

FESA Graph Editor

Athanasios Topaloudis

3rd Developers@CERN Forum

15/02/2017

Background – What is FESA?

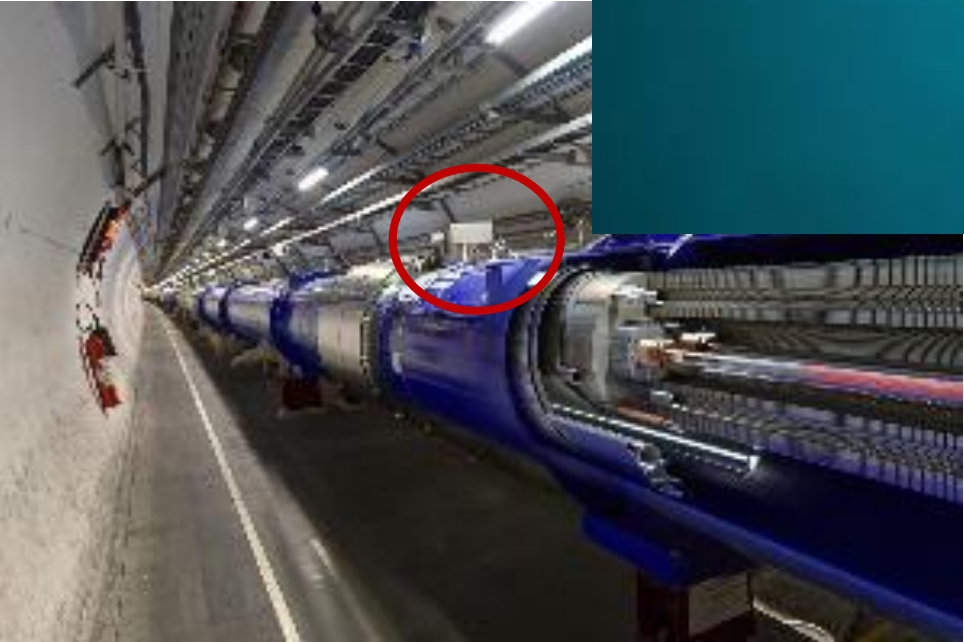
- Front End Software Architecture *is*
- A Framework *that was*
- Developed at CERN *for*
- Real-time SW for front-end computers



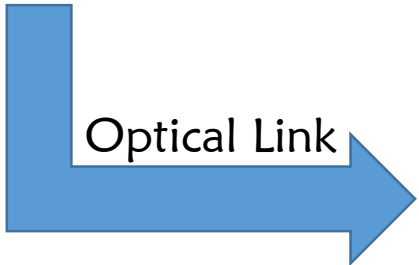
Background – What is FE?

Front End Computer with Acquisition Cards

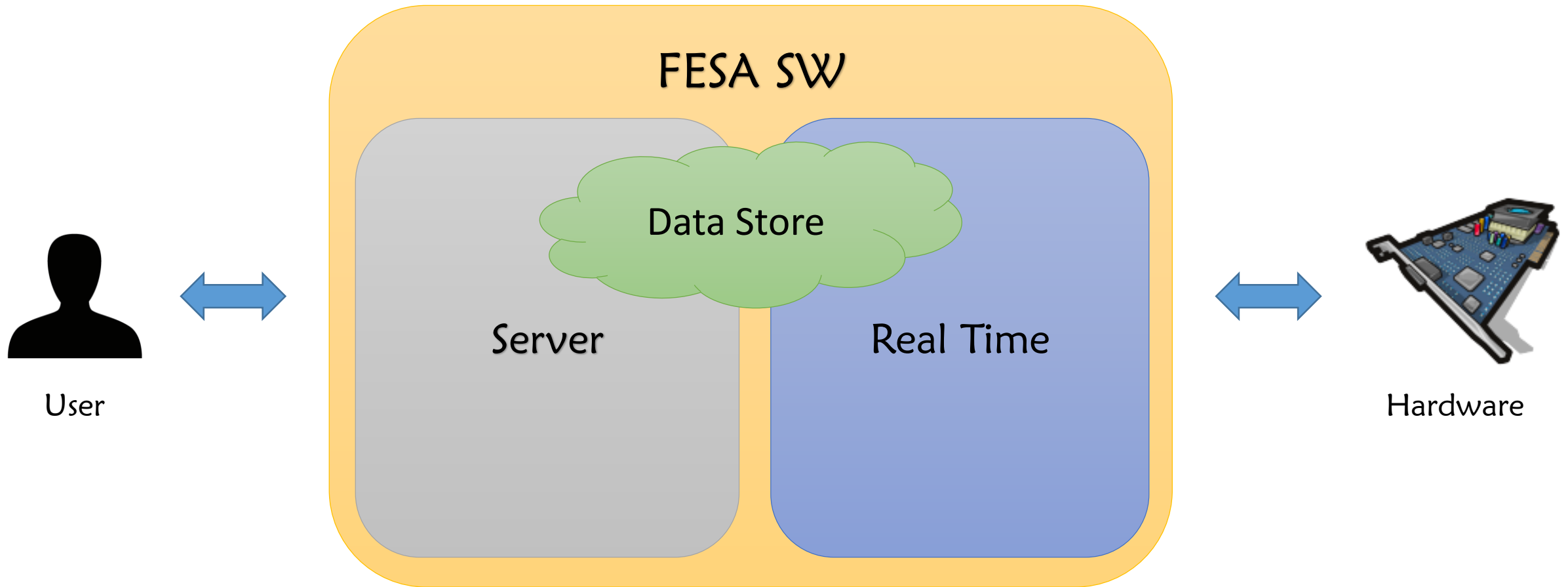
Accelerator Tunnel



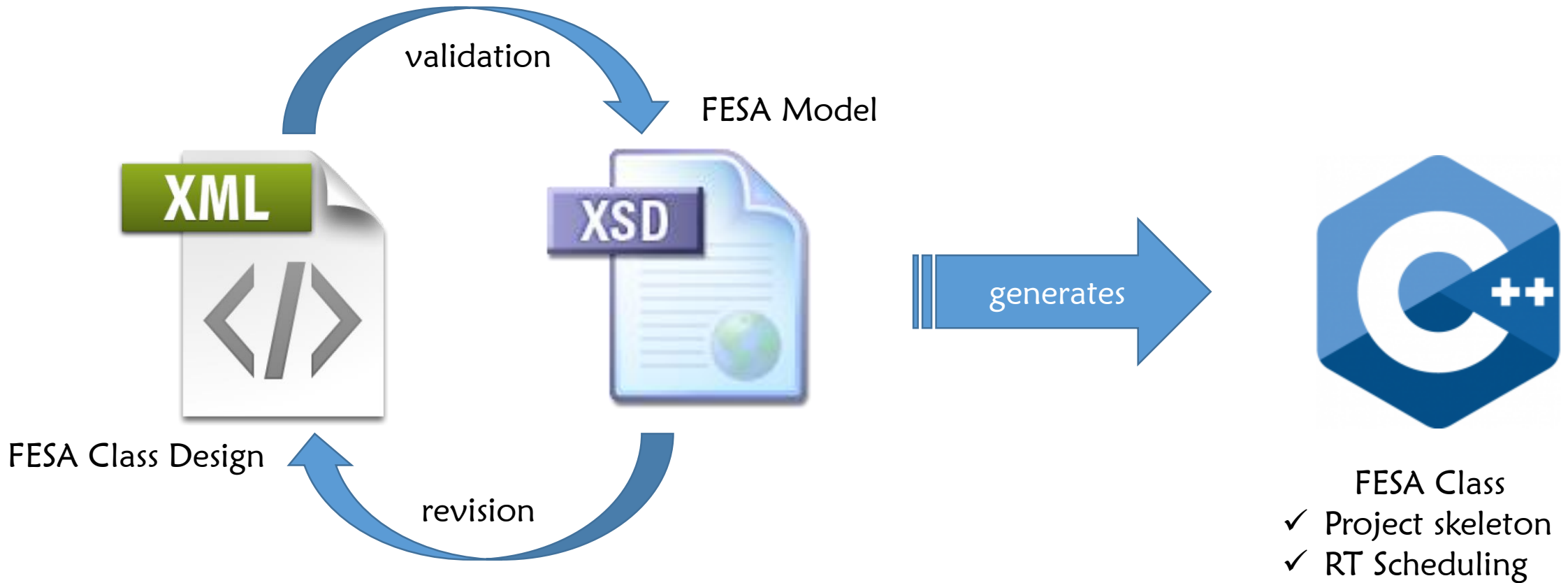
HW Equipment



Background – What is FESA SW?



Background – How does it work?



Background – User Interface

Design Editor

FESA Toolbar

The screenshot displays the Eclipse IDE interface with the FESA Design Editor open. The Design Editor shows a tree view of the design elements, including a 'server-action-ref' element. The Console window shows a red error message: 'FESA Console BPMITLHC.design is not valid'. The Validation Error View at the bottom shows three errors:

Description	Resource	Path	Location	Type
cvc-identity-constraint.4.3: Key 'get-server-a	BPMITLHC.desi	/BPMITLHC/src	line 3414	FESA Problem
FesaRule: Each get-server-action-ref needs t	BPMITLHC.desi	/BPMITLHC/src	Unknown	FESA Problem
FesaRule: Server action cannot be orphan ((/	BPMITLHC.desi	/BPMITLHC/src	Unknown	FESA Problem

C++ Project Explorer

Console

Validation Error View

Background – Summary FESA

- Essential framework for real-time SW development
- C++ code generation based on XML document
- RT scheduling
- Comes as an Eclipse plugin

- Impossible overview of the SW
- Connection among components is not obvious
- Design editing can be cumbersome

- Let's do something about it...



Users – Tools Improvement – Round 1



Design Analysis



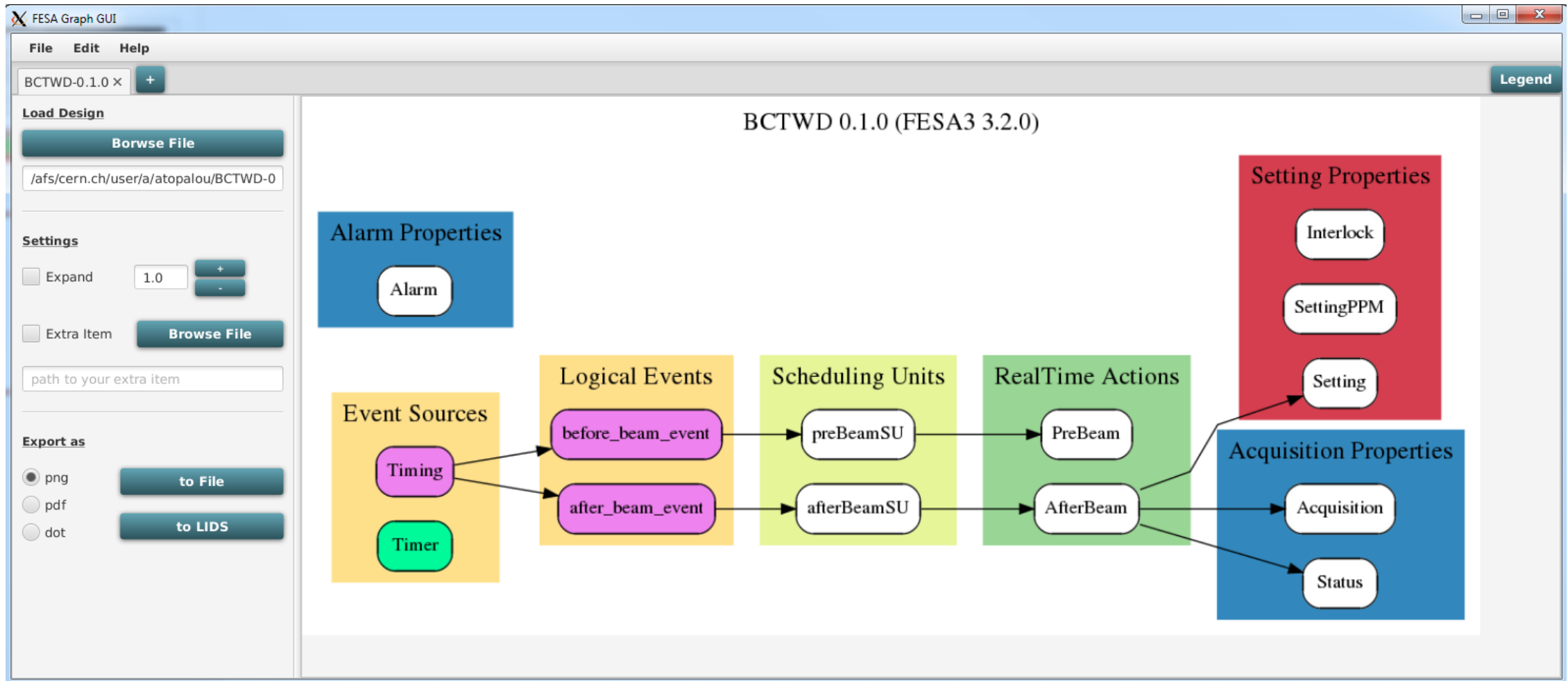
Graph Library



Output



Graph Viewer – Design Overview

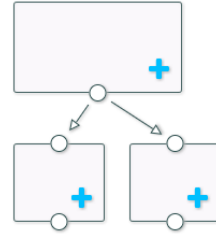


Users – Tools Improvement – Round 2



Dynamic Model

- ✓ XML parsing
- ✓ DOM tree
- ✓ Use of FESA model
- ✓ Design Fragmenting (Server, RT, Data Store)
- ✓ Command Stack (*undo/redo*)



Graph Editor Library

- ✓ MVC
- ✓ CSS custom view
- ✓ Drag 'n Drop
- ✓ Large Graph Support
- ✓ Zoom

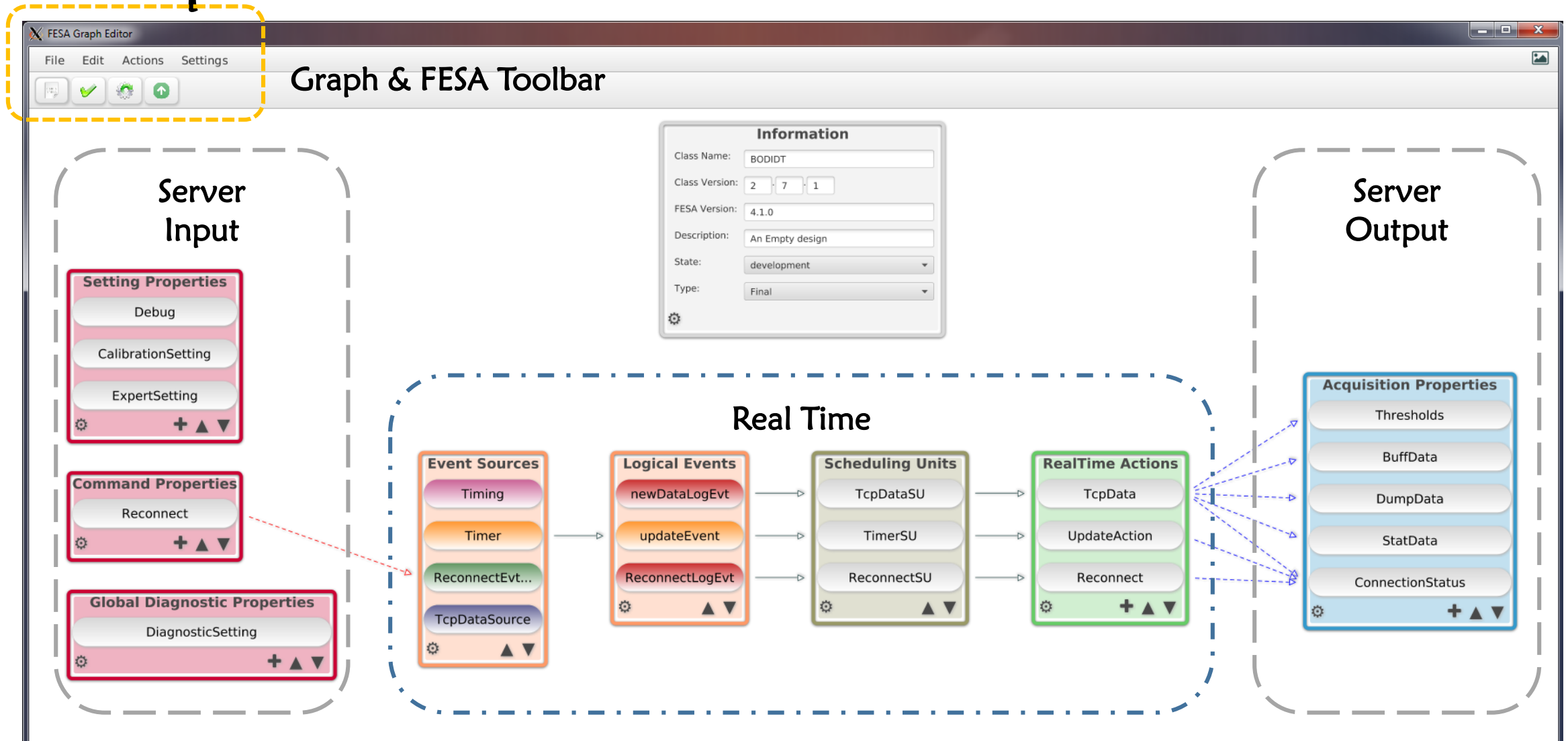


FESA API

- ✓ FESA Model
- ✓ Validation
- ✓ Code Generation
- ✓ Design Editing



Graph Editor – Overview



Graph Editor – Data Store

BPMITLHC.design

Node	Content
?? xml	version="1.0" encoding="U1
equipment-model	(xmlns:xsi=http://www.w3.o
xsi:noNamespaceSchemaLocation	file:/nfs/cs-ccr-nfs1/vol30/loc
information	BPMITLHC 1.8.2 FESA 2.3.0
ownership	BE/BI
interface	Reset, IntLckTrigger, XpocTrig
custom-types	NOTIFICATION_UPDATE, DIA
data	dumpGroup, acqClkSource, t
device-data	dumpGroup, acqClkSource, t
configuration	dumpGroup, acqClkSource, t
setting	clockPhase, nbOfOrbitTurns,
acquisition	acqSensitivity, horMinDump
field	(acqSensitivity: SENSITIVITY
data-consistent	false
multiplexed	false
name	acqSensitivity
custom-type-scalar	(data-type-name-ref=SENSI
field	(horMinDumpAdc: int16_t, d
field	(horMaxDumpAdc: int16_t, d
field	(verMinDumpAdc: int16_t, d
field	(verMaxDumpAdc: int16_t, d
field	(dabTemperature: int16_t[M/
field	(dabTemperatureGeneric: int
acq-stamp-field	(captAcqStamp: int64_t, data

Shared memory of BCTFI

Device Global Timing-Domain Custom types Builtin types

Configuration Setting Acquisition

Acquisition field

name	persistent	multiplexed	data-consistent	data-type
gateStamp		false	false	Data type: array Type: float Cst name ref: MAX_NB...
totalIntensity		true	false	Data type: array Type: float Cst name ref: MAX_NB...
lastTotIntLow		false	false	Data type: scalar Type: float

Acquisition cycle name field

name	multiplexed	data-consistent
cycleName	true	false

Cycle stamp field

name	multiplexed	data-consistent	data-type	description
cycleStamp	true	false	Data type: scalar	

Cywin/XX

multiplexed

Data consistent

Data type

Data type: array
Type: float
Cst name ref: MAX_NB_OF_MEAS

Unit

Default

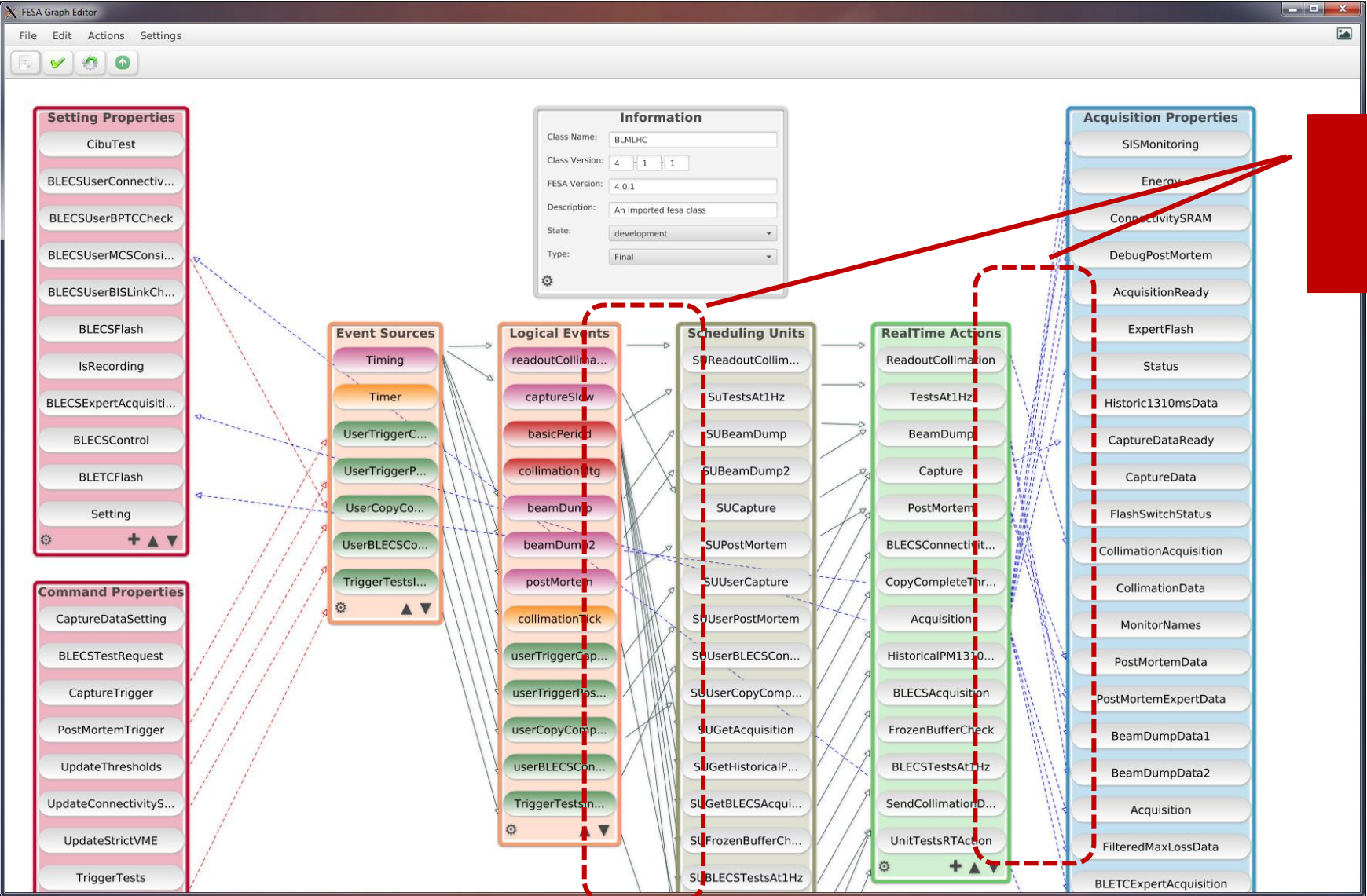
Meaning

Description

Total Number of Charges per Measurement

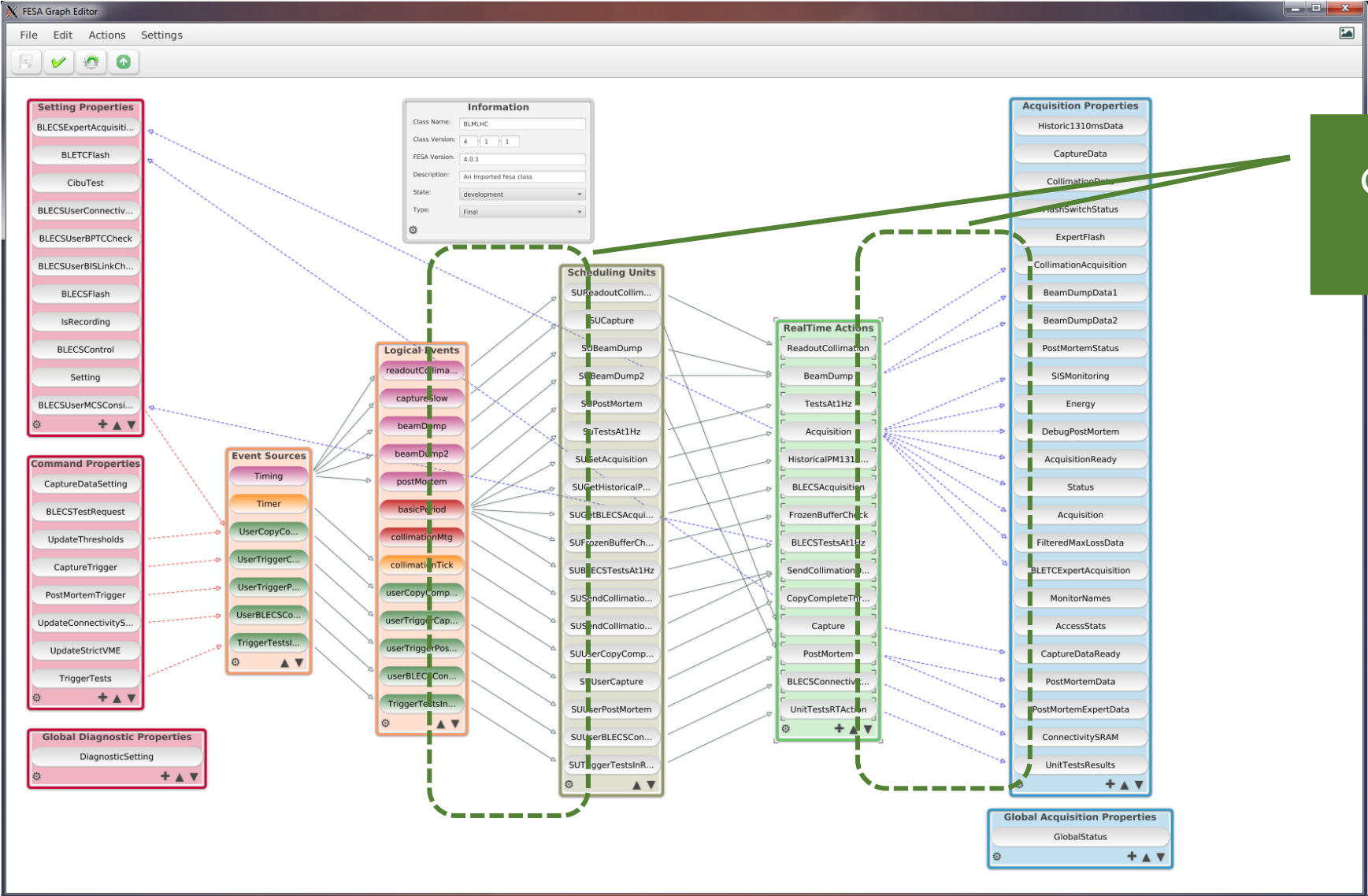
Ok

Graph Editor – Design Overview



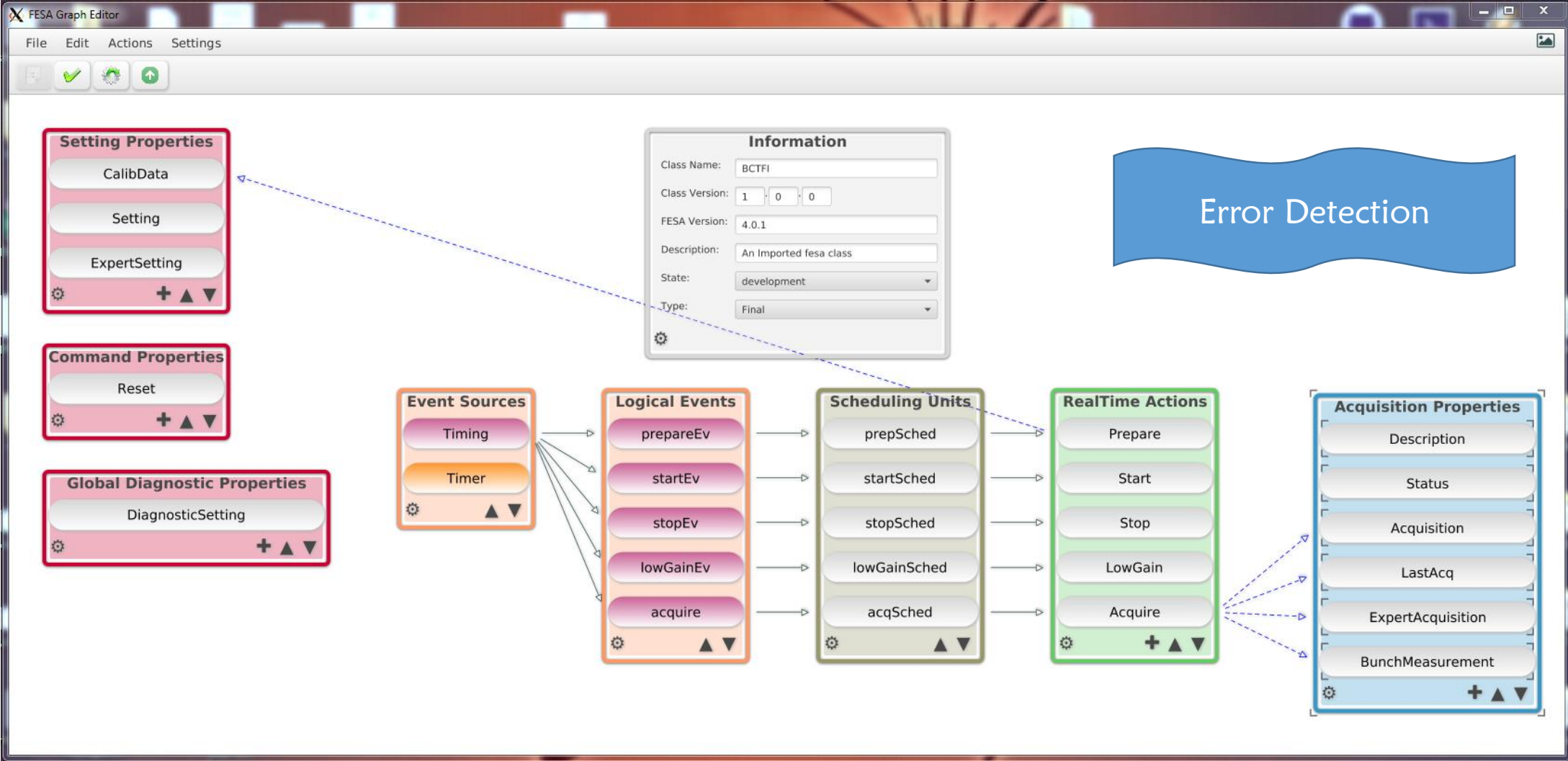
Complex & Messy

Graph Editor – Design Overview

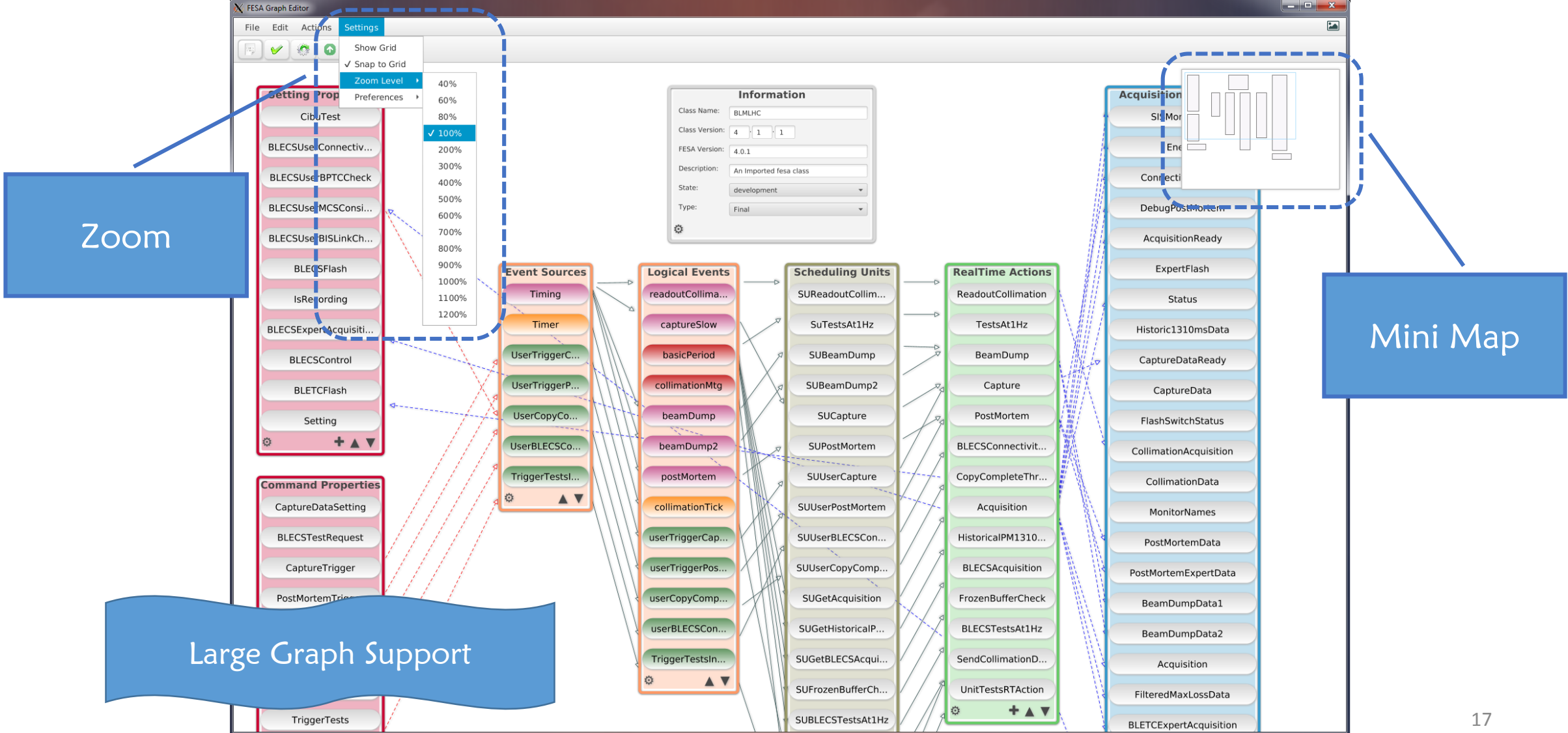


Complex but Clean

Graph Editor – Components Connection



Graph Editor – Additional Functionality



Graph Editor – Customization

The image displays the FESA Graph Editor interface, which includes a central graph area with nodes like 'Setting Properties', 'Information', 'Event Sources', 'RealTime Actions', and 'Acquisition Properties'. A blue callout box labeled 'CSS Based View' points to the graph. Two 'Preferences' dialog boxes are overlaid on the graph. The top dialog shows color selection for 'Event Type' (generic, timing, timer, on-demand) with color swatches. The bottom dialog shows the 'Background' section with the 'grayscale' checkbox checked and highlighted by a dashed blue box. Other dialog boxes include 'Acquisition Properties' and 'Global Diagnostic Properties'.

Graph Editor – Design Editing

The image displays the FESA Graph Editor interface. On the left, there are three property panels: **Setting Properties** (with CalibData, Setting, ExpertSetting), **Command Properties** (with Reset), and **Global Diagnostic Properties** (with DiagnosticSetting). In the center, a flow diagram shows the sequence: **Event Sources** (Timing, Timer) → **Logical Events** (prepareEv, startEv, stopEv, lowGainEv, acquire) → **Scheduling Units** (prepSched, startSched, stopSched, lowGainSched, acqSched) → **RealTime Actions** (Prepare, Start, Stop, LowGain, Acquire, NewRTAction). An **Information** panel for class BCTFI is also visible. On the right, an **Insert Real Time Action** dialog box is open, showing fields for Real-time action name (NewRTAction), Scheduling unit name (empty), Logical event name (dropdown with prepSched, startSched, stopSched, lowGainSched, acqSched), Logical event type (empty), and Logical event source name (empty). The **acqSched** option is selected in the dropdown. Below the dialog, a zoomed-in view of the RealTime Actions panel shows the **NewRTAction** component highlighted with a blue circle. A blue banner at the bottom left contains the text "Adding FESA Components".

Graph Editor

Wrap Up:

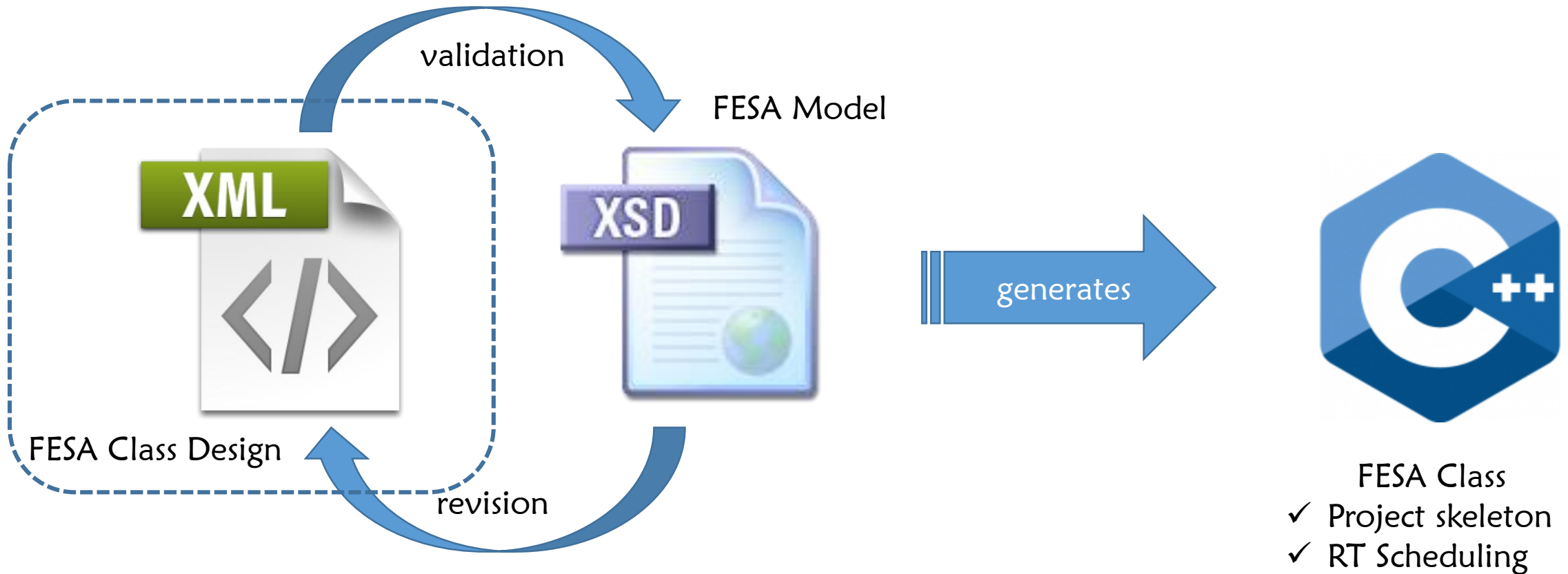
- Standalone JavaFX application
- Graphical abstraction of low level XML
- FESA SW Design overview – Easy Documentation!
- Fragmentation of the FESA design
- Made by Users... for Users

Future:

- Included in the plugin
- Included in the release process
- Fully editable design



Graph Editor – The whole picture



Graph Editor – The whole picture

