BOY, A Modern Graphical Operator Interface Editor and Runtime

Xihui Chen, Kay Kasemir
chenx1@ornl.gov
PAC’11
What is BOY?
What is BOY?

- BOY (Best OPI, Yet) – An Operator Interface (OPI) development and runtime environment
- OPI – Graphical Interface to view or operate the accelerator locally or remotely
What is BOY?

• Best of SDS (Synoptic Display Studio: DESY)
  ✓ Java portability
  ✓ CSS integration
  ✓ Every property can be dynamic
  ✓ Modern Graphical Editor based on GEF
    (BOY reused some GEF-related SDS code)

• Best of EDM (Extensible Display Manager: John Sinclair)
  ✓ Simple things (Label, Textupdate, …) are simple
  ✓ Macros

• Combined with New Ideas
  ✓ Dynamic via PV-triggered scripts or rules
  ✓ Web browser like Runtime (tab, CTRL, SHIFT click)
  ✓ Comprehensive types of Widgets
  ✓ Good ideas from EPICS community

Thanks to the creators of SDS and EDM for their great efforts!
**OPI Editor**

- All-In-One workbench for OPI editing
OPI Editor

- All-In-One workbench for OPI editing
OPI Editor

• What You See Is What You Get (WYSIWYG)

• Comprehensive editing functions on toolbar and context menu
  – Copy/Paste/Delete
  – Drag & Drop
  – Undo/Redo
  – Alignment & Distributing
  – Snap to G (Grid/Geometry/Guide)
  – Zoom In/Out
  – Copy Properties
  – Changing Orders
  – …
A simple OPI

- Create a functional OPI in a breeze
  1. Create a new OPI file
  2. Drag a widget (Knob for example) from palette to editor
  3. Enter the PV name in Properties view
  4. Click the “Run” button to execute it!

- What you will get
  - PV value as text and via knob position
  - PV severity reflected in border color
  - PV name and value shown in tool-tip
  - PV’s display limits set the knob’s default range
  - PV’s HiHi, High, Low, LoLo limits displayed in ramp
  - Indicate ‘disconnected’ state via a pink border
  - Widget will be greyed-out if write not allowed (read-only)
OPI Runtime

- Behaves like a Web Browser
  - OPIs display in Tabs. Tabs can be rearranged.
  - Open related OPI in a new Tab via Ctrl+click, or in a new Window via Shift+click.
  - Navigate Backward/Forward
  - Zoom In/Out
  - Full Screen/Compact Mode
  - Screenshot to printer, Email, Logbook
Rules
- Easily make widget properties dynamic
  - Directly send PV’s value to a property
  - Condition dependent property value
  - Multiple rules per widget
Scripts
- Intelligentize your OPI

```javascript
importPackage(Packages.org.eclipse.jface.dialogs);
importPackage(Packages.org.csstudio.opibuilder.scriptUtil);

var flagName = "popped";

if(widgetController.getExternalObject(flagName) == null){
    widgetController.setExternalObject(flagName, false);
}

var b = widgetController.getExternalObject(flagName);

if(FVUtil.getDouble(pvArray[0]) > 80){
    if(b == false){
        widgetController.setExternalObject(flagName, true);
        MessageDialog.openWarning(null, "Warning", "The temperature you set is too high!");
    }
}
else if(b == true){
    widgetController.setExternalObject(flagName, false);
}
```

![Image of a warning message box with a temperature of 82°](image)
Scripts
- Intelligentize your OPI

```javascript
importPackage(Packages.org.eclipse jäface.dialogs);
importPackage(Packages.org.csstudio.opibuilder.scriptUtil);

var flagName = "popped";

if (widgetController.getExternalObject(flagName) == null) {
    widgetController.setExternalObject(flagName, false);
}

var b = widgetController.getExternalObject(flagName);

if (FVUtil.getDouble(pvArray[0]) > 80) {
    if (b == false) {
        widgetController.setExternalObject(flagName, true);
        MessageDialog.openWarning(null, "Warning", "The temperature you set is too high!");
    }
} else if (b == true) {
    widgetController.setExternalObject(flagName, false);
}
```

Call Java code
## Macros

- **Format:** \$(macro\_name) or \${macro\_name}
- **Embedded in text-base properties**
  - PV Name, tooltip, rules …
- **Replaced at runtime**
- **Powerful way for duplicating**
Drag & Drop

- Drag Source could be text or PV
- Drop to OPI editor to create widgets
Widgets

- Support various data types
  - Double, Integer, String, Enum, Boolean, Waveform...
  - Allow plugging in customized widgets
- Data Browser Widget – browsing history and live data in BOY
Technical View

- Coding in Java, based on Eclipse, GEF and CSS platform
- Portable to Windows, Mac OS and Linux
- BOY is a set of Eclipse Plugins
- Integrated with CSS natively, also possible to integrate it with other RCP applications
BOY At SNS

• SNS still primarily on EDM
• BOY is used for special displays
BOY At SNS

- Top-level displays created by operators
BOY at SNS

- Special OPIs
  - BLM
  - Save & Restore
BOY At SNS

- Used in laboratories for hardware and IOC test
Collaborators

• John Hammonds (APS)
  – MEDM to BOY Converter
  – Byte Monitor

• Ralph Lange (BNL): Many good suggestions

• ITER CODAC Group: Many good suggestions

• EPICS community
People using BOY

The data is based on my email records
Summary

• BOY is an integrated OPI Editor and Runtime
  – Technically, Java/Eclipse RCP plugins
• Modern graphical editor with comprehensive functions to support your OPI creation
• OPI Runtime behaves like a web browser
• Rules and JavaScripts can add logic to your OPI
• Your feedback is important!
Thank you!

- BOY Home Page
  - http://sourceforge.net/apps/trac/cs-studio/wiki/BOY

- Download
  - SNS CSS Home Page:
    http://ics-web.sns.ornl.gov/css
  - Unpack and Run CSS, no installation needed.

- Tutorials
  - Online Help
  - BOY Examples

Based on
Planning

• EDM2BOY Converter
  – Basic widgets are convertible now
  – More widgets will be added

• WebOPI
  – Run BOY OPI in web browser
  – Under investigation

• Continually improving BOY by listening to users’ feedback