f

# Drag and Drop Controls Display and Builder

(Synoptic Display)

Timofei Bolshakov, Andrey Petrov
Fermilab Accelerator Controls Department

March 26, 2007

f

# Motivation

- Operations of complicated colliders (Tevatron, the LHC , and the ILC ) require sophisticated control systems.
  - Large interlocking sub-systems
  - Security
  - Data pool management
  - Alarms
  - Logging
- The people who must build, operate, and maintain these accelerators
  - Operators
  - Engineers
  - Accelerator physicists

  require rapid development of control displays and application programs.

f

# Motivation

- For rapid development, the system expert (operator, engineer, or physicist) should be the one to develop the displays or applications

- These advanced control systems can seem overwhelming to non controls experts.
  - This is why Lab View is so popular
  - However Lab View offers little of the benefits of an advanced control system.

f

# Solution

- Drag and Drop is an environment that gives non-control system experts the ability to quickly build controls displays <u>which operate inside the control system framework</u>.
- Drag and Drop:
  - is easy to use
  - sophisticated enough to handle complex displays
  - operates inside the control system framework.
  - uses web browsers and/or Java Web Start (requires no extra software installation)
  - is easily extendible
  - is a mature application
    - Developed in 2001
    - Fermilab Cryogenics department are heavy users
    - Well documented

f

# Drag and Drop Controls Display and Builder

- Drag and Drop a consists of two parts:
  - Display
  - Builder
- The Drag and Drop Display can be run from a web browser (readings only) so it can viewed anywhere in the world.
  - Files are stored on a web server (well organized and secure)
  - Displays are extremely quick because it uses Scalable Vector Graphics (SVG) so that the screen does not constantly have to be re-drawn.
- The Builder has a simple graphical user interface that offers a rich set of graphical components
  - Display can be built and deployed in a matter of minutes
  - The builder is easily extendible
  - can be run on any machine because it is based on Java

f

# Demo

- Cryo/Meson/CompressorRoom.xml

http://www-bd.fnal.gov/synoptic/start?id=http://ww

- florian/linac6.xml

http://www-bd.fnal.gov/synoptic/start?id=http://ww

- Cryo/A0/A0PiCryo9cell.xml

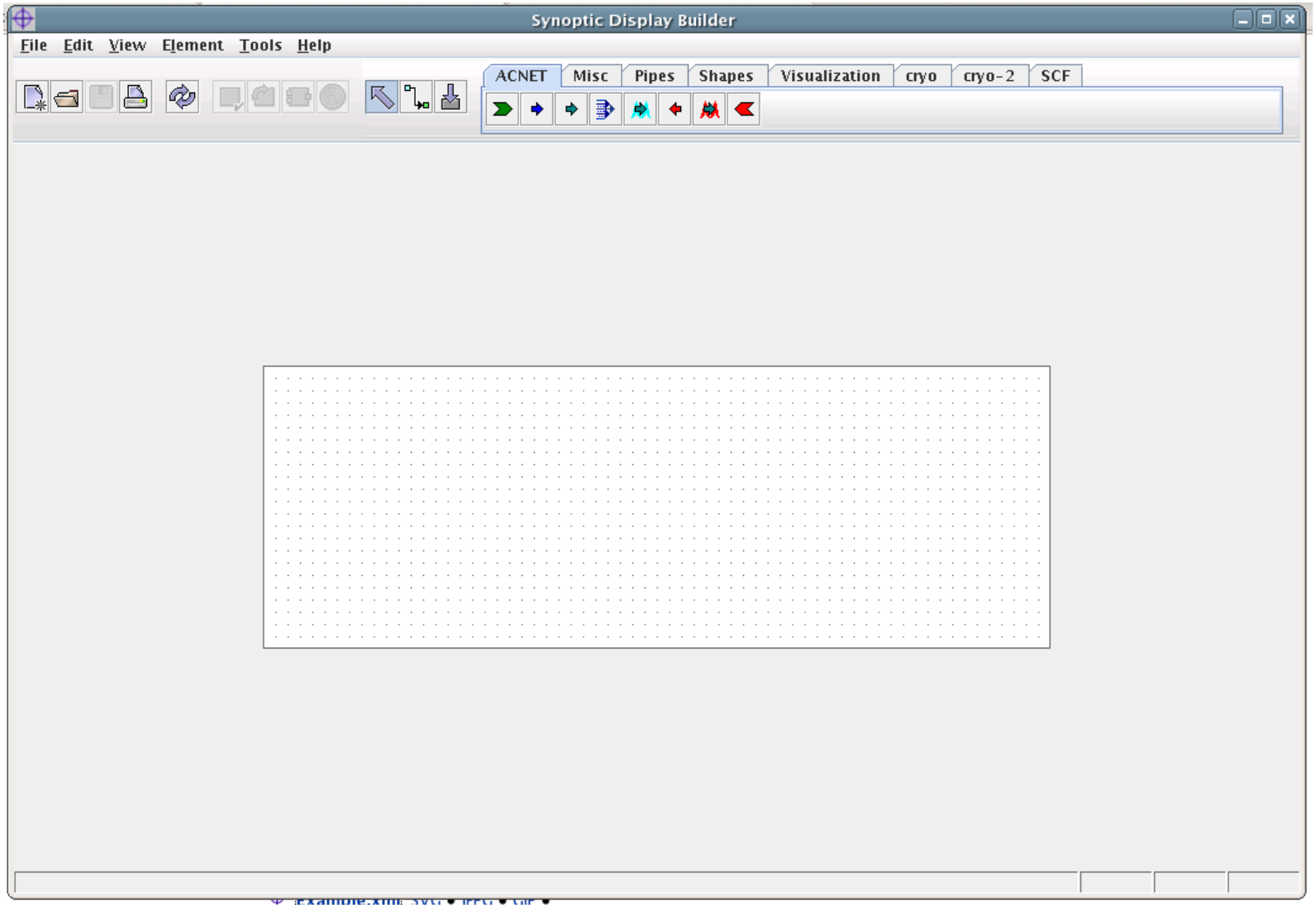
http://www-bd.fnal.gov/synoptic/start?id=http://ww

- test/Example.xml

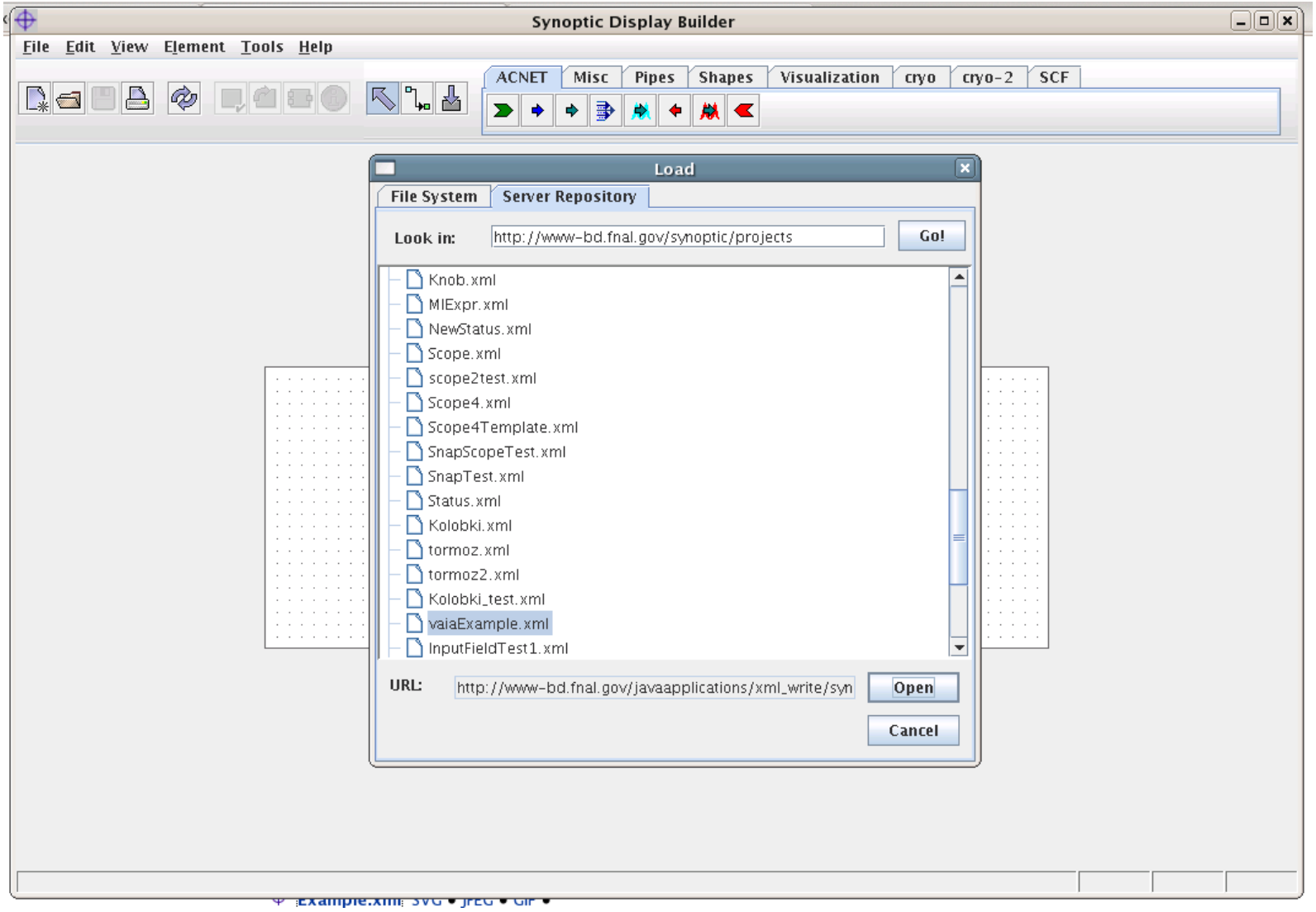
http://www-bd.fnal.gov/synoptic/start?id=http://ww

f

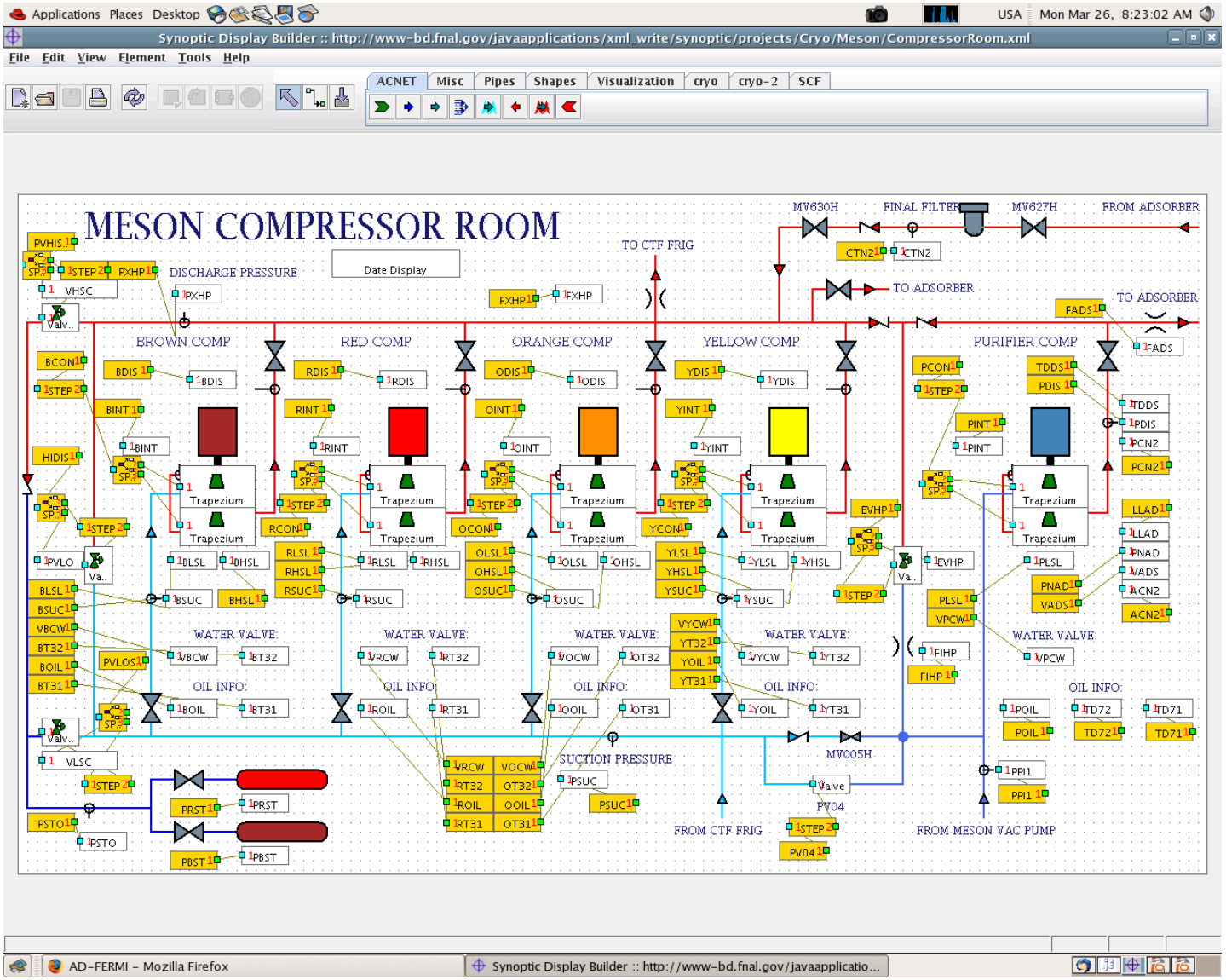
# Demo – Empty Builder



# Demo – Builder – Open Project

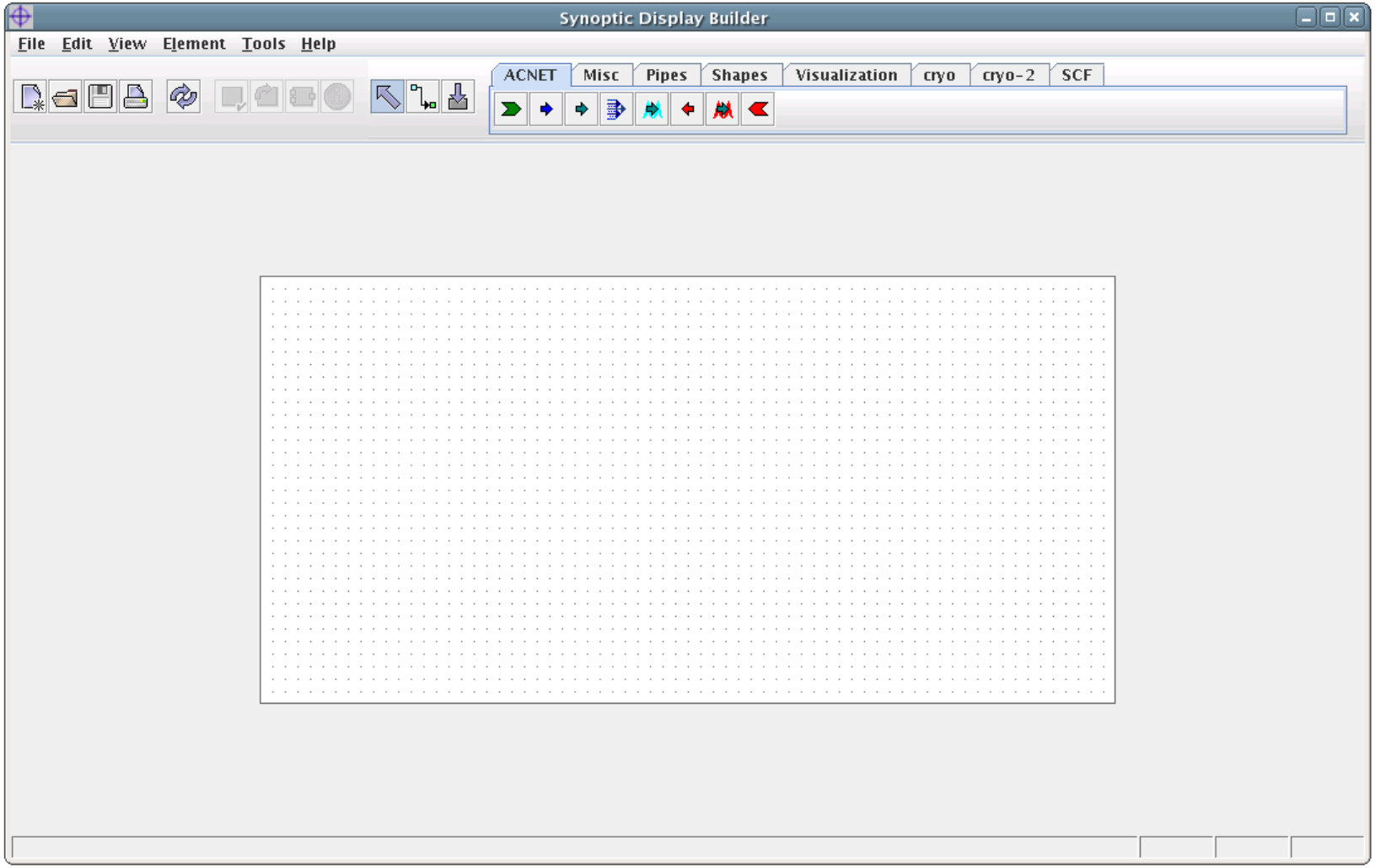


# Demo – Builder, Meson Compressor Room

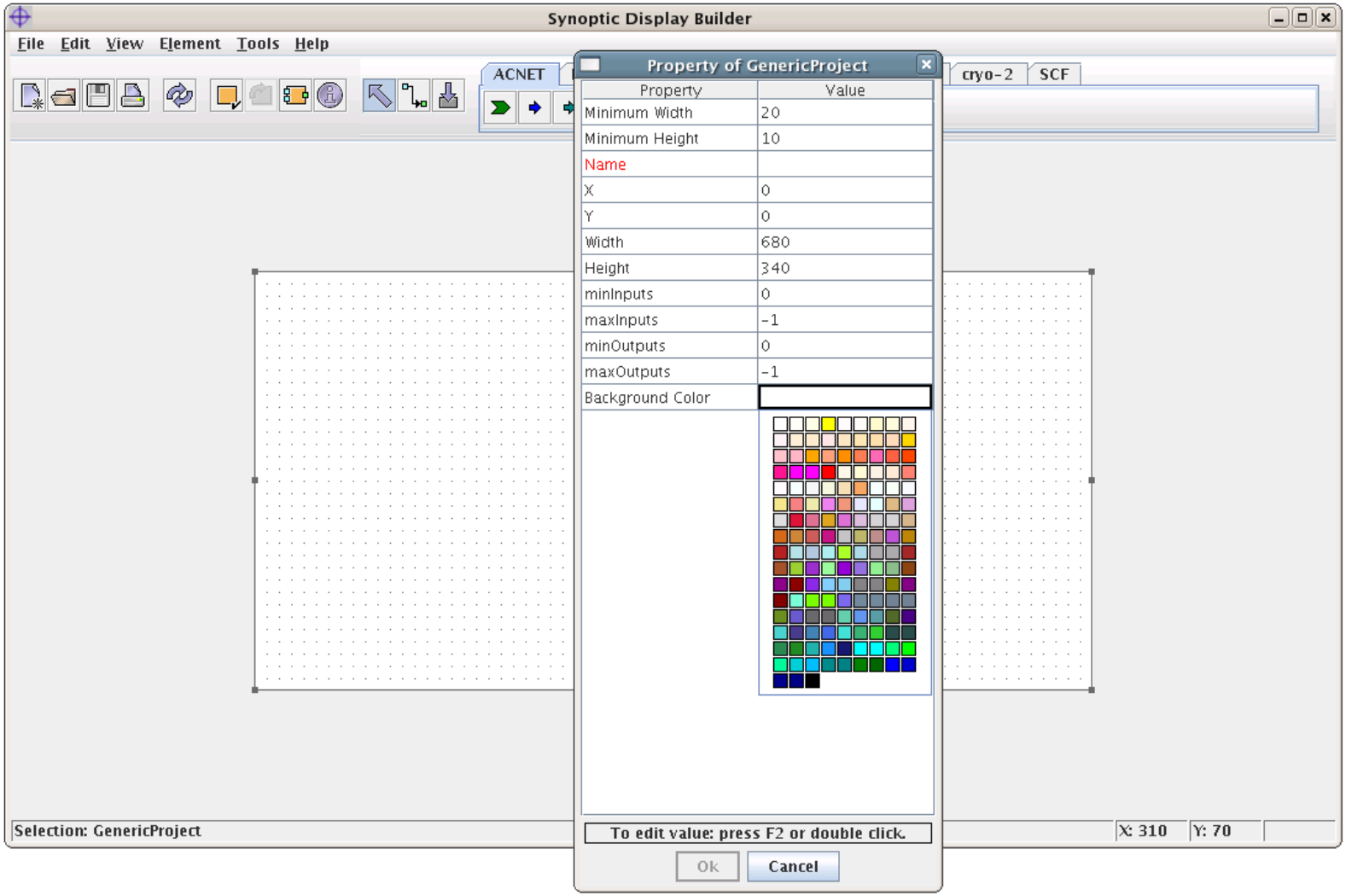


f

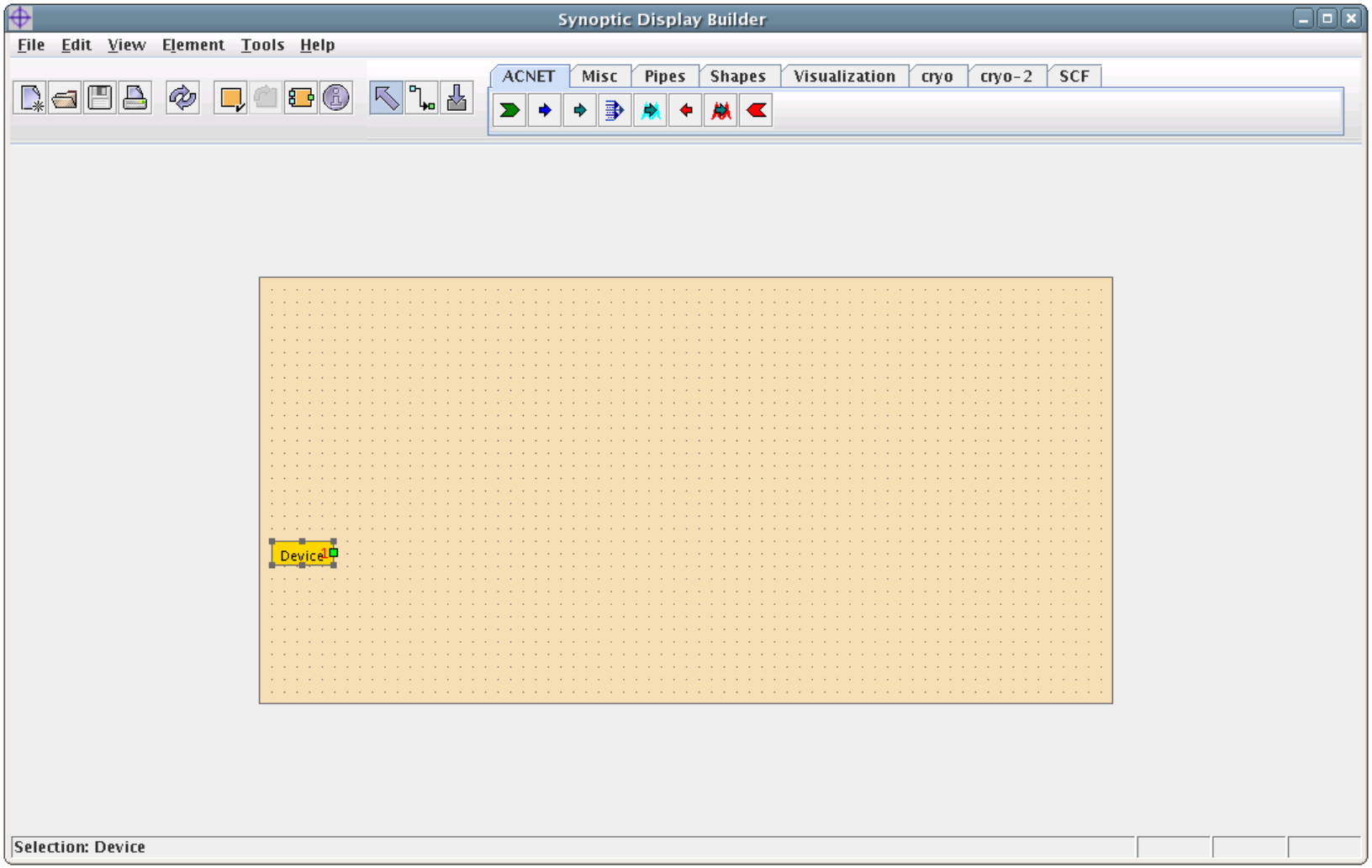
# Demo – Klystron Step 1, Empty Builder



# Demo – Klystron, Step 2, Project Properties



# Demo – Klystron, Step 3, Adding a Device



# Demo – Klystron, Step 4, Setting Up Device

Synoptic Display Builder

File Edit View Element Tools Help

ACNET cryo-2 SCF

Device

Property of Device

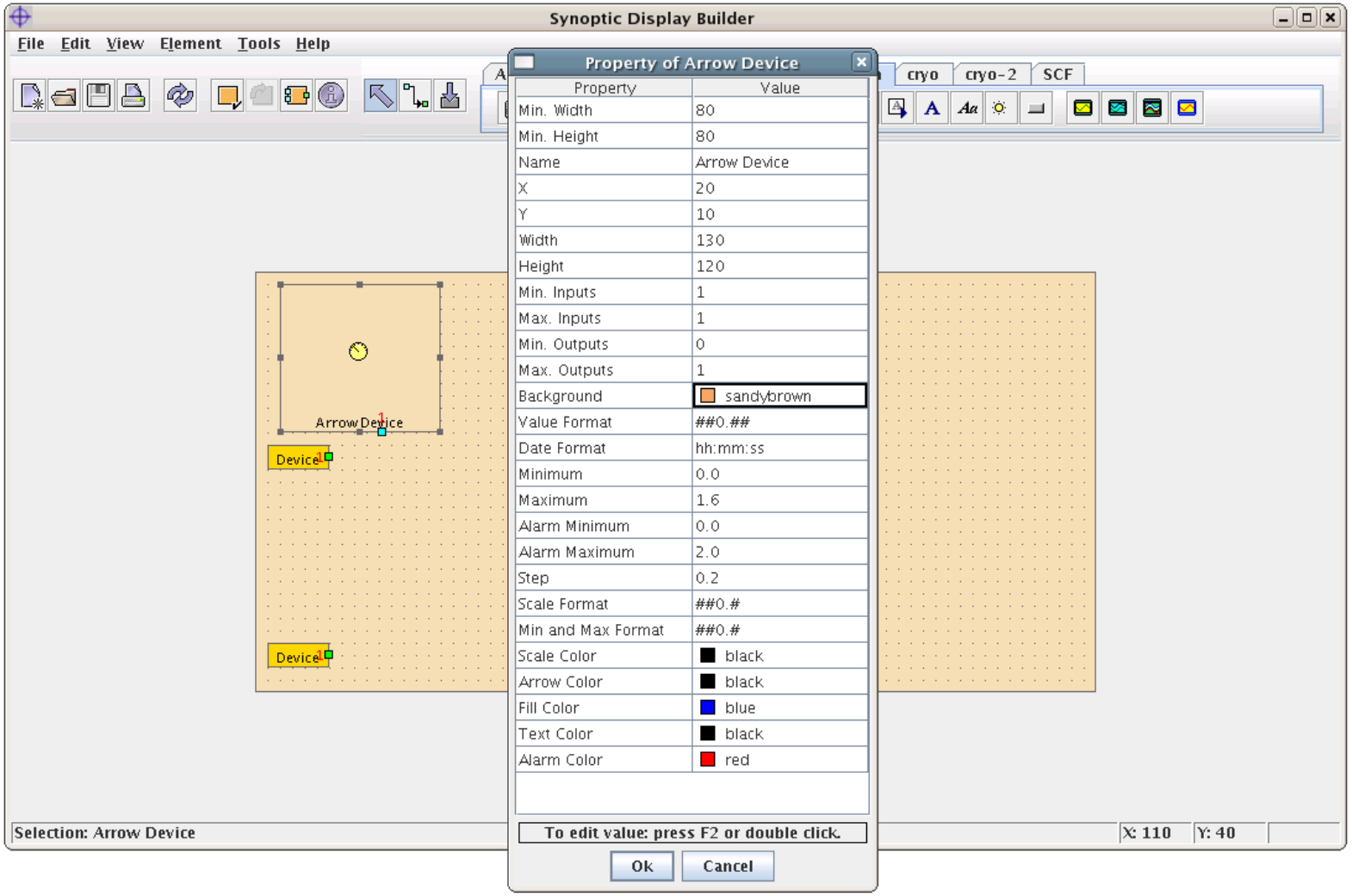
| Property | Value |
|---|---|
| Min. Width | 20 |
| Min. Height | 10 |
| Name | Device |
| X | 10 |
| Y | 210 |
| Width | 50 |
| Height | 20 |
| Min. Inputs | 0 |
| Max. Inputs | 0 |
| Min. Outputs | 1 |
| Max. Outputs | 1 |
| Background | gold |
| Device Name | L:C6GRAD,Reading |
| Event | periodic, 1Hz |
| Source | Accelerator |
| Reflect chages only | true |

To edit value: press F2 or double click.

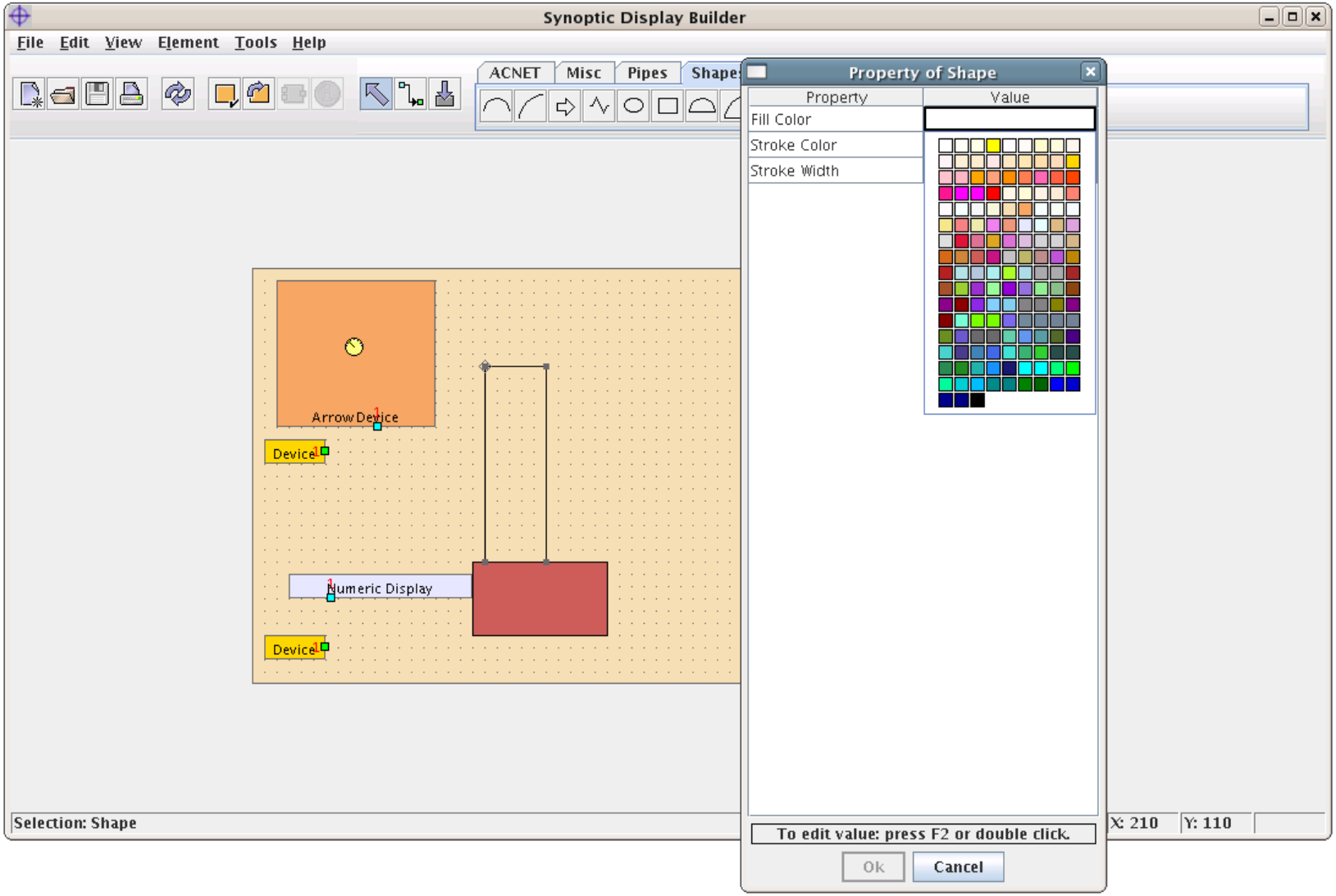
Ok Cancel

Selection: Device X: 40 Y: 220

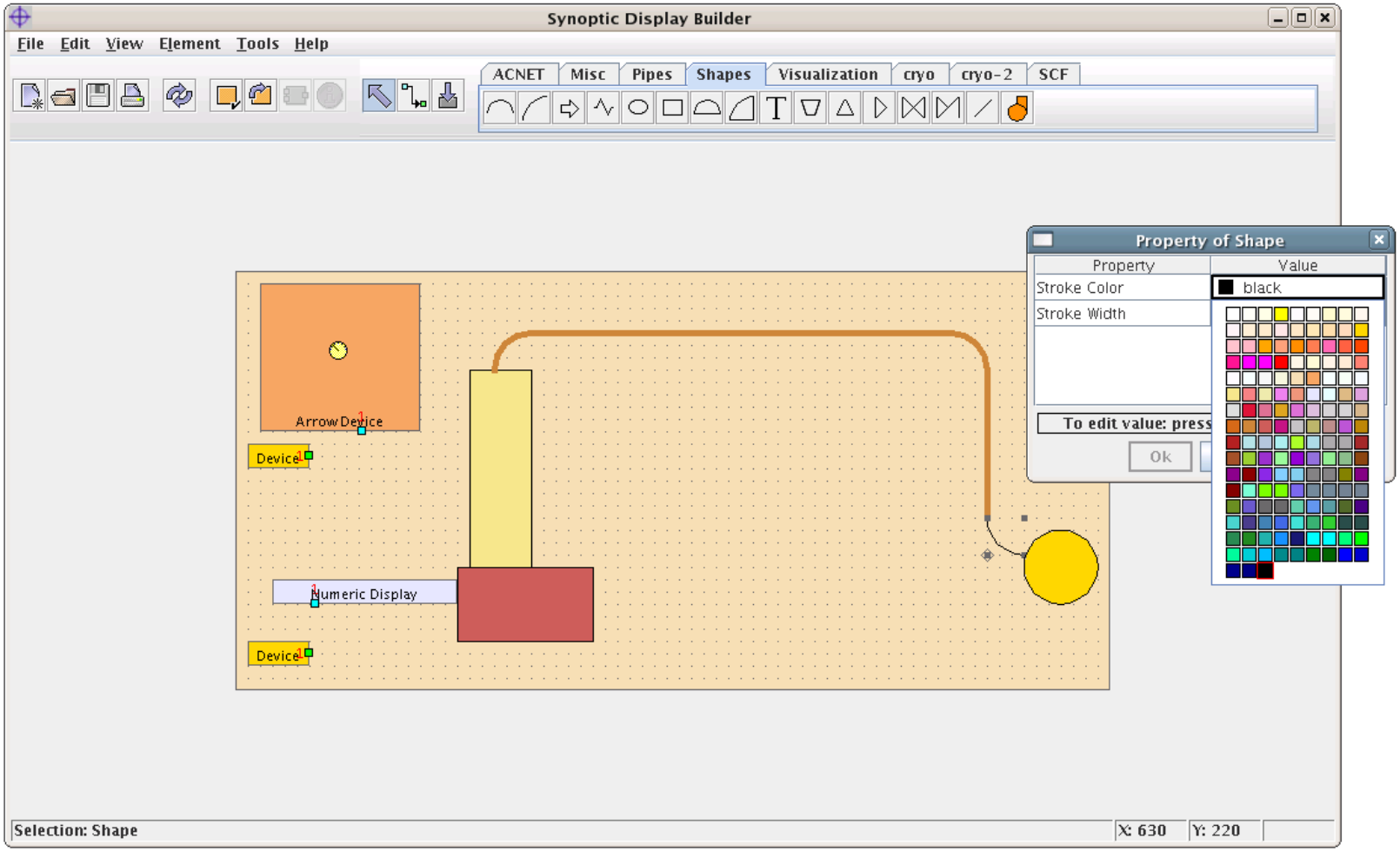
# Demo – Klystron, Step 5, Adding Indicator



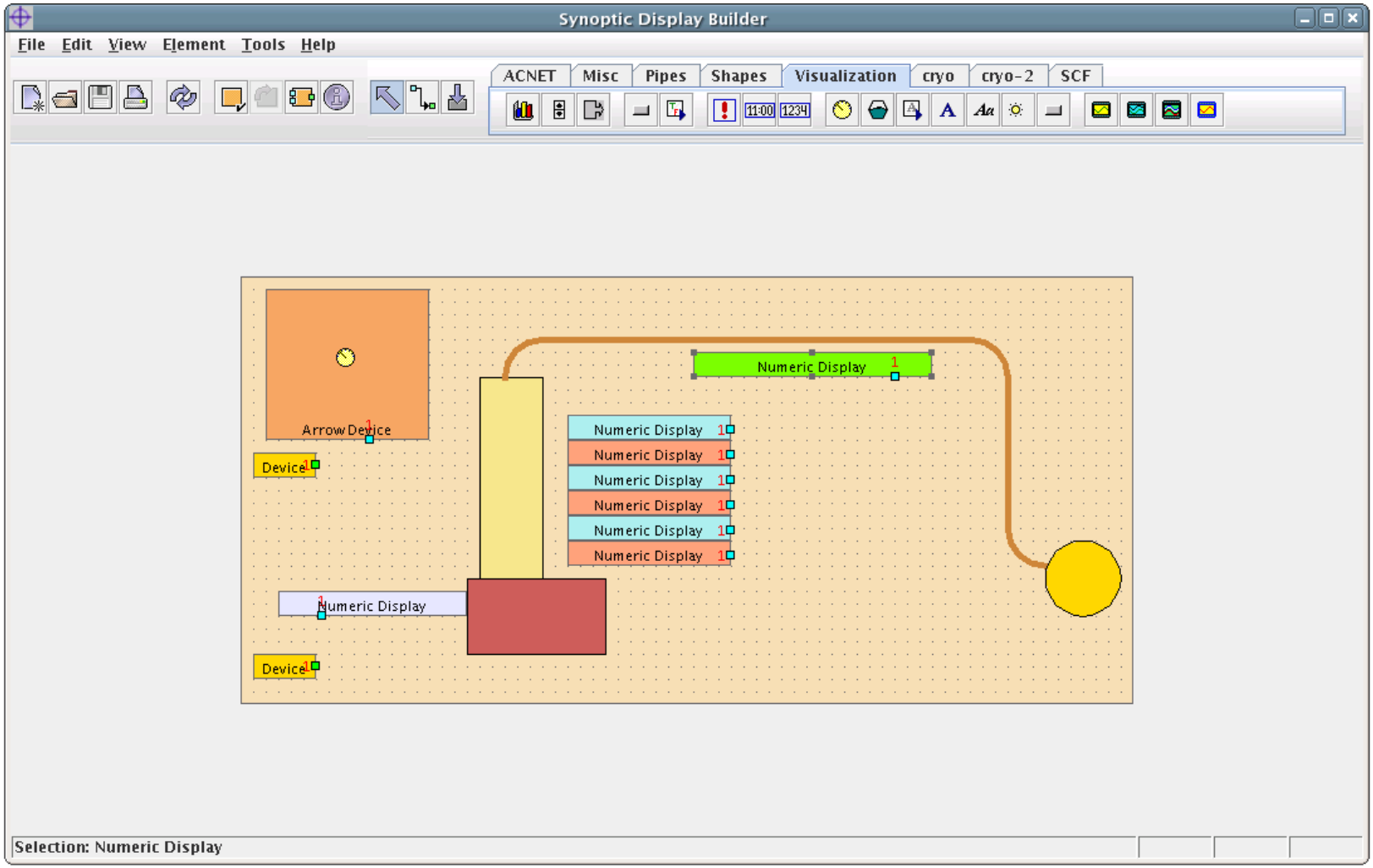
# Demo – Klystron, Step 6, Adding Graphics



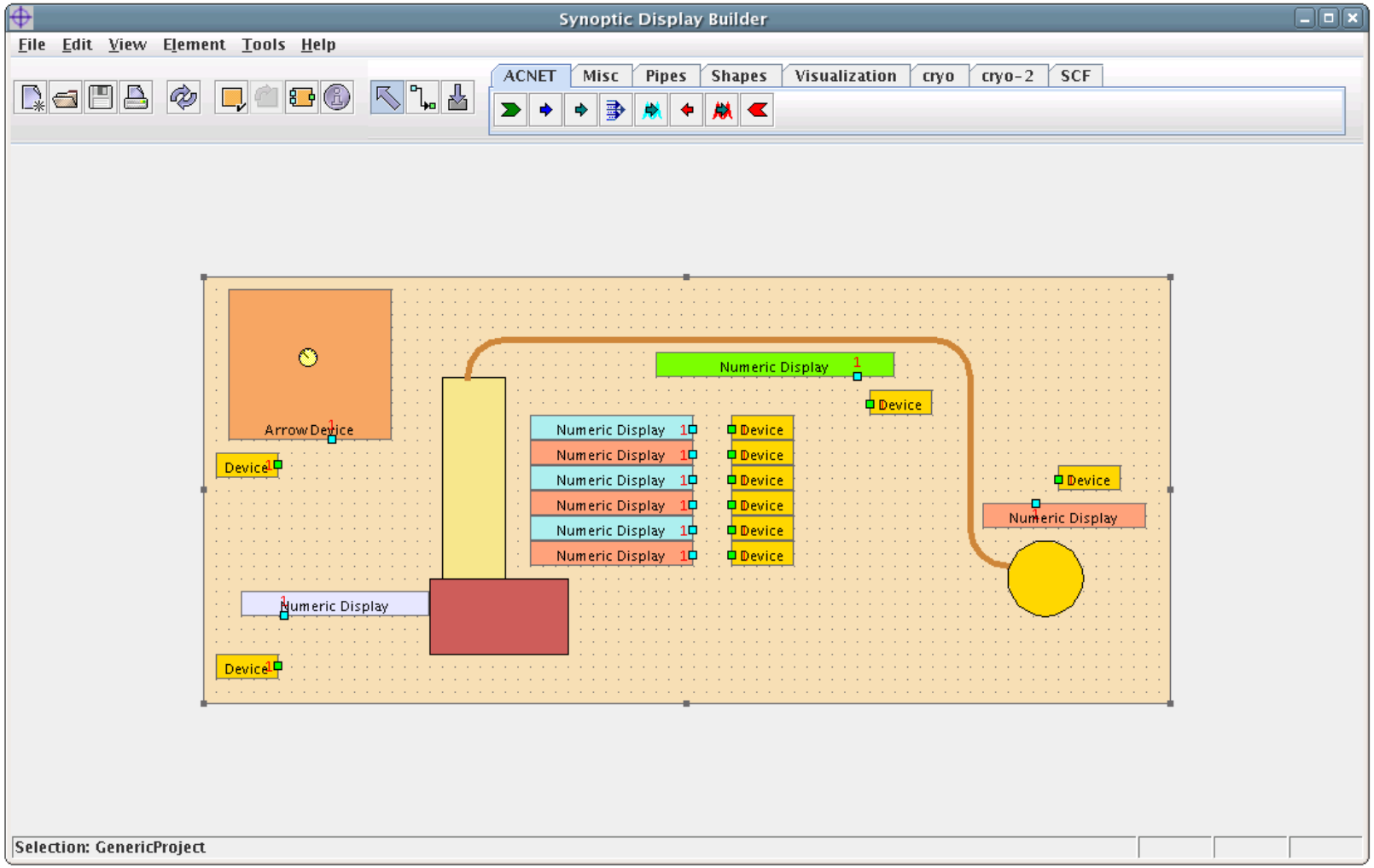
# Demo – Klystron, Step 7, Adding Graphics



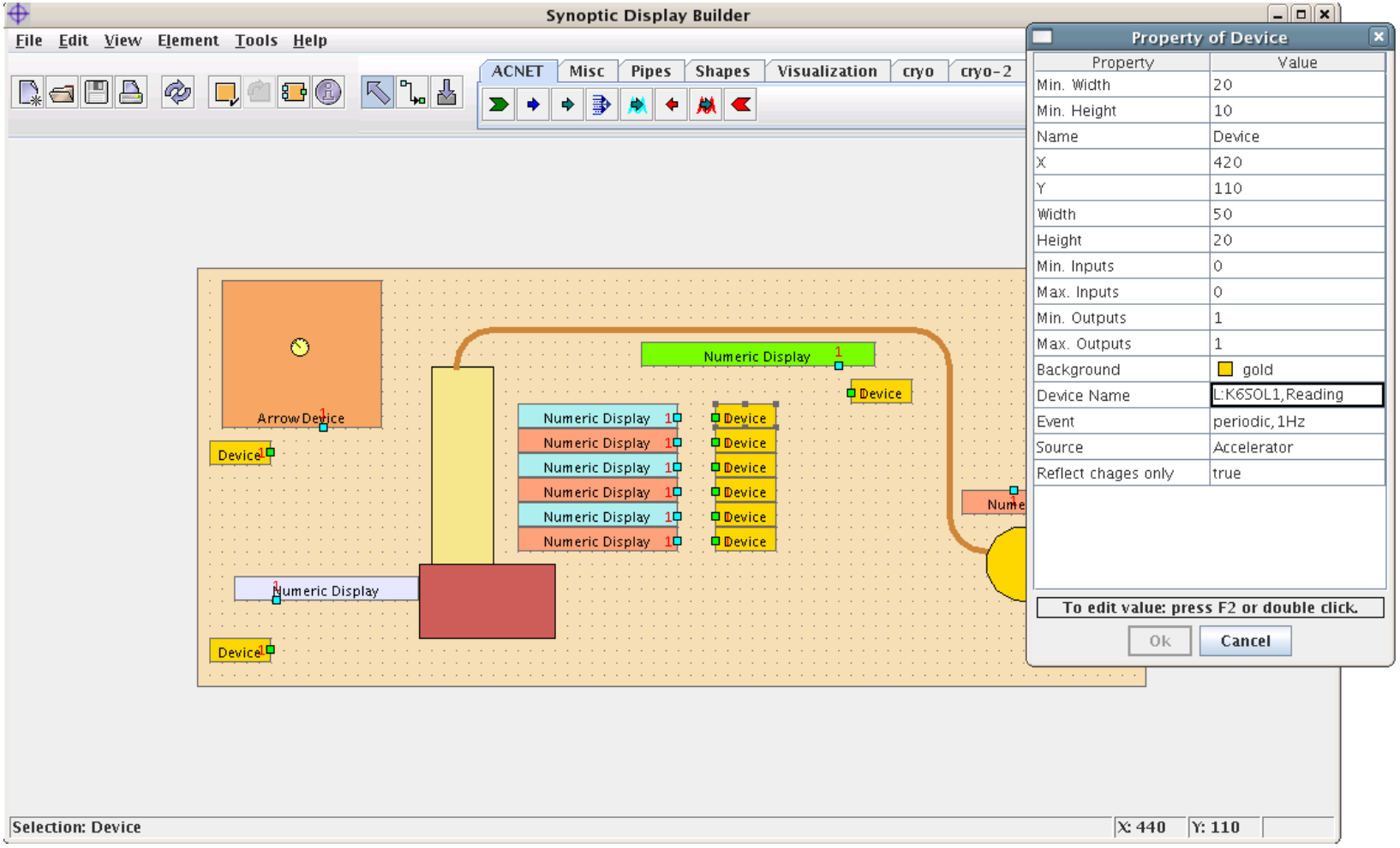
# Demo – Klystron, Step 8, More Components



# Demo – Klystron, Step 9, More Devices

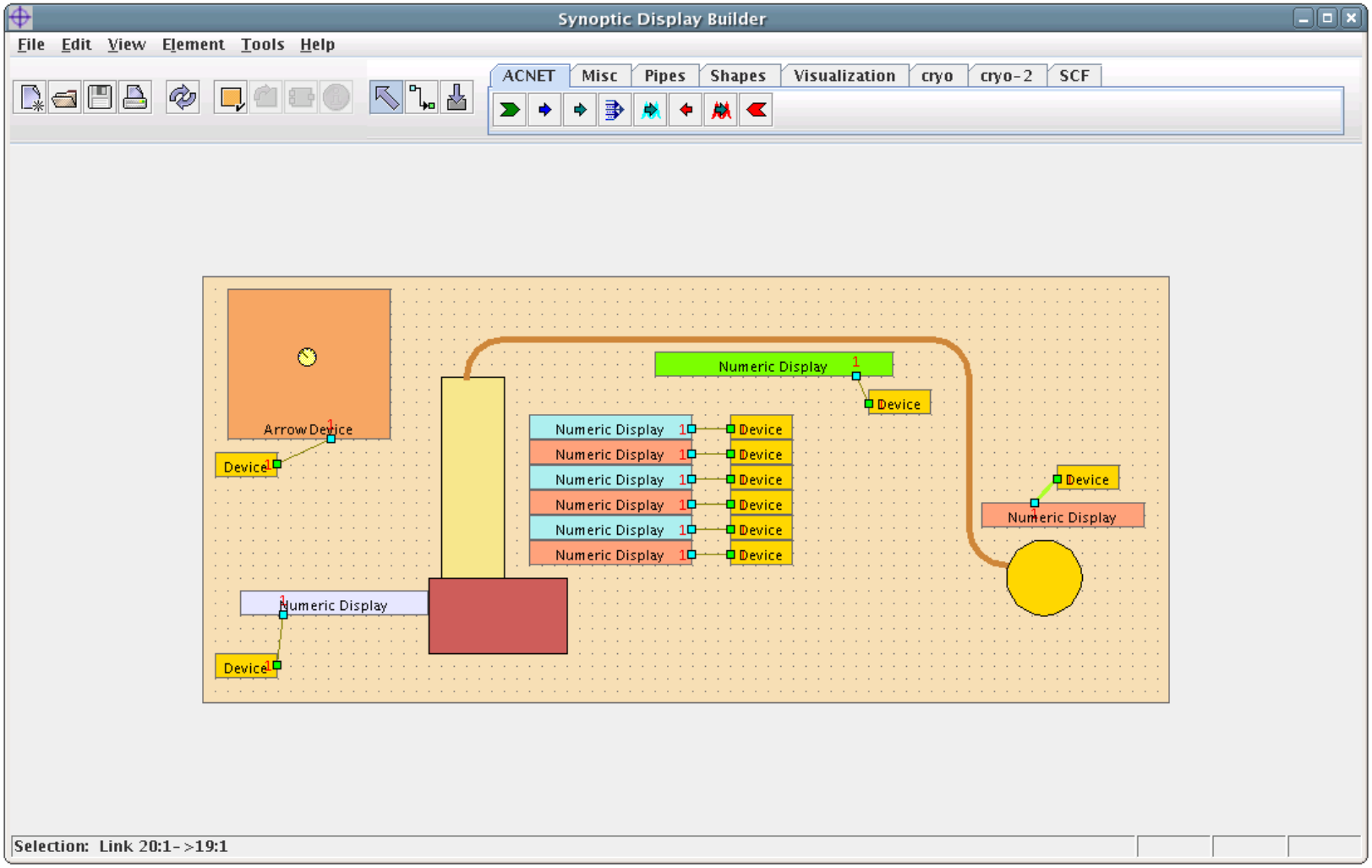


# Demo – Klystron, Step 10, Setting Up Device

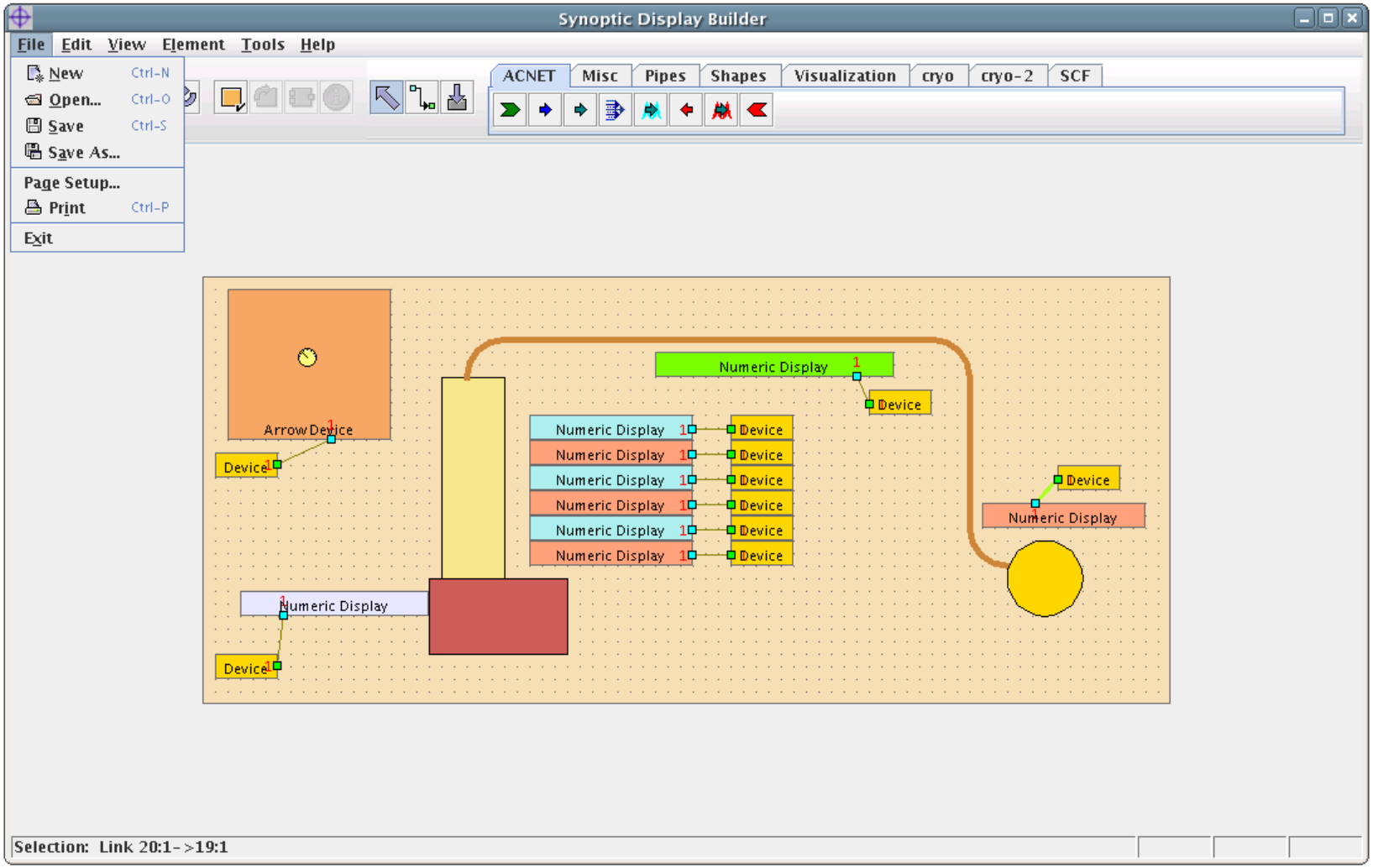


f

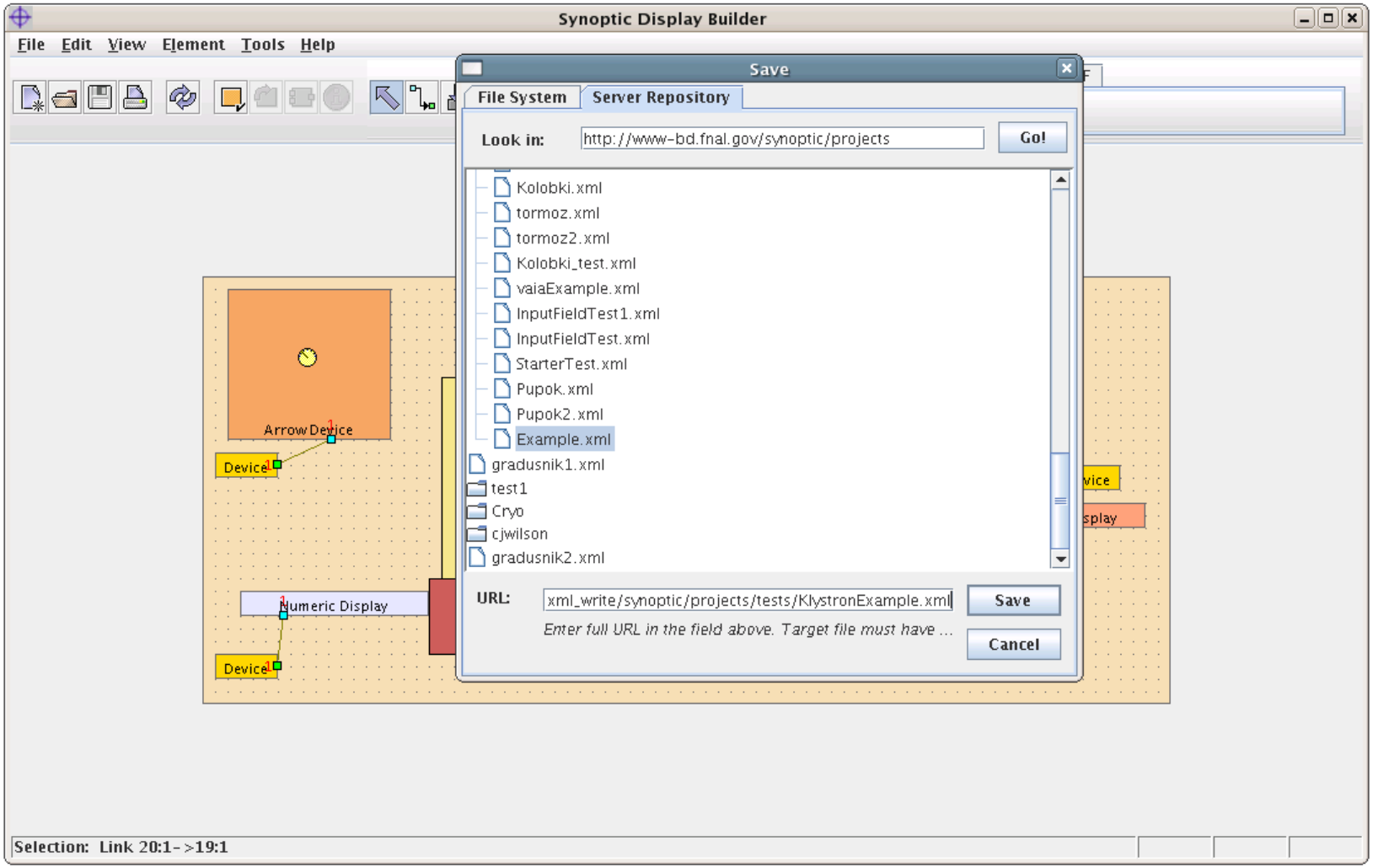
# Demo – Klystron, Step 11, Connecting



# Demo – Klystron, Step 12, Saving Project



# Demo – Klystron, Step 13, Saving Project 2



# Demo – Klystron, Step 14, Projects Repository

Applications Places Desktop USA Mon Mar 26, 8:53:13 AM

AD-FERMI – Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www-bd.fnal.gov/synoptic/tree.html

Red Hat, Inc. Red Hat Network Support Shop Products Training Current gov/fnal/controls/ap...

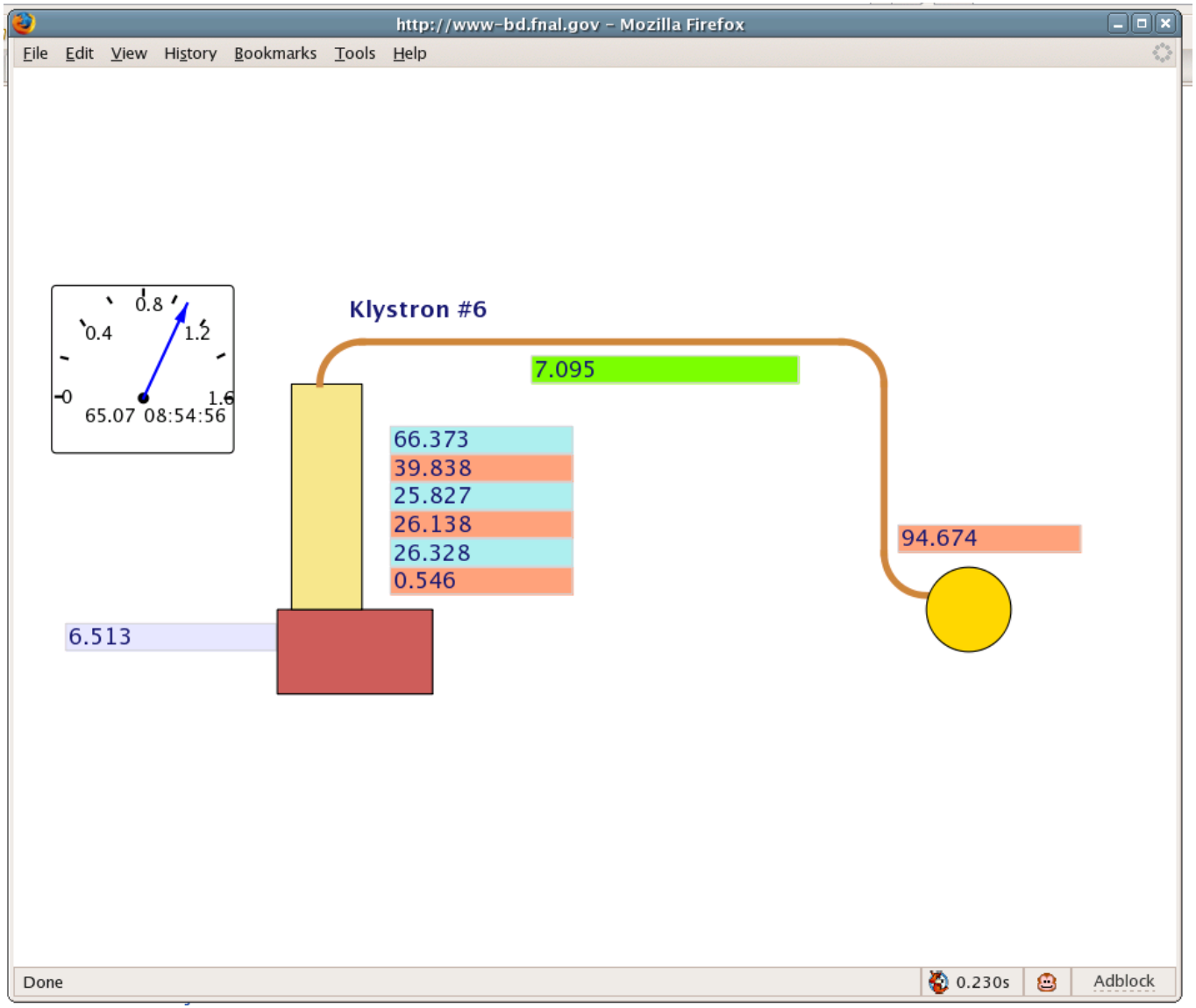
Gmail – Входящие (8) AD-FERMI AD-FERMI

- SlowPlots
- test
- tests
  - BooleanDiscriminator.xml SVG • JPEG • GIF •
  - Button.xml SVG • JPEG • GIF •
  - CompDeviceTest.xml SVG • JPEG • GIF •
  - delay.xml SVG • JPEG • GIF •
  - FB.xml SVG • JPEG • GIF •
  - FSMPanelTest.xml SVG • JPEG • GIF •
  - FSMTest.xml SVG • JPEG • GIF •
  - FTPHistTest.xml SVG • JPEG • GIF •
  - FTPTest.xml SVG • JPEG • GIF •
  - HistogrammTest.xml SVG • JPEG • GIF •
  - JustATest.xml SVG • JPEG • GIF •
  - Knob.xml SVG • JPEG • GIF •
  - MIExpr.xml SVG • JPEG • GIF •
  - NewStatus.xml SVG • JPEG • GIF •
  - Scope.xml SVG • JPEG • GIF •
  - scope2test.xml SVG • JPEG • GIF •
  - Scope4.xml SVG • JPEG • GIF •
  - Scope4Template.xml SVG • JPEG • GIF •
  - SnapScopeTest.xml SVG • JPEG • GIF •
  - SnapTest.xml SVG • JPEG • GIF •
  - Status.xml SVG • JPEG • GIF •
  - Kolobki.xml SVG • JPEG • GIF •
  - tormoz.xml SVG • JPEG • GIF •
  - tormoz2.xml SVG • JPEG • GIF •
  - Kolobki_test.xml SVG • JPEG • GIF •
  - vaiaExample.xml SVG • JPEG • GIF •
  - InputFieldTest1.xml SVG • JPEG • GIF •
  - InputFieldTest.xml SVG • JPEG • GIF •
  - StarterTest.xml SVG • JPEG • GIF •
  - Pupok.xml SVG • JPEG • GIF •
  - Pupok2.xml SVG • JPEG • GIF •
  - Example.xml
  - KlystronExample.xml SVG • JPEG • GIF •
- test1
- Cryo
- cjwilson
- composite.xml SVG • JPEG • GIF •
- D11.xml SVG • JPEG • GIF •
- gradusnik.xml SVG • JPEG • GIF •
- gradusnik_1.xml SVG • JPEG • GIF •
- gradusnik_2.xml SVG • JPEG • GIF •
- internal.xml SVG • JPEG • GIF •

Done 0.322s Adblock

AD-FERMI – Mozilla Firefox [Synoptic Display Builder :: http://www-bd.fnal.gov/javaappli...

# Demo – Klystron, Step 15, Display



f

# Summary

- Drag and Drop is an environment that gives non-control system experts the ability to quickly build controls displays <u>which operates inside the control system framework</u>.
- Drag and Drop:
  - is easy to use
  - sophisticated enough to handle complex displays
  - operates inside the control system framework.
  - uses web browsers and/or Java Web Start (requires no extra software installation)
  - is easily extendible
  - is a mature application
    - Developed in 2001
    - Fermilab Cryogenics department are heavy users
    - Well documented

# Fermilab – Requirements

No requirements –

System is working in production for 5 years,

actively used in Cryogenic Department

(some enhancements possible).

f

# LHC – Requirements

- Administrator's point of view:
  - 1 computer with Tomcat / Jetty web server inside Technical Network – Windows, Linux, Mac or Solaris.
  - One proxy – record from department web-server.
  - File system for storing XML files (component description and projects) and / or class files – 100 Mb.
- Software Engineer point of view:
  - Some refactoring needed to utilize Generics & java.util.concurrent.
  - Device-type Components for JAPC.
  - Runtime Engine and Builder have to use RBAC and to be standard LSA application.
  - Visual Components with CERN DataViewer and other libraries.
- Porting can be done in days or in weeks (in case of refactoring).

f

# Some software details.

- Components:
  - Simple interface: ***init, start, stop*** (lifetime), **properties** (how many inputs and outputs etc),
  - I/O pipes handling – read and write standard objects.
  - Visible components have to be AWT.
  - Some restrictions to be effectively rendered on SVG.
  - All properties are described in XML document – used in Builder and during Runtime.
- Runtime Engine:
  - Just standard (LSA for CERN) application with standard authorization (RBAC for CERN, Kerberos for FNAL).
  - Reads XML project description from web or from file.
  - Read only if started from web.
- Builder – produces projects XML, reads components XML descriptions.