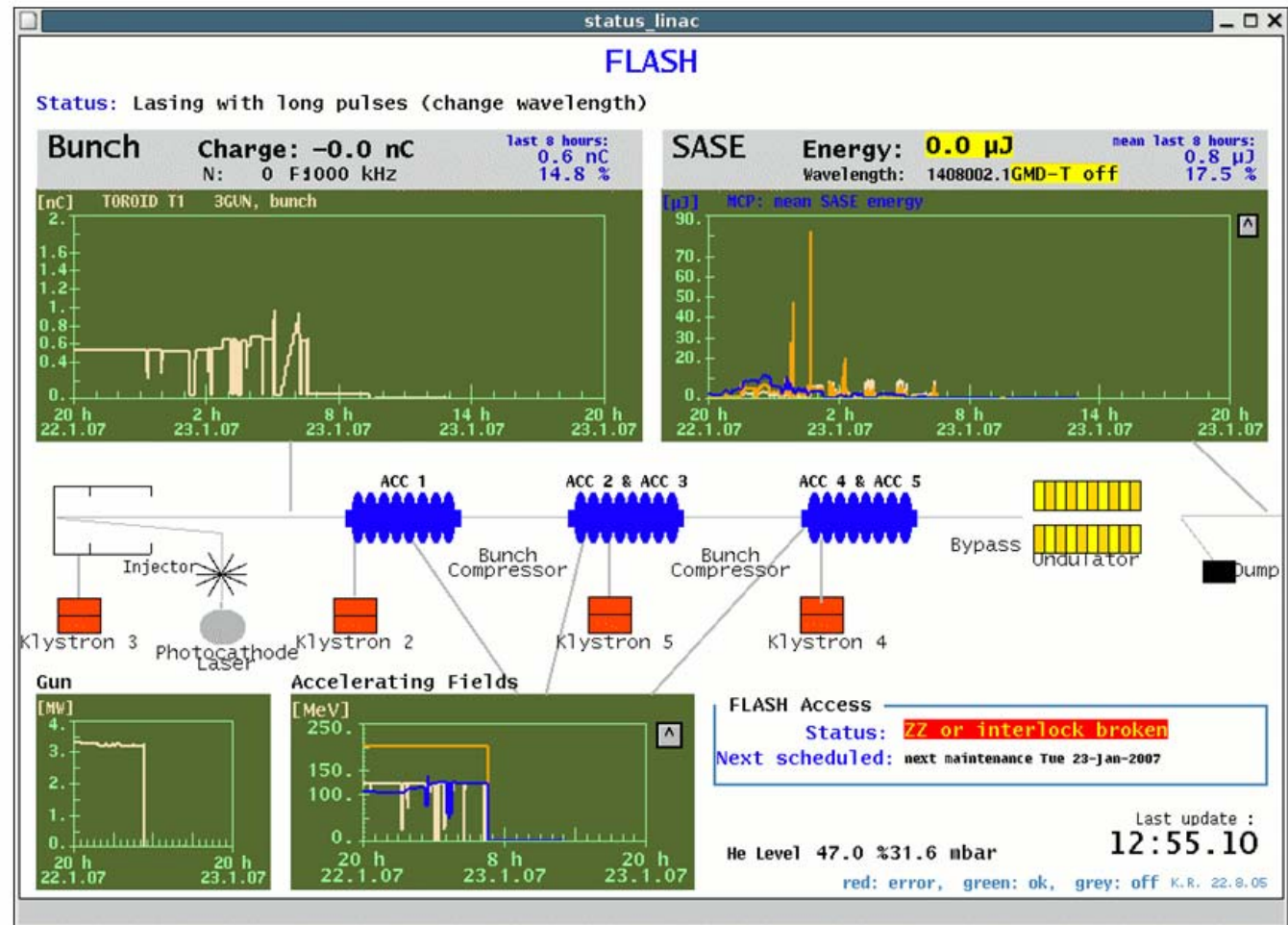
- **We don't want to design all panels in the controls group**
 - Non-programmers / subsystem experts should develop graphical controls panels
- **We already have ddd (C++) since 10 years**
 - synapical displays editor
 - but ddd needs an “update”

The “old” ddd-Editor



ddd Screenshot

Flash: ~ 1300 control panels



UNIX / C++ version:

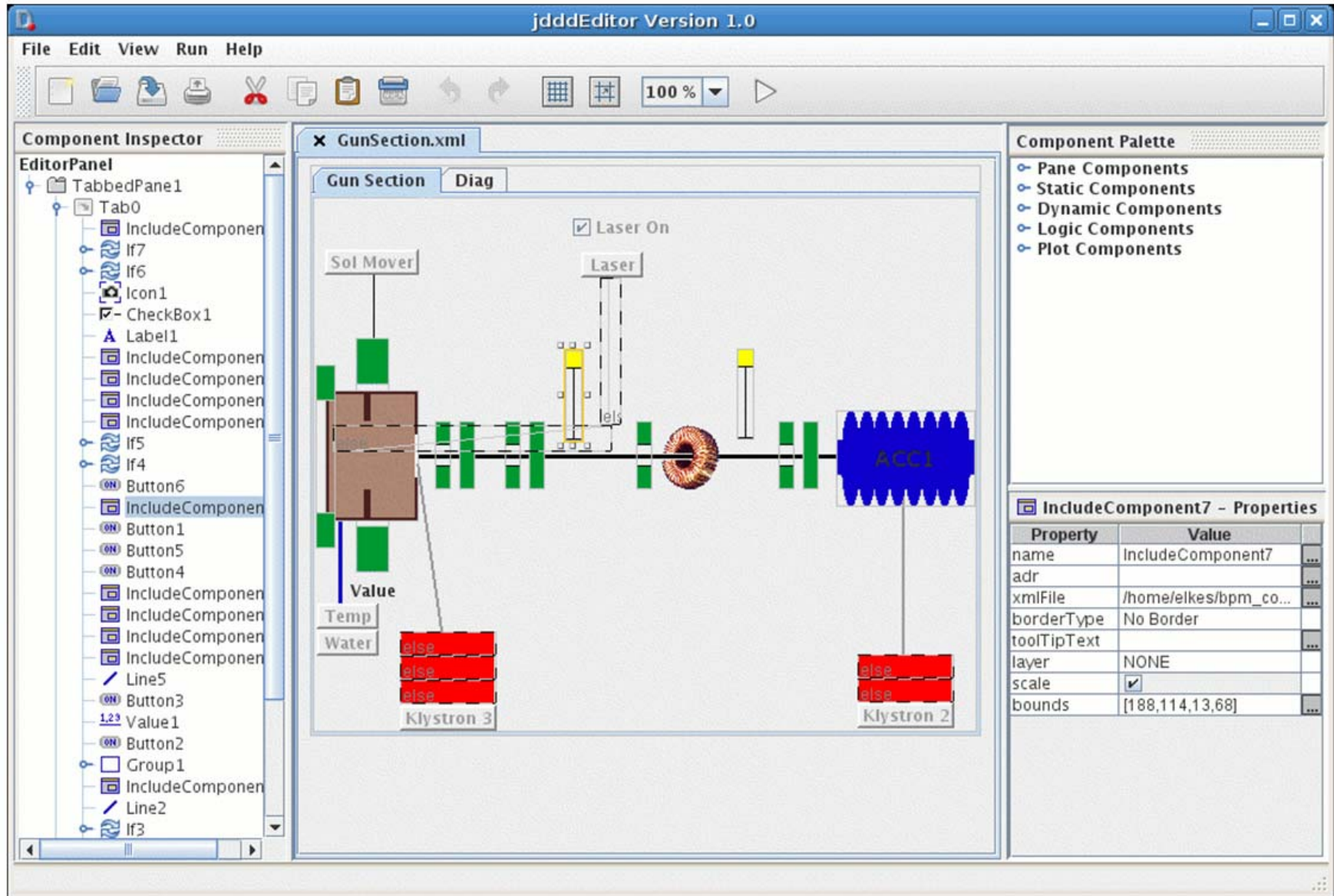
- + Editor: simple creation of control panels with a set of standard components: text, buttons, graphical components, values, dials, plots, "If", animated components
- + Control panels are saved in text-files and parsed at runtime
- Some new editor features are difficult to implement (e.g. undo functionality)
- Open look library is no longer supported

What we need in Future:

- Keep the good things of ddd: simple Editor
- Editor with modern look&feel and functionality
- Improved components (plots with math functions, ...)
- New components (TabbedPane, ...)
- Platform independence
- More flexible usage:
 - Reuse of displays in other applications
 - Web interface (Applet)
- Convertibility of the old text-files to a new format

jddd implementation:

- Use JAVA
 - Standard JAVA technology (Swing, Java Beans)
 - Save panels in standard xml file format
 - Options for jddd editor:
 - use Eclipse or Netbeans and write Plugins
 - develop a GUI Editor based on the Netbeans visual library
- **we decided to write a completely new editor, because we want to be independent of external libraries and we want to have the highest flexibility**



jdddEditor Version 1.0

File Edit View Run Help

100 %

Component Inspector

EditorPanel

- TabbedPane1
 - Tab0
 - IncludeComponent
 - If7
 - If6
 - Icon1
 - CheckBox1
 - Label1
 - IncludeComponent
 - IncludeComponent
 - IncludeComponent
 - IncludeComponent
 - If5
 - If4
 - Button6
 - IncludeComponent
 - Button1
 - Button5
 - Button4
 - IncludeComponent
 - IncludeComponent
 - IncludeComponent
 - IncludeComponent
 - Line5
 - Button3
 - Value1
 - Button2
 - Group1
 - IncludeComponent
 - Line2
 - If3

GunSection.xml

Gun Section Diag

☑ Laser On

Sol Mover

Laser

Value

Temp

Water

else

else

else

Klystron 3

else

else

else

Klystron 2

ACC1

Component Palette

- Pane Components
- Static Components
- Dynamic Components
- Logic Components
- Plot Components

IncludeComponent7 - Properties

Property	Value
name	IncludeComponent7
adr	
xmlFile	/home/elkes/bpm_co...
borderType	No Border
toolTipText	
layer	NONE
scale	☒
bounds	[188,114,13,68]

Editor Window

jdddEditor Version 1.0

File Edit View Run Help

100 %

Component Inspector

EditorPanel

- TabbedPane1
 - Tab0
 - IncludeComponent7
 - If7
 - If6
 - Icon1
 - CheckBox1
 - Label1
 - IncludeComponent7
 - IncludeComponent7
 - IncludeComponent7
 - IncludeComponent7
 - If5
 - If4
 - Button6
 - IncludeComponent7
 - Button1
 - Button5
 - Button4
 - IncludeComponent7
 - IncludeComponent7
 - IncludeComponent7
 - IncludeComponent7
 - Line5
 - Button3
 - Button1
 - Button2
 - IncludeComponent7
 - Line2
 - If3

GunSection.xml

Gun Section Diag

Value

Temp

Water

Laser On

Laser

Klystron 3

Klystron 2

ACC1

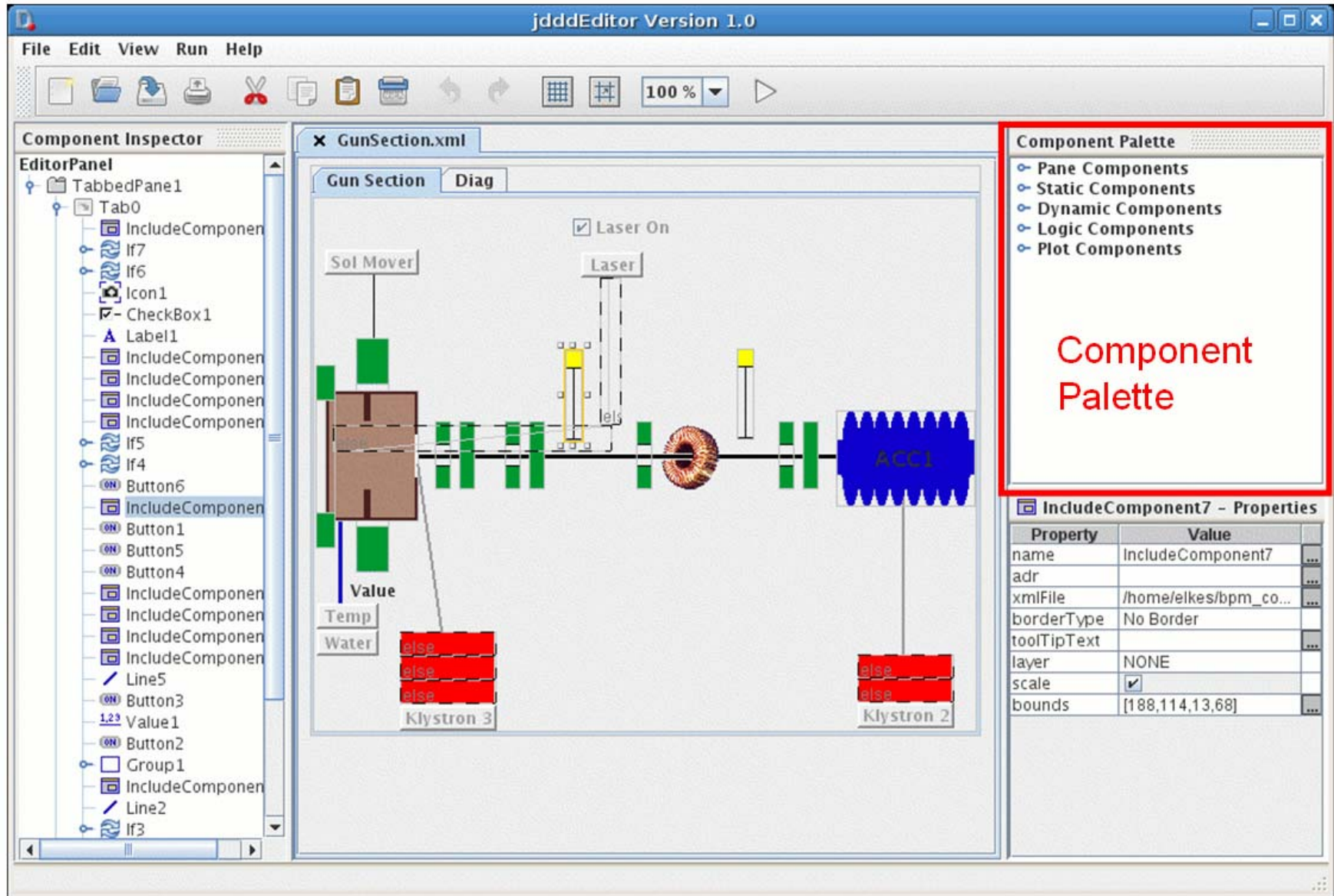
Component Palette

- Pane Components
- Static Components
- Dynamic Components
- Logic Components
- Plot Components

IncludeComponent7 - Properties

Property	Value
name	IncludeComponent7
adr	
xmlFile	/home/elkes/bpm_co...
borderType	No Border
toolTipText	
layer	NONE
scale	☒
bounds	[188,114,13,68]

Component Inspector



The screenshot displays the jdddEditor Version 1.0 interface. The main workspace shows a schematic diagram of a 'Gun Section' with components like 'Sol Mover', 'Laser', 'Klystron 3', 'Klystron 2', and 'ACC1'. The interface includes a 'Component Inspector' on the left, a 'Component Palette' on the right (highlighted with a red box), and a 'Properties' panel at the bottom right.

Component Palette

- Pane Components
- Static Components
- Dynamic Components
- Logic Components
- Plot Components

IncludeComponent7 - Properties

Property	Value
name	IncludeComponent7
adr	
xmlFile	/home/elkes/bpm_co...
borderType	No Border
toolTipText	
layer	NONE
scale	☒
bounds	[188,114,13,68]

jdddEditor Version 1.0

File Edit View Run Help

100 %

Component Inspector

EditorPanel

- TabbedPane1
 - Tab0
 - IncludeComponent
 - If7
 - If6
 - Icon1
 - CheckBox1
 - Label1
 - IncludeComponent
 - IncludeComponent
 - IncludeComponent
 - IncludeComponent
 - If5
 - If4
 - Button6
 - IncludeComponent
 - Button1
 - Button5
 - Button4
 - IncludeComponent
 - IncludeComponent
 - IncludeComponent
 - IncludeComponent
 - Line5
 - Button3
 - Value1
 - Button2
 - Group1
 - IncludeComponent
 - Line2
 - If3

GunSection.xml

Gun Section Diag

☑ Laser On

Sol Mover

Laser

Value

Temp

Water

else

else

else

Klystron 3

else

else

else

Klystron 2

ACC1

Component Palette

- Pane Components
- Static Components
- Dynamic Components
- Logic Components
- Plot Components

IncludeComponent7 - Properties

Property	Value
name	IncludeComponent7
adr	
xmlFile	/home/elkes/bpm_co...
borderType	No Border
toolTipText	
layer	NONE
scale	☒
bounds	[188,114,13,68]

Component Properties

Pane Components:

- **TabbedPane**
- LayeredPane
- IncludeComponent
- **DynamicList**

Static Components:

- Label
- Line
- Oval
- Rectangle
- Triangle
- Icon

Dynamic Components:

- Button
- Value
- Dial
- **CheckBox**
- **ProgressBar**
- **Slider**
- StatusRegister
- **Audio**
- **ColouredIndicator**
- **LocationChooser**
- CameraImage
- **TextArea**

Logic Components:

- If
- **Switch**

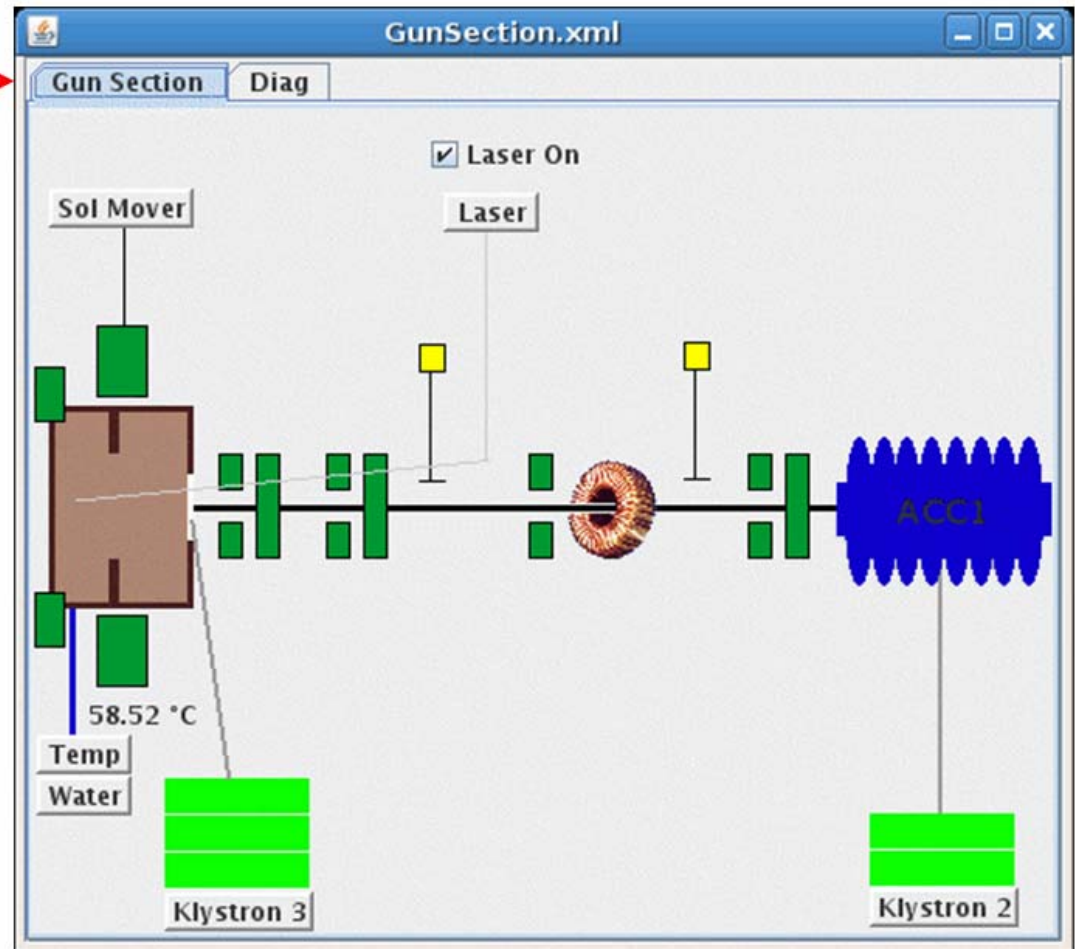
Plot Component:

- PlotSpectrum

New Components

Pane Components

TabbedPane

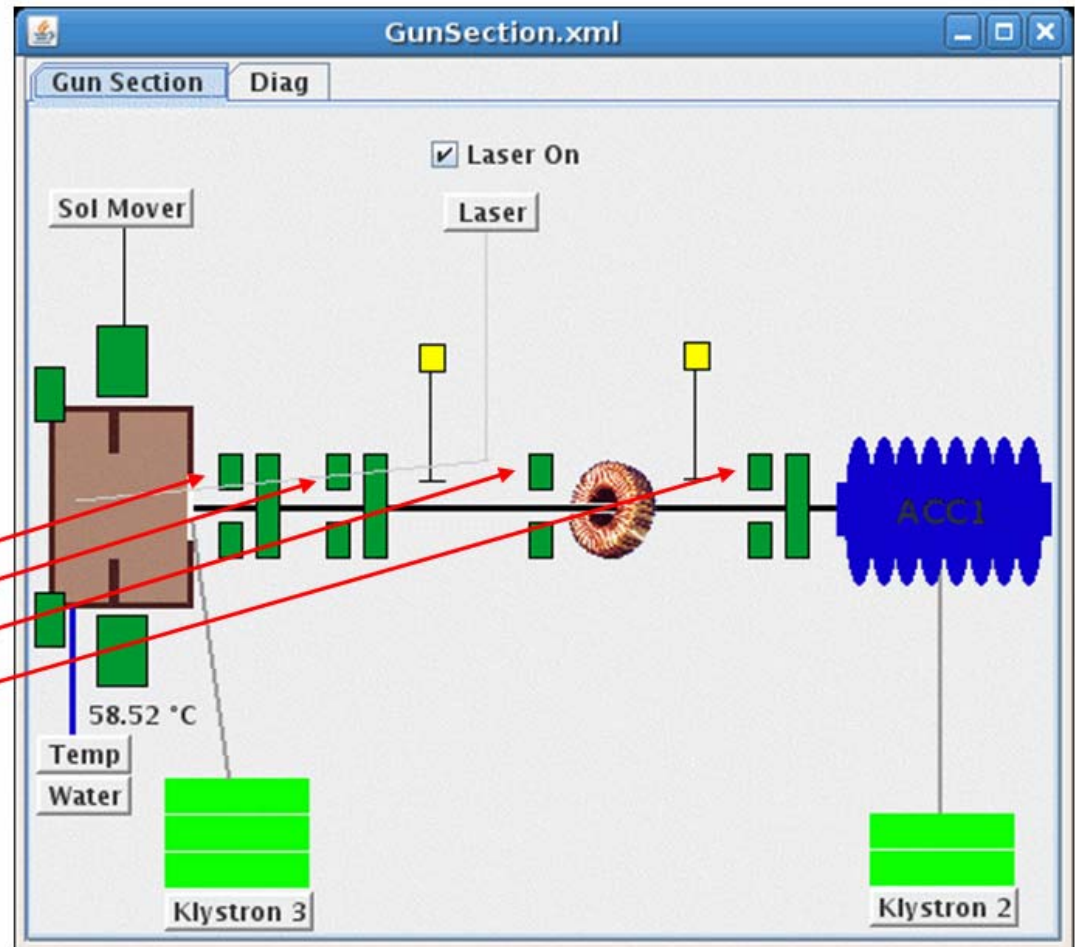


Pane Components

IncludeComponent

Library component which can
be used in multiple panels

steerer.xml



Dynamic Components

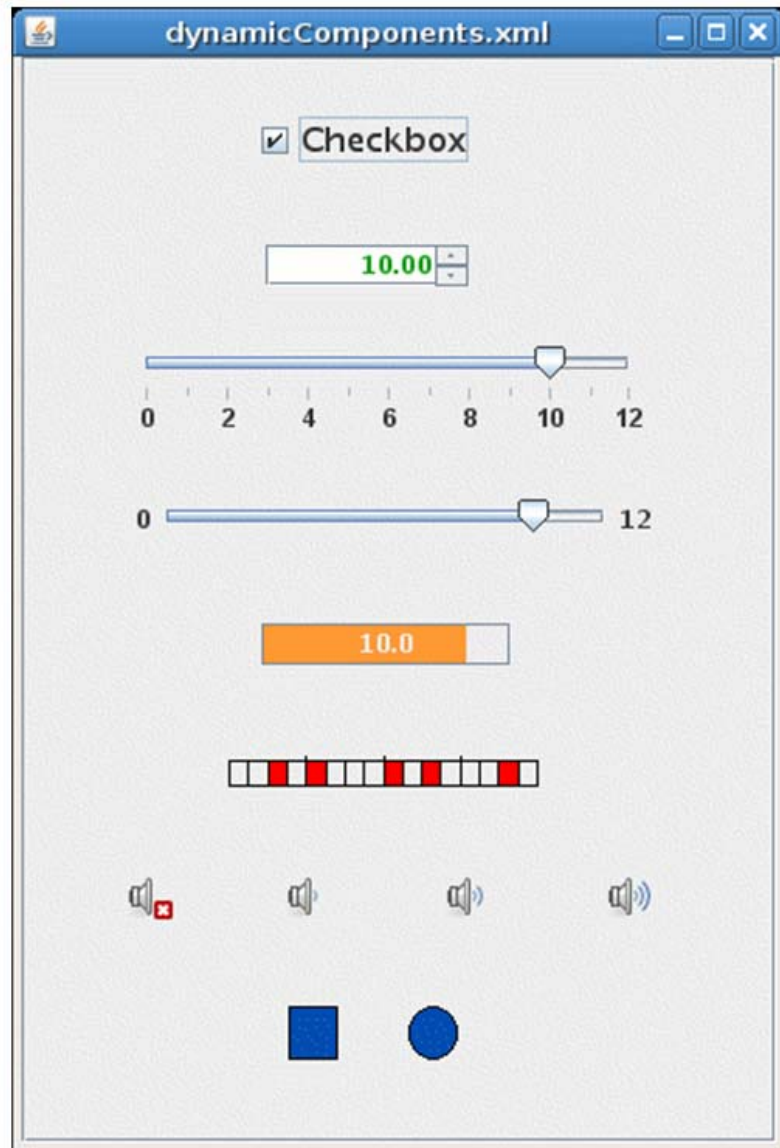
Checkbox

Dial

Slider

ProgressBar

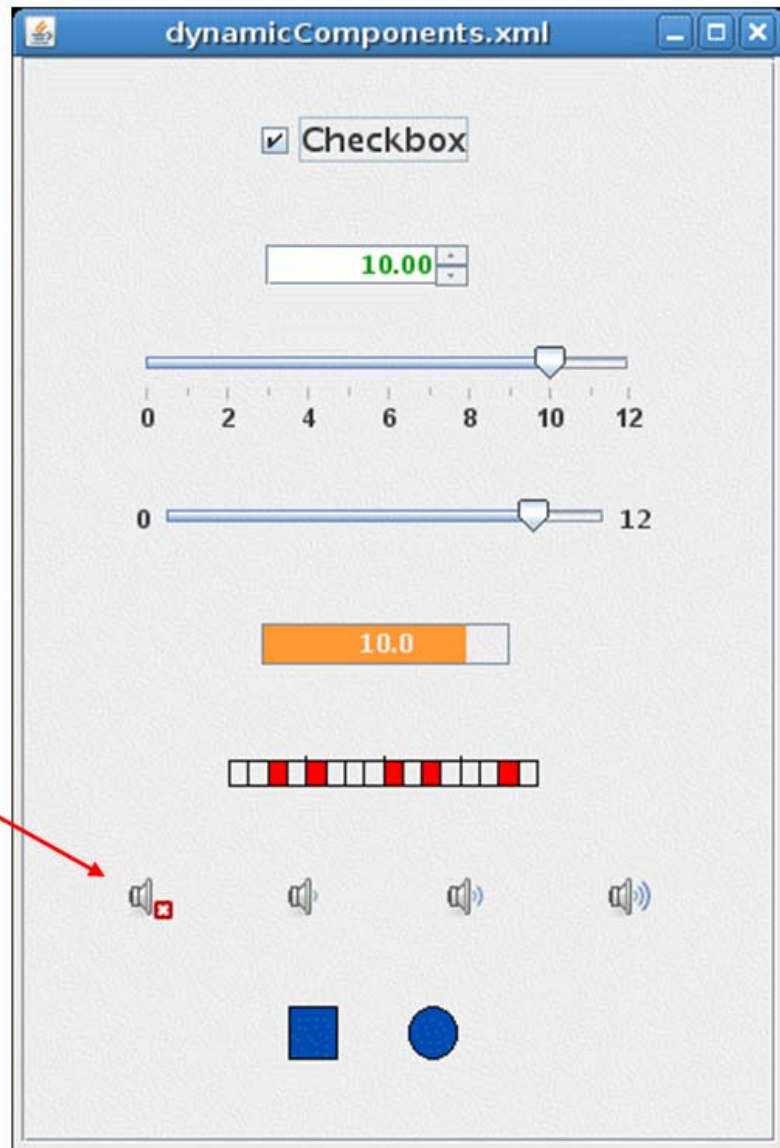
StatusRegister



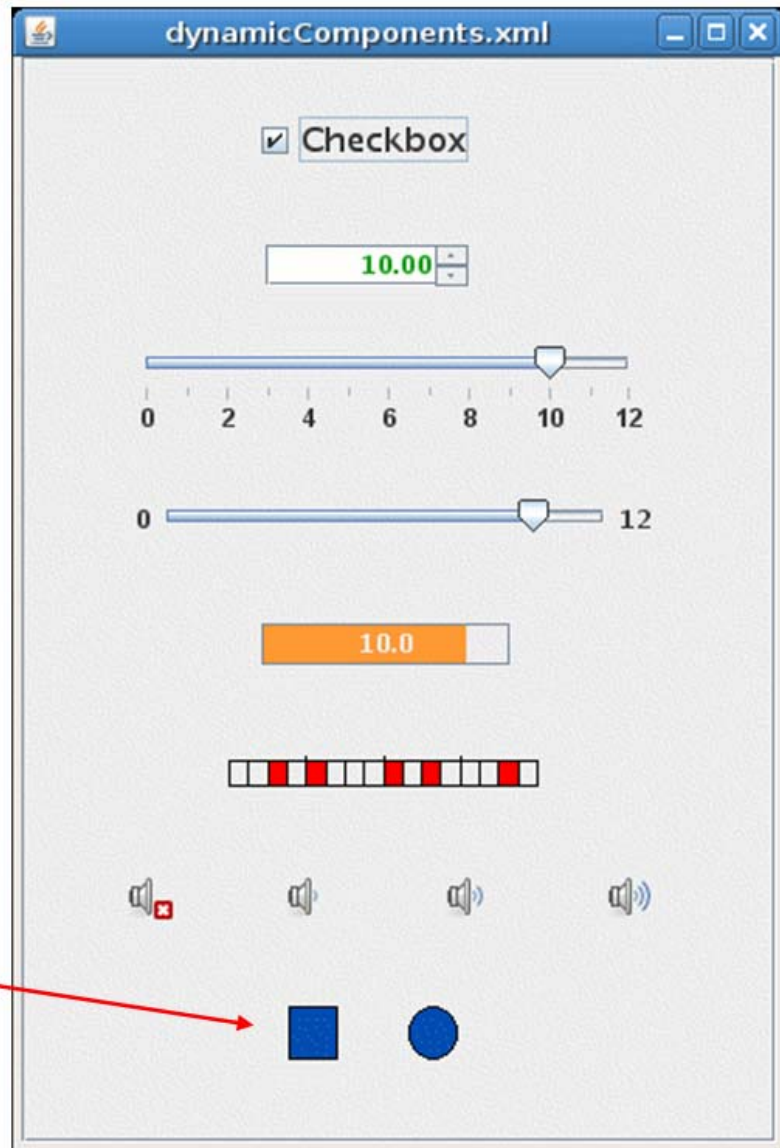
Dynamic Components

Audio Component

- plays .wav file
- plays dynamic beep
f(variable in contr. sys)



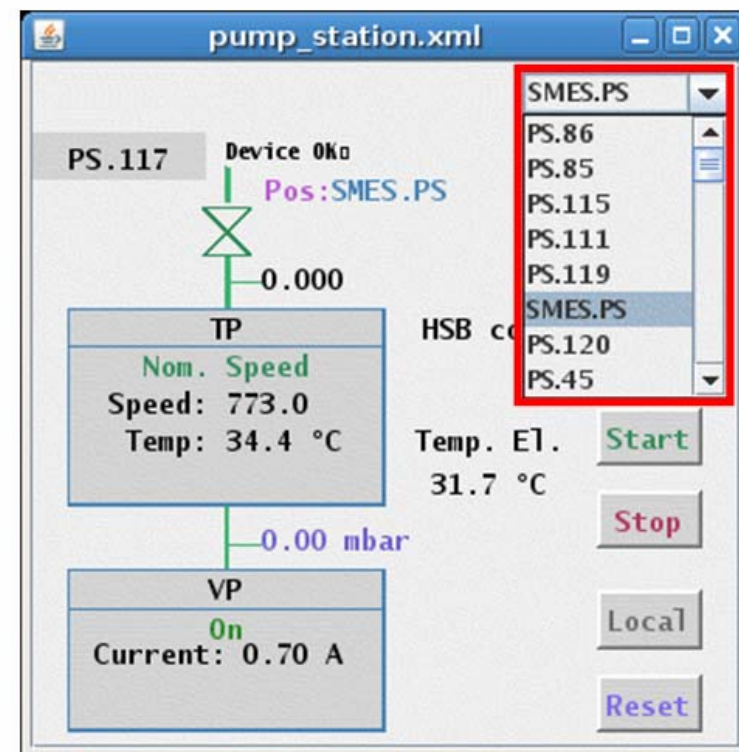
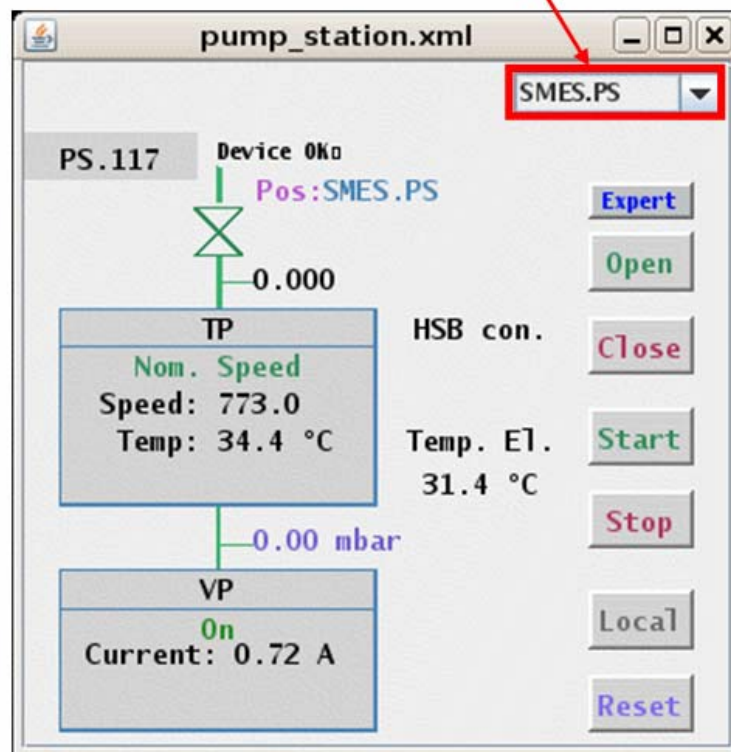
Dynamic Components



ColouredIndicator
- colour(variable in contr. sys)

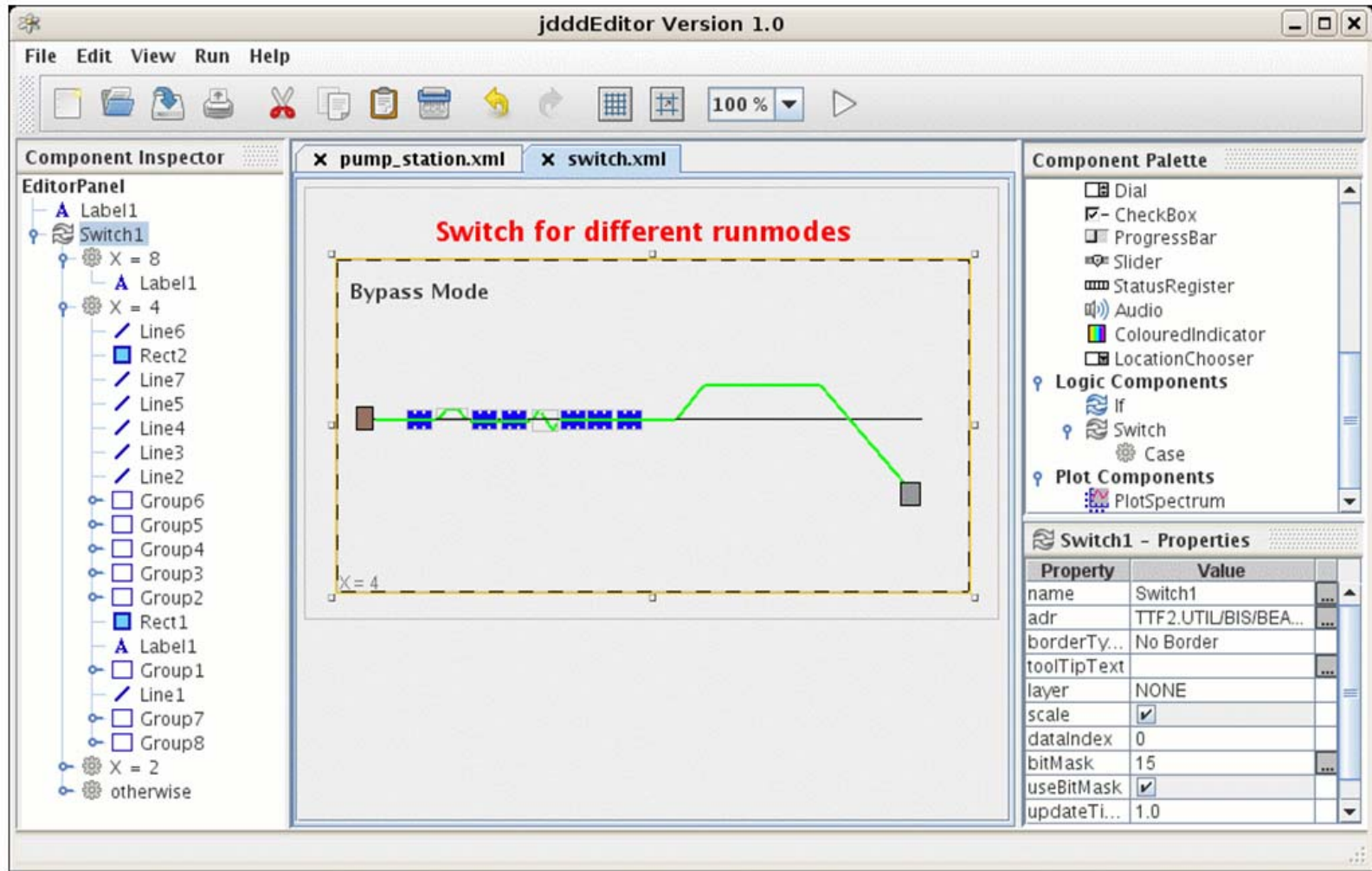
Dynamic Component LocationChooser:

changes the base address of the
whole panel



Logic Component

Switch: selects one of multiple layers



The screenshot shows the jdddEditor Version 1.0 interface. The main window displays a diagram titled "Switch for different runmodes" with a sub-label "Bypass Mode". The diagram features a green line representing a signal path, which is controlled by a switch component. The switch is currently set to "X = 4".

The left panel, labeled "Component Inspector", shows a tree view of the diagram's components. The "Switch1" component is selected, and its properties are listed below it:

- Label1
- Switch1
- X = 8
- Label1
- X = 4
- Line6
- Rect2
- Line7
- Line5
- Line4
- Line3
- Line2
- Group6
- Group5
- Group4
- Group3
- Group2
- Rect1
- Label1
- Group1
- Line1
- Group7
- Group8
- X = 2
- otherwise

The right panel, labeled "Component Palette", lists various components available for use in the diagram. The "Logic Components" section is expanded, showing the "Switch" component selected.

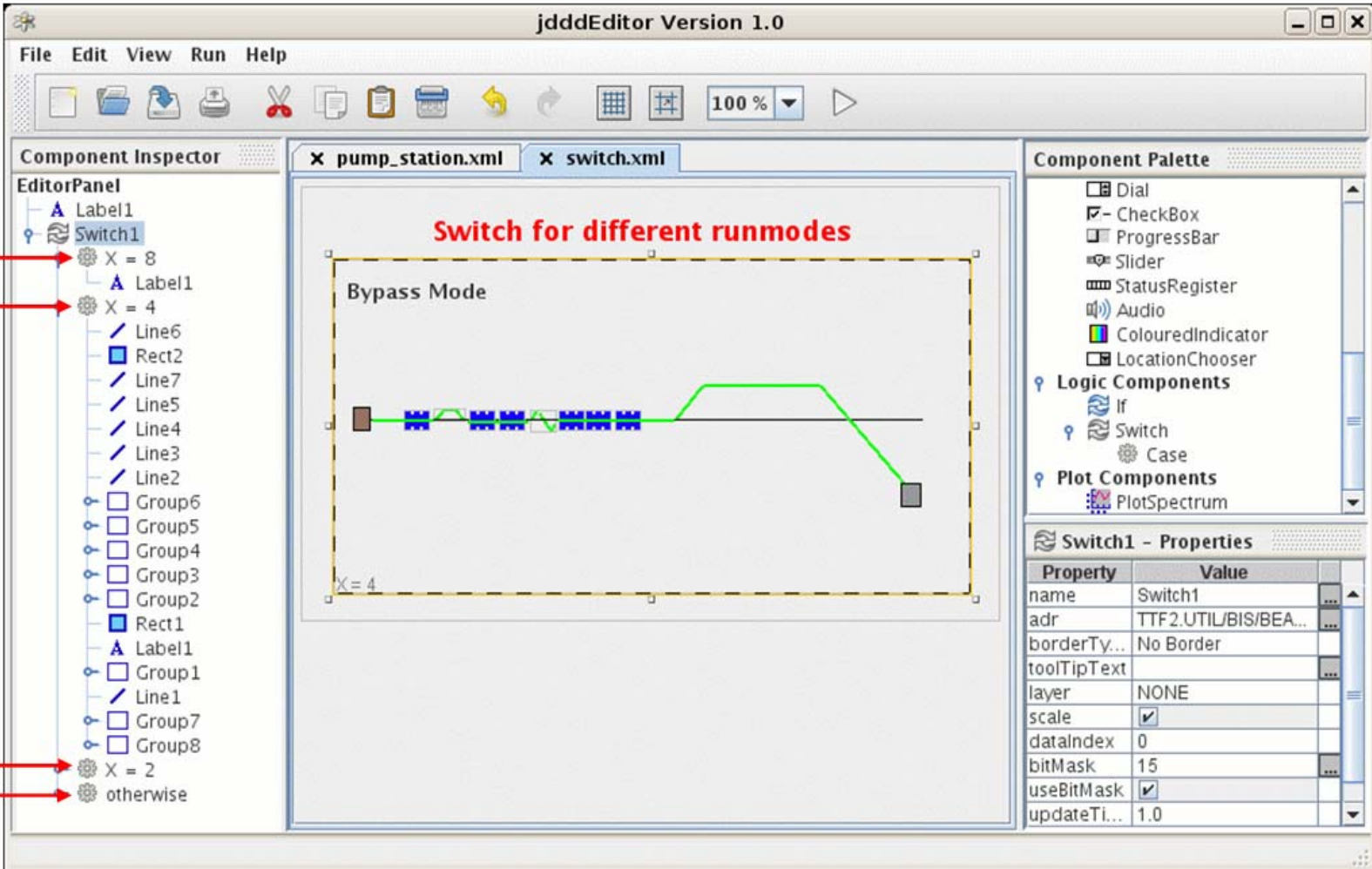
The bottom right panel, labeled "Switch1 - Properties", displays the properties of the selected switch component:

Property	Value
name	Switch1
adr	TTF2.UTIL/BIS/BEA...
borderTy...	No Border
toolTipText	
layer	NONE
scale	☒
dataIndex	0
bitMask	15
useBitMask	☒
updateTi...	1.0

Logic Component

Switch: selects one of multiple layers

Cases

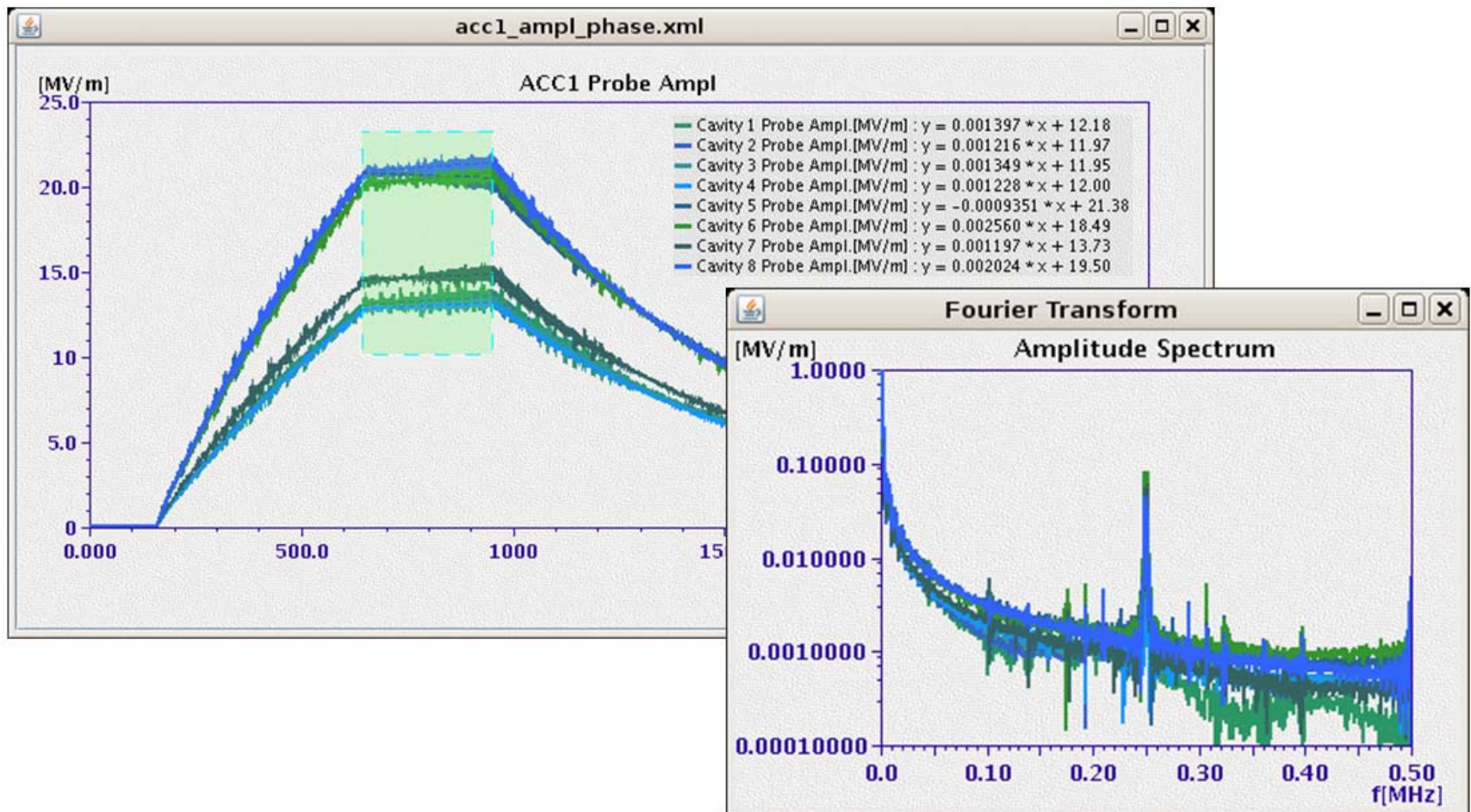


The screenshot shows the jdddEditor Version 1.0 interface. The main window displays a diagram titled "Switch for different runmodes" with a "Bypass Mode" section. A green line represents a signal path, and a dashed yellow box indicates the switch area. The left panel, labeled "Component Inspector", shows a tree structure of components. The "Switch1" component is selected, and its properties are shown in the right panel. The "Switch1 - Properties" panel includes a table of properties and values.

Property	Value
name	Switch1
adr	TTF2.UTIL/BIS/BEA...
borderTy...	No Border
toolTipText	
layer	NONE
scale	☒
dataIndex	0
bitMask	15
useBitMask	☒
updateTi...	1.0

Plot Component

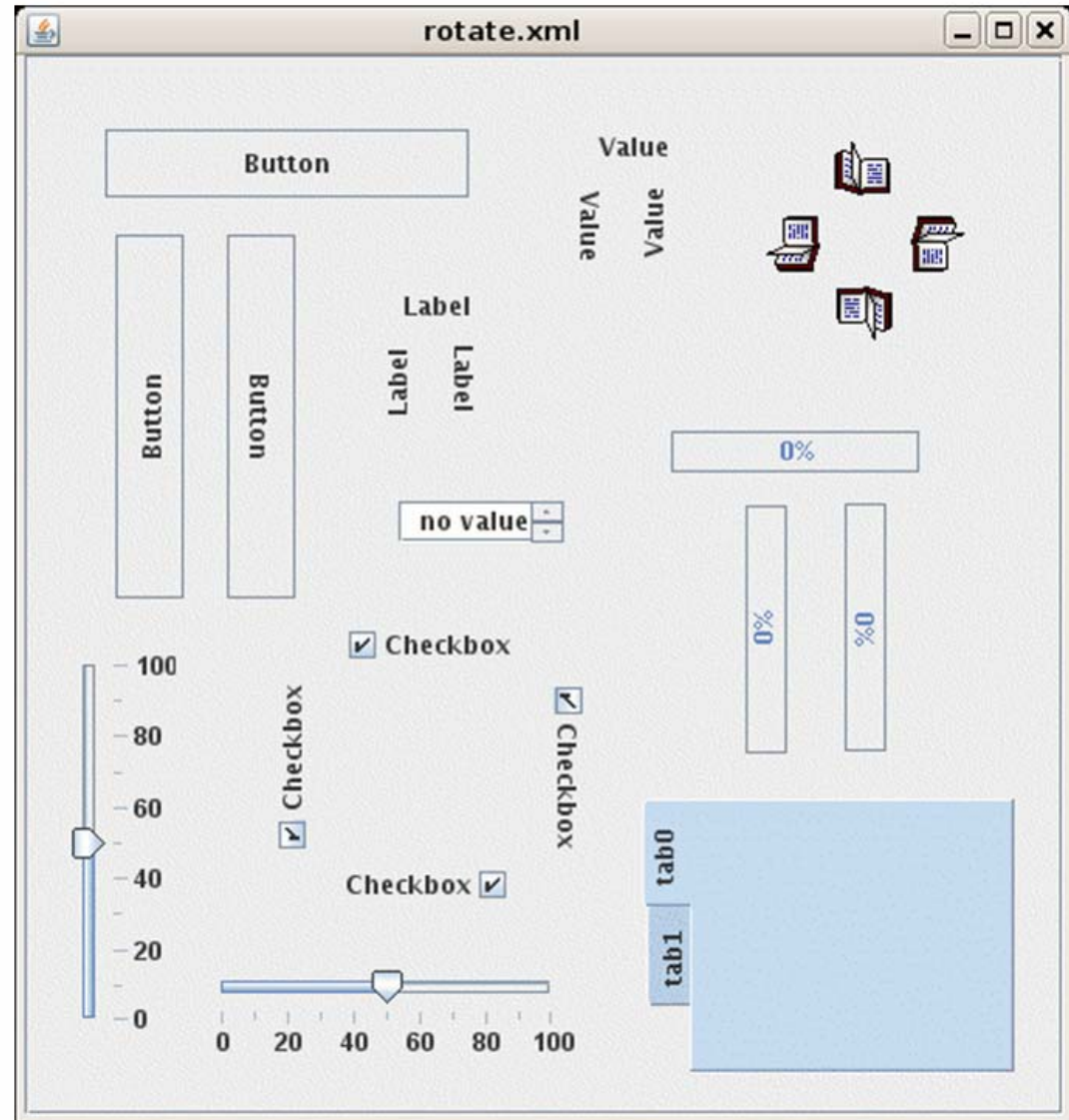
Plots with mathematical functions, autoscale



Special Features:

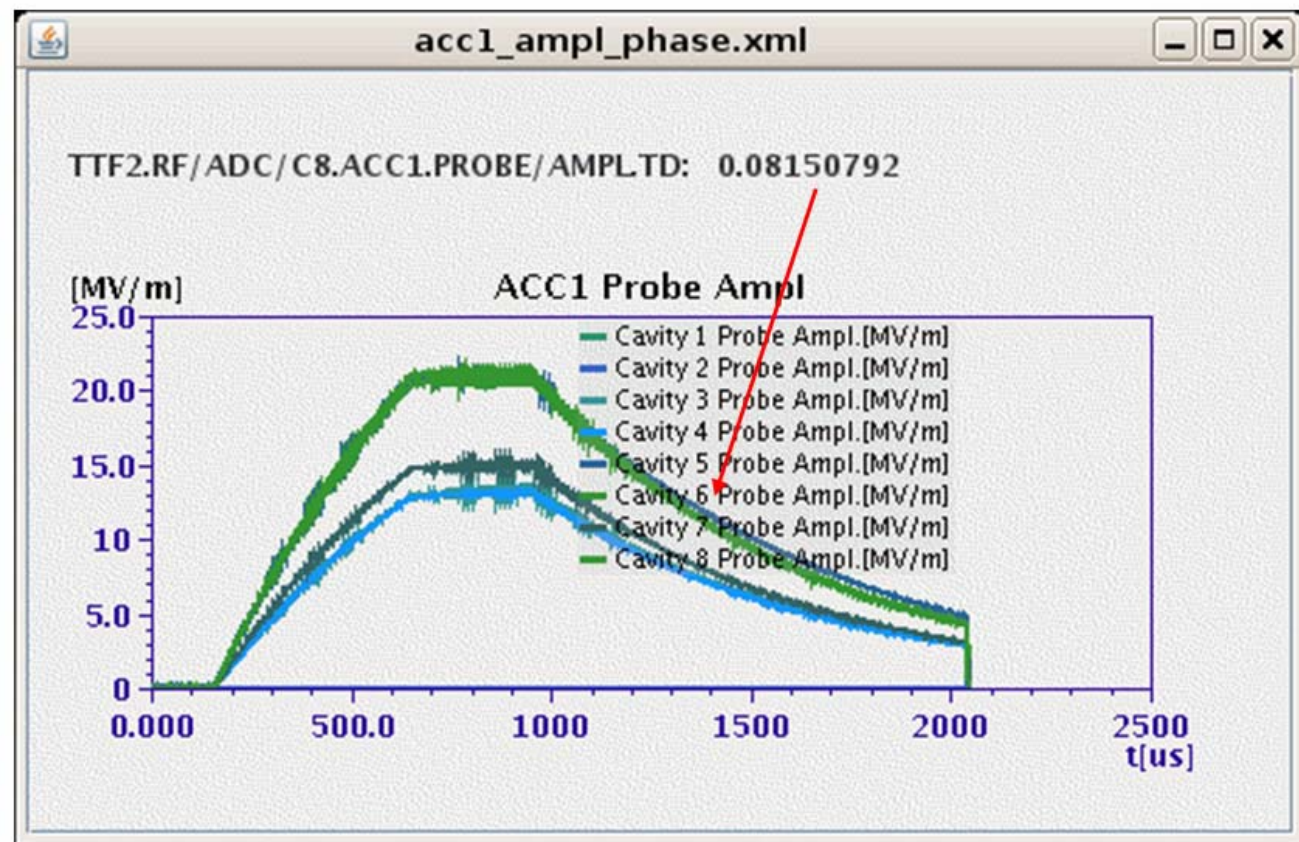
- Rotated components
- Drag & Drop
- Tooltips
- Layers
- Applet
- Use jddd for high level applications
- Convert old text-files to new xml file format

Rotated Components

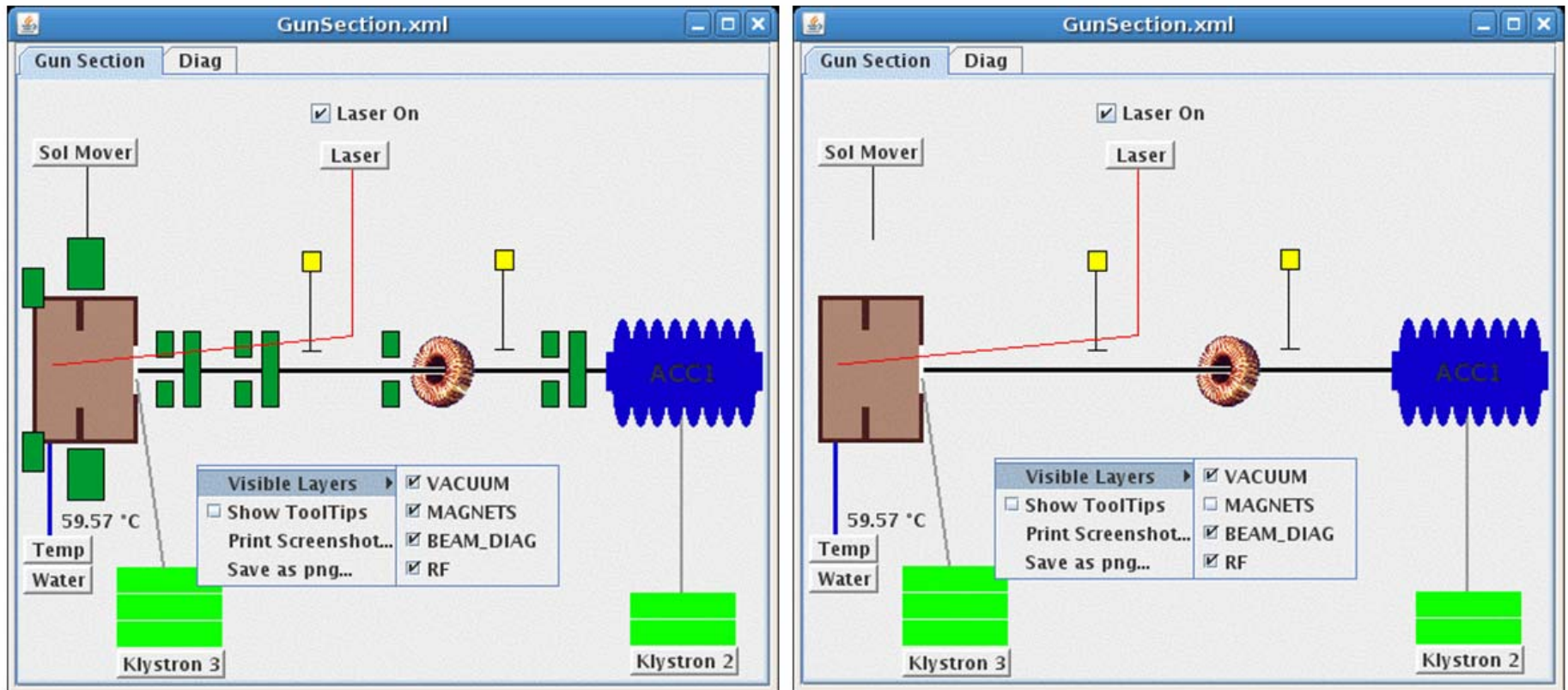


Drag & Drop

A control system address can be dragged & dropped from one component to another.



Layers



jddd for high level applications:

1) Export jddd panels as Java source code:

- Export as JFrame or JPanel
- Disadvantage: Panels can't be revised with the jddd editor after the export.

2) Use jddd panels as Java Beans:

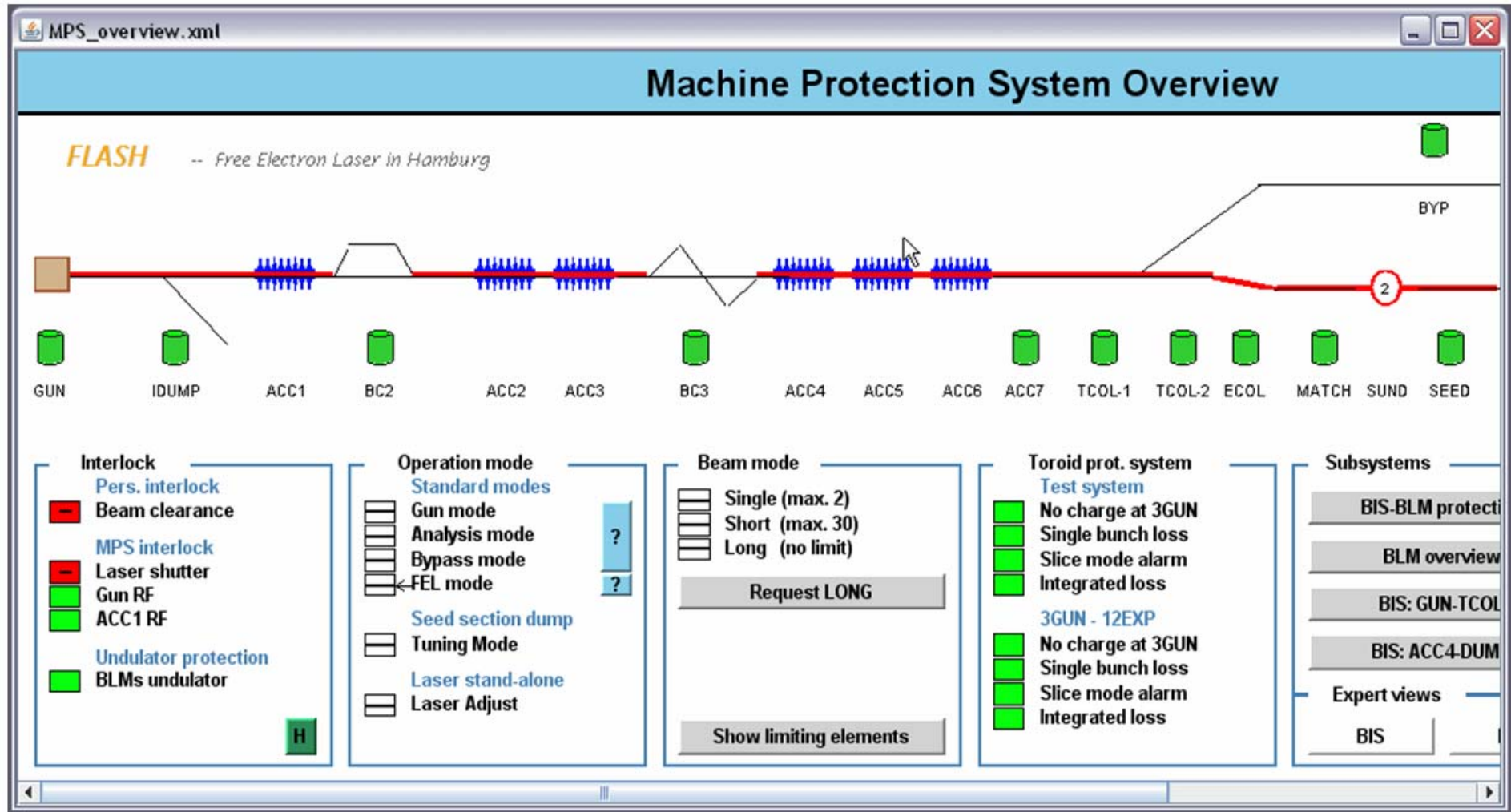
- Create a java application and add the following lines to insert a jddd panel as Java Bean:

```
jdddPanel panel = new jdddpanel();  
panel.setXmlFile(new java.io.File("/home/ttflinac/jdddFileName.xml"));
```

- To access components of the jddd panel:

```
DOOCSOval oval1 =  
(DOOCSOval)panel.getDoocsComponent("LayeredPane1/Oval1");  
oval1.setDoocsFillColor(Color.orange);
```

Old ddd panel converted to new jddd panel.



How to start jddd

<http://jddd.desy.de/>

Java Web
Start

jddd

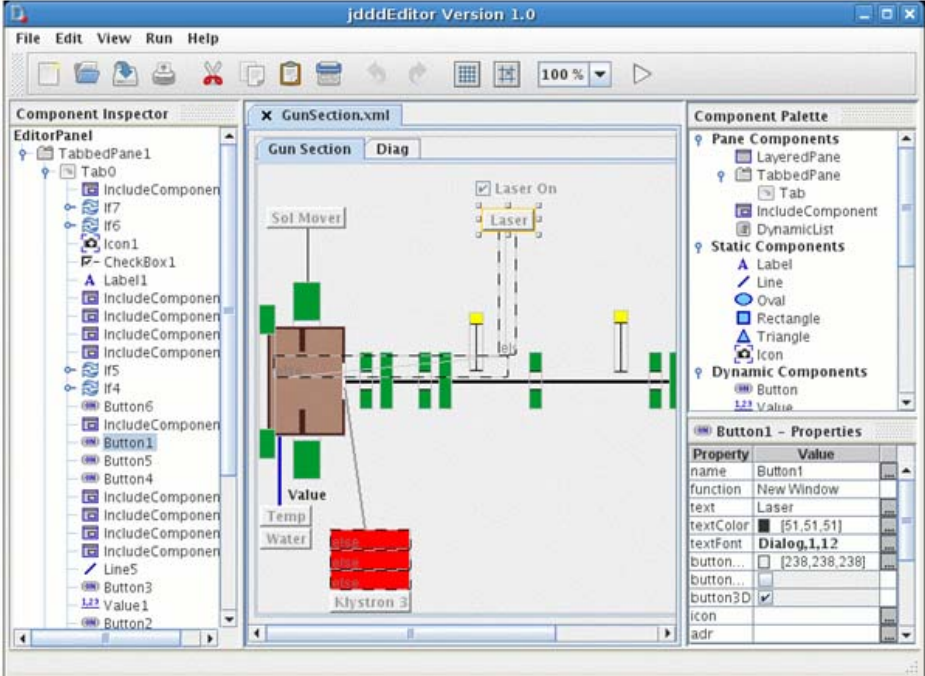


- Introduction
- DOOCS Home
- Screenshots:
 - TabbedPane
 - IncludeComponent
 - DynamicList
 - Slider, ProgressBar, etc.
 - Audio, ColouredIndicator
 - LocationChooser
 - TextArea
 - Switch
 - PlotSpectrum
 - Converted ddd panel
- **JAVA Web Start**
- Flash Tutorial
- Help
- Contact

DOOCS

jddd Introduction

The Java DOOCS Data Display is a program to draw control system panels with an editor and to run the created drawings with real data. Two versions of the program are available. One is the jdddEditor to draw and test the drawings. The second one is a run-time version only, which does not allow to change a drawing. The jdddEditor offers reusable Java components like labels, buttons, plots, etc. to draw control system panels. The panel structure is stored in xml-format.



General Experience:

- Java has proved to be a good choice for GUI development
- Java / jdoocs API is fast enough
- Java Webstart works fine

Outlook:

- Replace ddd by jddd as fast as possible
- Save xml-files in a subversioning system
- and many more ideas...

Thank you for your attention!