SOA Tools Platform
Ganymede Simultaneous Release

SOA Tools 1.0 Release Review

SCA
SERVICE CREATION
ENTERPRISE INTEGRATION
POLICY EDITOR
SOA SYSTEM
BPEL 2 JAVA
INTERMEDIATE MODEL
BPMN

- Multi-component project indicating significant diversity in problem area
- Active committers from IONA, Intalio, Obeo, INRIA, Engineering, SOPERA, Scapa, Bull
- All new code provided under EPL
- All 3rd party (non-EPL) approved by EMO
- Graduation from incubation state

Developer Community

- Community diversity includes active developers from Intalio, IONA, Obeo, Scapa, Engineering, SOPERA, INRIA, Bull
- Total 16 active committers and 3 contributors from 8 companies
  - *up from 11 committers, 2 contributors at Europa*
- Promoting diversity by soliciting contributions from organizations in the SOA industry
  - *added 5 new organizations over the last year*

IP Considerations

- IP Process followed for significant contributions
- Licenses and about files in place as per Eclipse Development Process
- Third party library usage confined to approved Orbit bundles
- IP Log available at http://www.eclipse.org/stp/development/ip_log.php

Consumer Community

- Elements of the SOA Tools Project are being used in products and distributions from (at least) the following organizations

ENGINEERING
INGEGNERIA
INFORMATICA

INTALIO

IONA

eteration
source of open

SOA TOOLS PLATFORM PROJECT

Significant Updates since Europa

- Four new components contributed

SOA TOOLS PLATFORM SCA

SOA TOOLS PLATFORM ENTERPRISE INTEGRATION

SOA TOOLS PLATFORM POLICY EDITOR

SOA TOOLS PLATFORM INTERMEDIATE MODEL

- One component ‘retired’ as inactive

SOA TOOLS PLATFORM CORE FRAMEWORKS

- Five components at 1.0 release

SOA TOOLS PLATFORM SCA

SOA TOOLS PLATFORM POLICY EDITOR

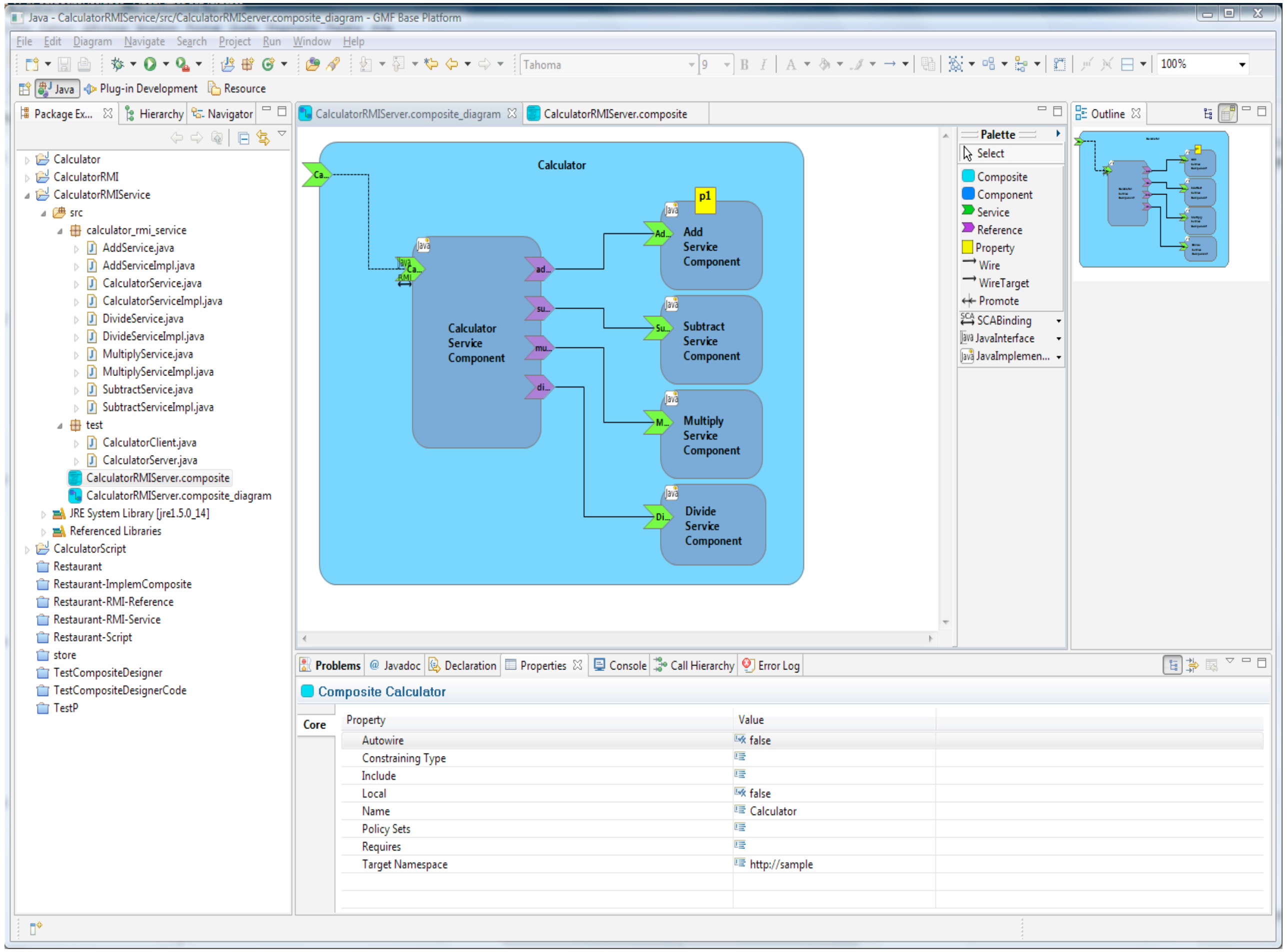
SOA TOOLS PLATFORM INTERMEDIATE MODEL

SOA TOOLS PLATFORM BPEL 2 JAVA

SOA TOOLS PLATFORM BPMN

SCA Composite Designer 1.0

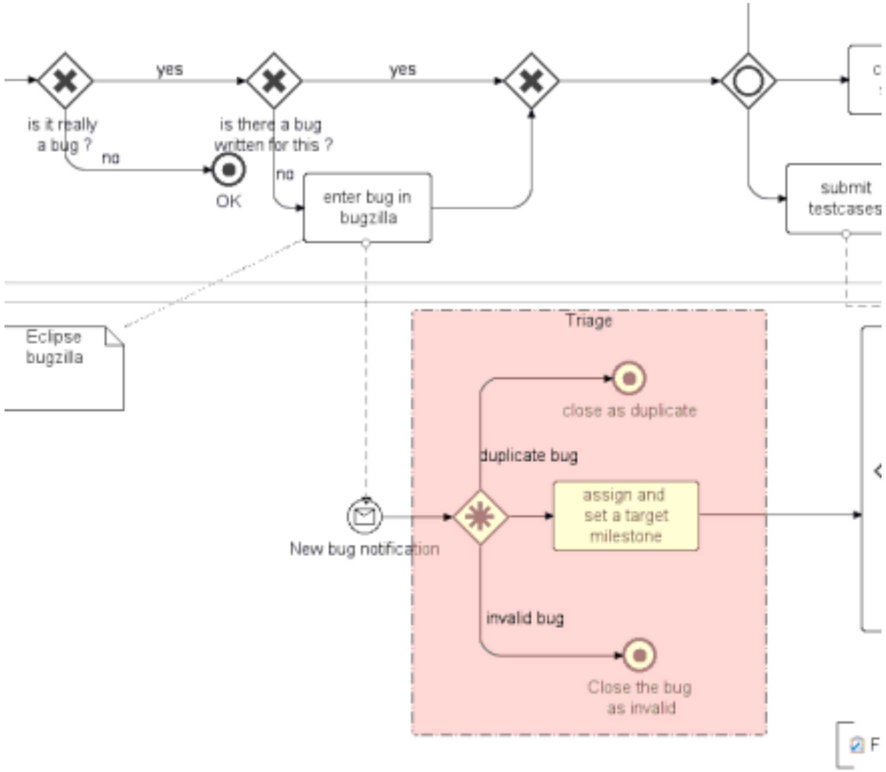
- Graphical editor to construct SCA assembly files
- Complies with Open SOA specifications 1.0
- Tuscany SCA elements are supported
- Works with Tuscany and Frascati runtimes



Composite Calculator

Property	Value
Autowire	false
Constraining Type	
Include	
Local	false
Name	Calculator
Policy Sets	
Requires	
Target Namespace	http://sample

BPMN Modeler 1.0



- Graphical editor to build business processes
- Complies with the OMG's BPMN 1.1 standard
- Designed for extensibility

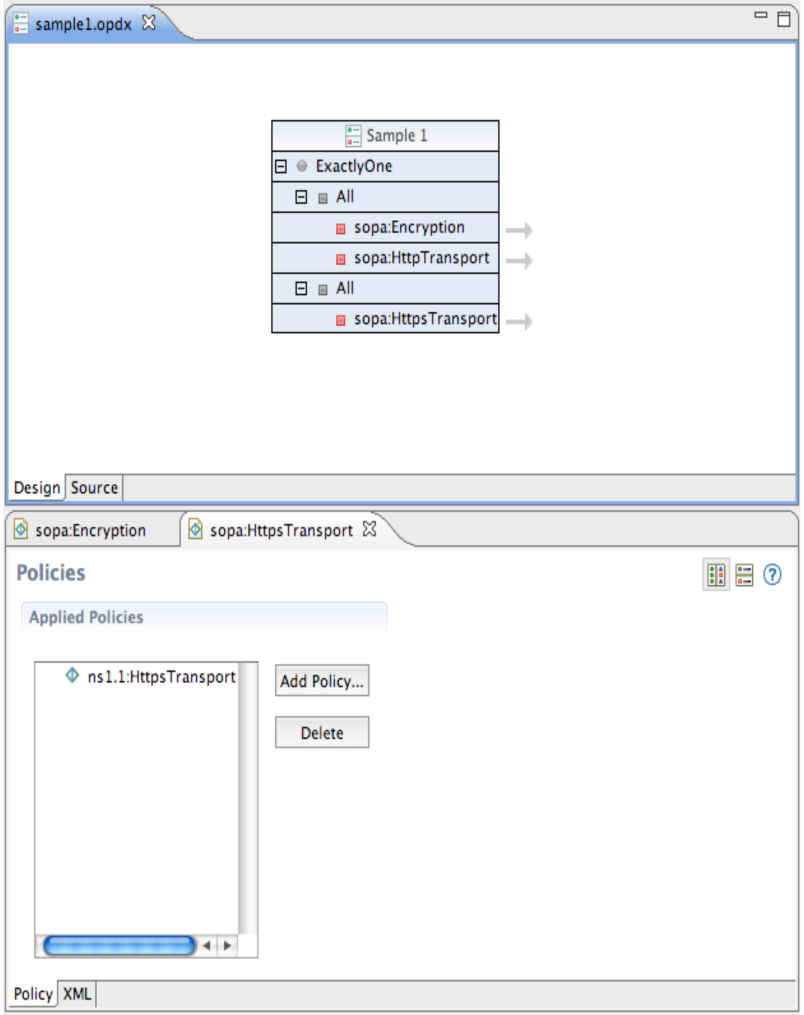
Intermediate SOA Model 1.0

- **Bridges different SOA platforms in STP**
  - Workflow / process: e.g. BPMN, BPEL
  - Architecture specification: e.g. SCA, EID, JBI
  - Service Creation: e.g. JAX-WS, Policy Specification

- **Facilitates interoperability between editors**
  - Provides a central SOA conceptual bridge
  - Avoids duplication of data
  - Minimizes amount of transformation code
  - Facilitates code generation from a variety of sources

WS-Policy Editor 1.0

- Provides support for editing, validation of WS-Policy files
- Uses standard W3C WS-Policy approaches
- Extensible with custom policy assertions
- Includes policy details editor and assertions editor



- Sample 1
  - ExactlyOne
    - All
      - sopa:Encryption
      - sopa:HttpTransport
    - All
      - sopa:HttpsTransport

BPEL 2 Java Compiler 1.0

- **Provides tool to translate BPEL to Java**
  - Upgraded to be BPEL 2.0 compliant
  - Aimed at embedded applications and test choreography

- **Server integration framework**
  - Allows adaptation to common ESB and other approaches service deployment
  - APIs have been stable
  - Tutorials available – http://www.eclipse.org/stp/b2j

Service Creation Framework 0.9

- Delivers a framework for the construction of Java-annotation based Services

- Includes code generation framework for WSDL-based services

- In Ganymede, exemplar code provided for
    - JAX-WS services with JAX-WS RI and CXF 2.1
    - SCA Java services with Tuscany 1.1/1.2

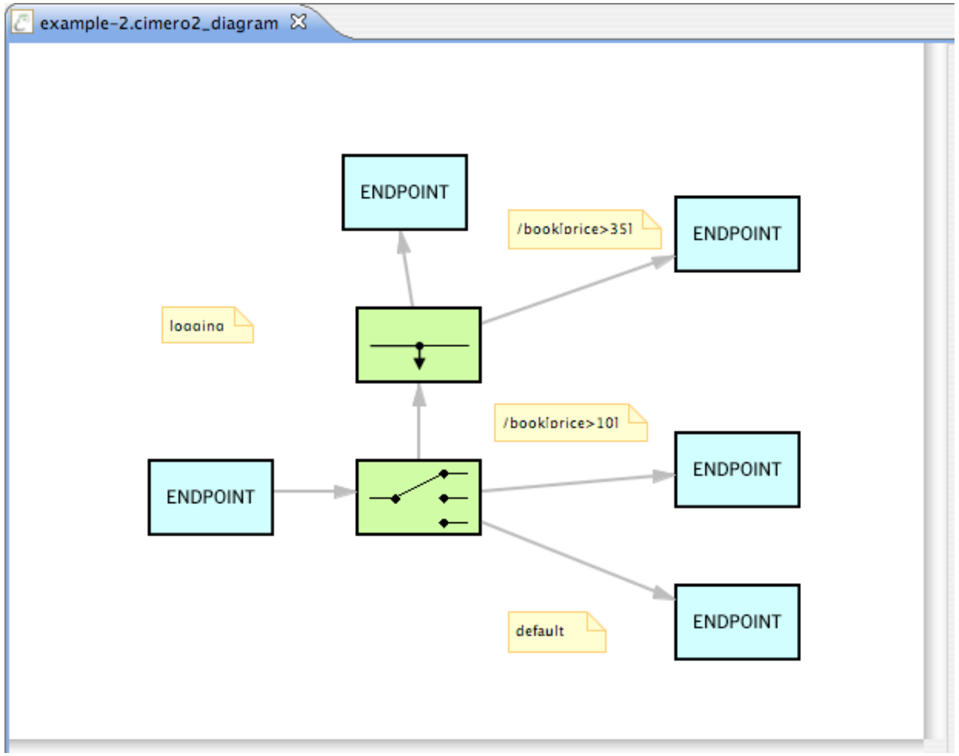
- Future plans include REST/WADL based services, smarter code generation

SOA System 0.8

- Provides a flexible and composable deployment model for build artifacts
- For Ganymede, this component successfully integrates with WTP Server Framework
  - Exemplar code using Tomcat
- Original sponsor Sybase has ceased operation with this component – IONA has taken stewardship for the moment
- Future plans include component viability assessment

Enterprise Integration Designer 0.8

- Diagrammatic editor for Enterprise Integration Patterns
- Provides a code and configuration generation framework
- Provides a runtime environment extension framework
- Initial runtimes ServiceMix3, PeTaLs, Camel



Scheduling

- STP overall project to reach 1.0 Ganymede
  - Ganymede milestones achieved

- Expectation that all components would reach 1.0 for Ganymede
  - 5 of 8 reached that goal
  - others still need some development before 1.0

Plans post-Ganymede

- Components at 1.0 to become full sub-projects with independent release schedules
- Proposal of an Incubator for new and experimental capabilities
- Integration project to tie together the technologies representing different technology domains
- Individual components have specific plans above and beyond these project-level needs

Thanks!

- Please provide us with your feedback, requirements and issues
- Enhancement requests, bugs to Bugzilla
  - https://bugs.eclipse.org/bugs/enter_bug.cgi?product=STP
- Questions and comment to STP newsgroup
  - news://news.eclipse.org/eclipse.stp
- Contributions, queries about project structure to PMC
  - stp-pmc@eclipse.org

Individual Component Contributions

The following slides have been submitted by individual components to supplement the high-level overview material

STP Intermediate Model

Ganymede Review

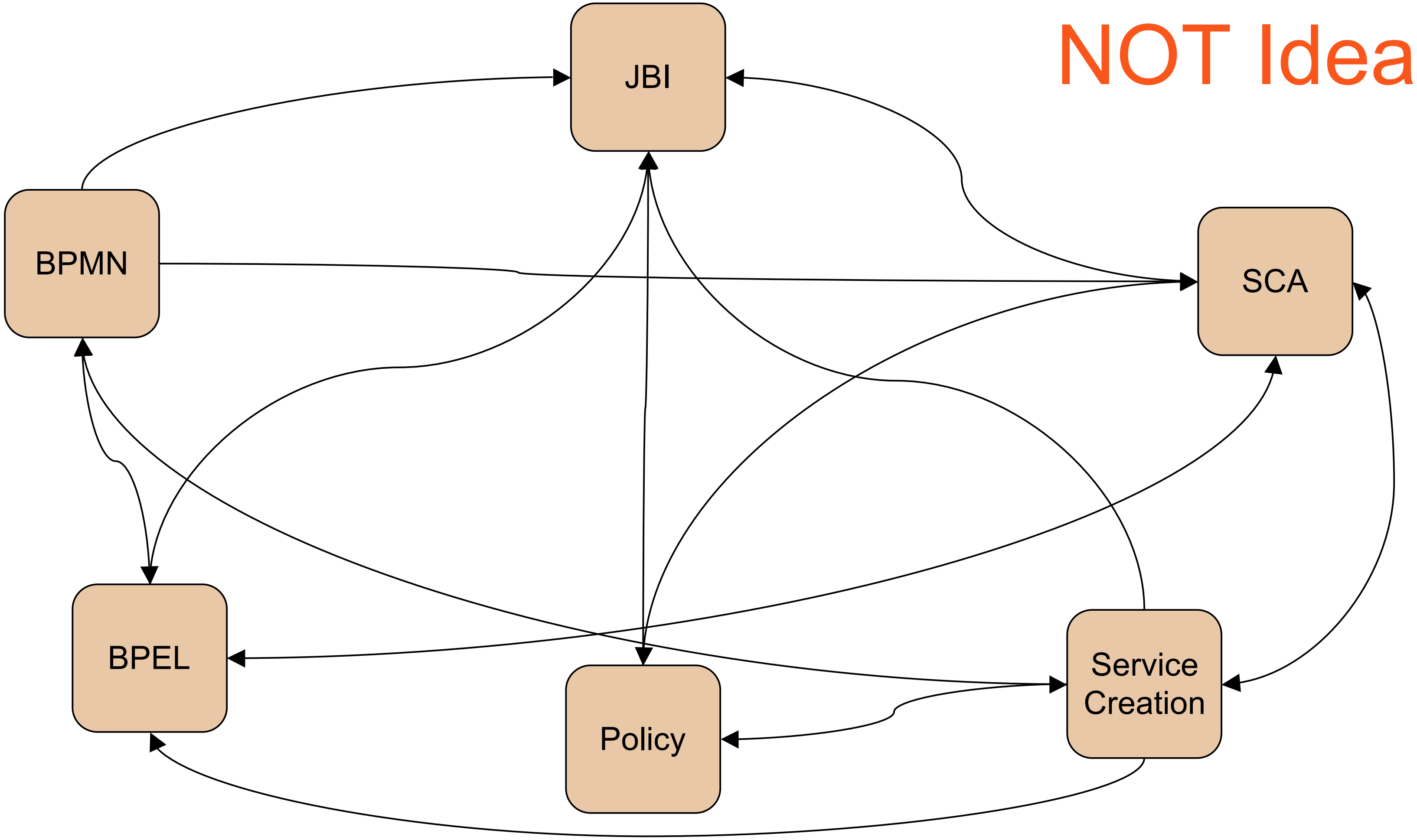
Intermediate Model Overview

- Bridges different SOA platforms in STP
  - Workflow / process: e.g. BPMN, BPEL
  - Architecture specification: e.g. SCA, EID, JBI
  - Service Creation: e.g. JAX-WS, Policy Specification
- Facilitates interoperability between editors
  - Provides a central SOA conceptual bridge
  - Avoids duplication of data
  - Minimizes amount of transformation code
  - Facilitates code generation from a variety of sources
- Initial Contribution: INRIA (FR) and Engineering (IT)
  - STP component: **org.eclipse.stp.model**
  - EMF model plugins + transformation plugins
  - Used in the Spagic 2.0 SOA Suite from Engineering

Large Variety of SOA Tools and Platforms

- Eclipse SOA Tools Platform Project hosts several SOA editors
  - BPMN
  - BPEL
  - Policy
  - EID
  - JAX-WS
  - SCA
  - JBI
- Different roles use different editors / platforms
- Information duplication is inevitable when moving across editors
  - Same service can be seen in different spaces
  - Processes, dependencies, compositions must be defined repeatedly

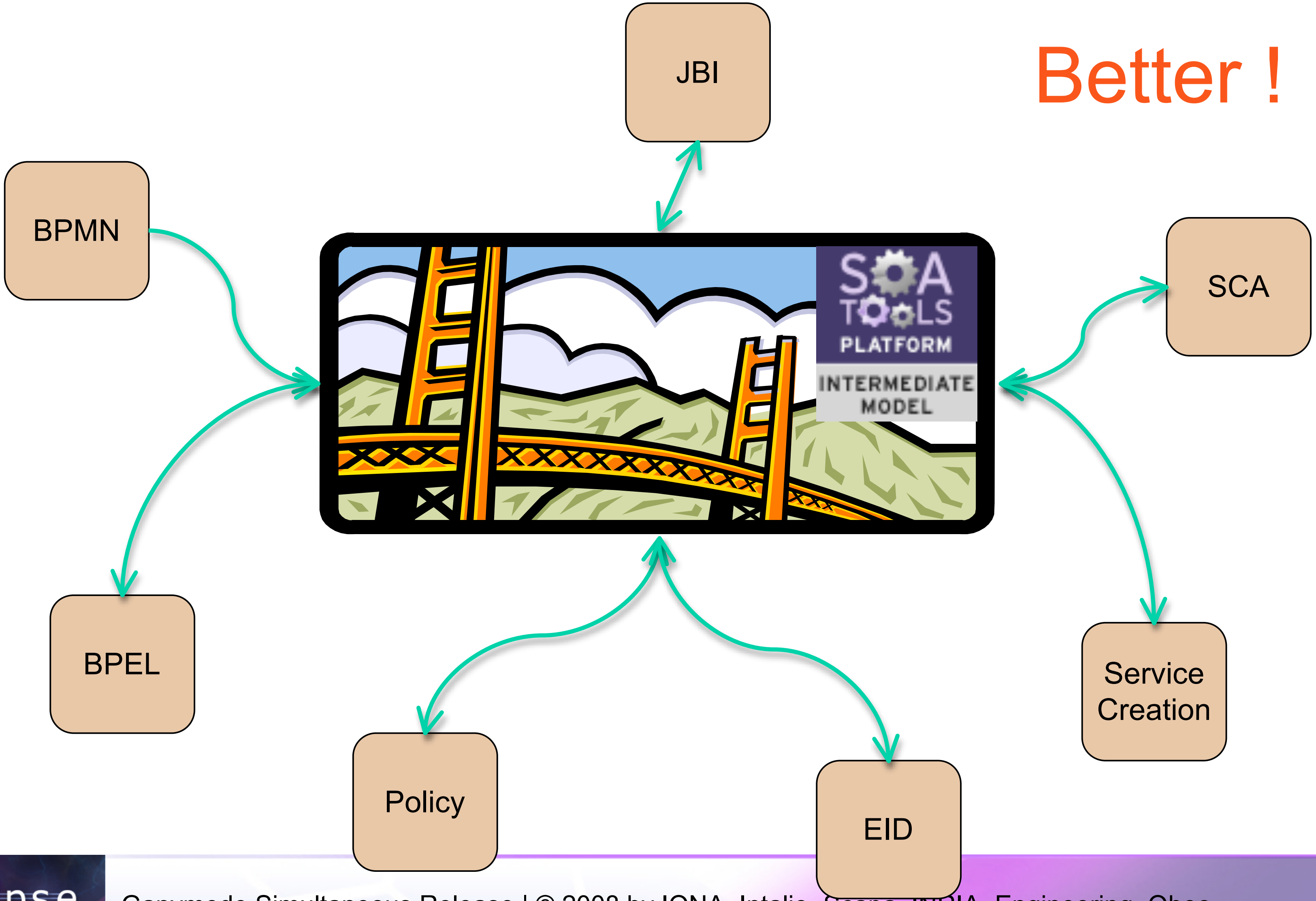
Integrating SOA Editors - A First Take



NOT Ideal !

Bridging SOA Editors with STP-IM

Better !



Current Status

- Passed the IP process for existing plugins
- Improves the overall functionality of STP
- Available Transformations (in the repository)
  - BPMN to STP-IM
  - SCA to STP-IM (basic functionality)
  - STP-IM to SCA (basic functionality)
- Runtime extension capabilities
- Used in production in Spagic 2.0
  - Additional JBI support
  - BPEL support (to be transferred to STP)

Upcoming Contributions

- New transformation plugins
  - STP-IM to BPEL (immediate release)
  - STP-IM ←→ EID
  - STP-IM ←→ Service Creation
  - Eclipse JWT → STP-IM

- Improvements and Extensions to existing plugins

- Updated documentation: wiki and guides

- Google Summer of Code - Juan Cadavid
  - Funded project to contribute to STP-IM
  - Transformations: BPMN-BPEL-SCA-EID
  - Will investigate declarative approaches

Get and Contribute to STP-IM

- Location:
  - HTTP://www.eclipse.org/stp/im
  - SVN://…/stp/org.eclipse.stp.model

- Plugin Structure:
  - org.eclipse.stp.im (STP-IM model)
  - org.eclipse.stp.im.runtime.* (e.g. bpel, jbi)
  - org.eclipse.stp.im.in.* (e.g. bpmn)
  - org.eclipse.stp.im.tool.in.* (e.g. bpmneditor)

BPMN Modeler

- A graphical editor to draw BPMN 1.0 and 1.1 diagrams.

- Active committers:
  - Hugues Malphettes, component lead
  - Antoine Toulme

BPMN modeler - contact

- By email (always add stp-dev@eclipse.org in CC):
  - hmalphettes@intalio.com, atoulme@intalio.com
- Newsgroup
  - org.eclipse.stp on news.eclipse.org
- IRC
  - #eclipse-stp on irc.freenode.net

BPMN modeler - goals

- Provide a graphical editor to draw business processes.
- Use the Business Process Modeling Notation (BPMN)
- No interpretation into executable artifacts
  - Transformation into other formats possible
    - STP-IM
    - JWT

BPMN modeler - architecture

- 100% eclipse
  - The BPMN modeler is built on an EMF model
  - It depends on the GMF and GEF frameworks
- 4 plugins
  - org.eclipse.stp.bpmn: the model
  - org.eclipse.stp.bpmn.diagram: the diagram editor
  - org.eclipse.stp.bpmn.edit: the model edit framework (providing labels and images)
  - org.eclipse.stp.bpmn.validation: the validation plugin

BPMN modeler - specification

- References are the OMG (Object Management Group) BPMN specifications
  - Respect the graphical notation as much as possible
  - Specification compliant, with a few holes
    - Data Object artifact label is misplaced
    - No decoration for ad hoc subprocesses

BPMN modeler – validation framework

- Validation based on a builder
  - When saving the diagram is introspected
  - Errors show on the shapes that are not respecting the specification rules

BPMN Modeler – since Europa

- Added support for BPMN 1.1
- Messages with pools are supported
- Internationalization of plugins
  - Recently added to the Babel website
- Many improvements in the UI and the routing of connections
  - Added shadows
  - Connection labels may have a background and a border color.

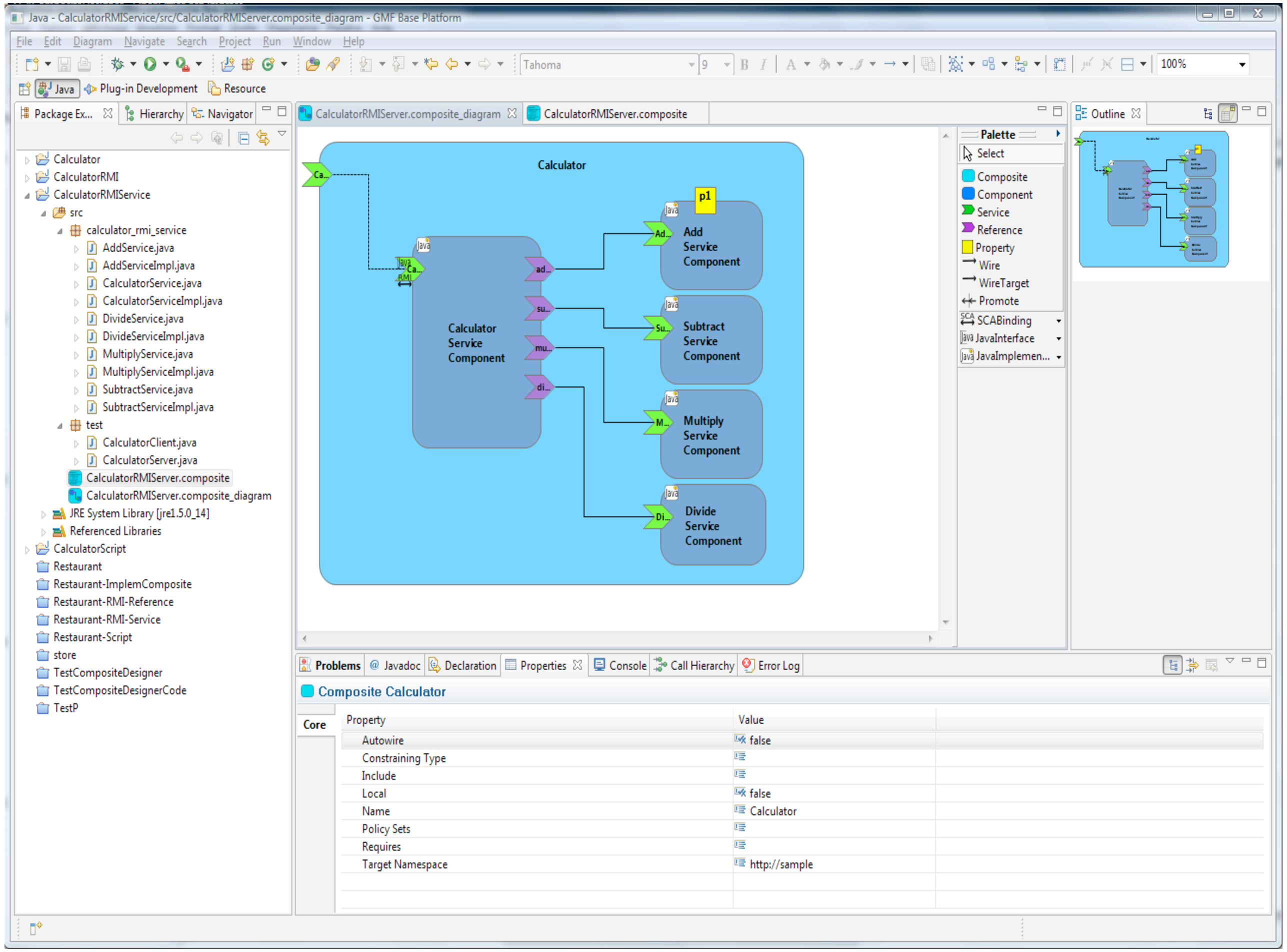
BPMN modeler – future plans

- Be fully specification compliant
- Fix the copy/paste mechanism, with the help of the GMF team
- Make sure undo/redo operations are well implemented
- Package as an EPP build, if possible

SCA Composite Designer
Ganymede Simultaneous Release

SCA Composite Designer (Incubation)

- Graphical editor to construct SCA assembly files
- Complies with Open SOA specifications 1.0
- Tuscany SCA elements are supported
- Works with Tuscany runtime and Frascati runtime



Composite Calculator

Property	Value
Autowire	false
Constraining Type	
Include	
Local	false
Name	Calculator
Policy Sets	
Requires	
Target Namespace	http://sample

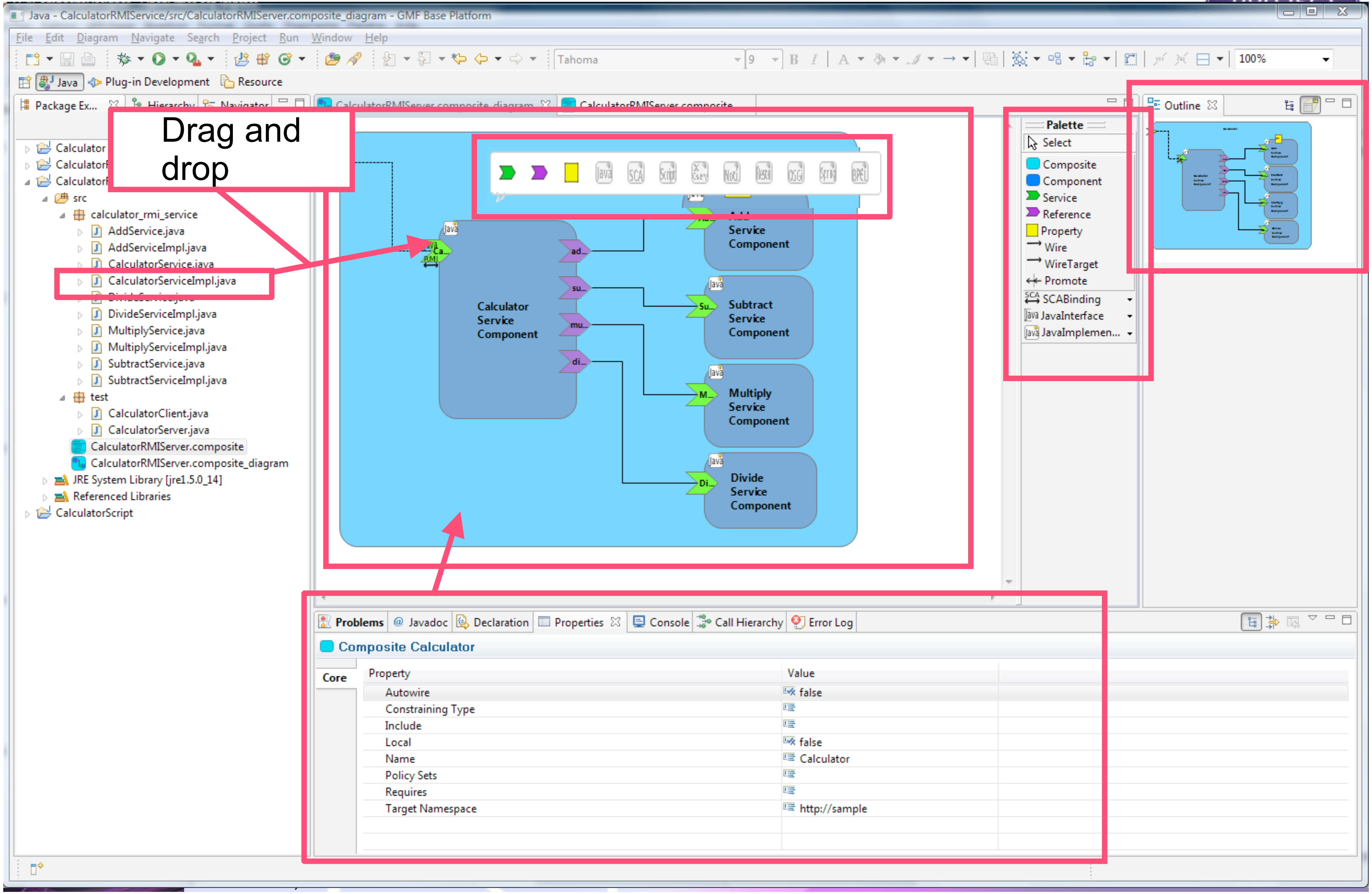
What is the SCA Composite Designer?

- The STP/SCA Composite Designer component provides a graphical (GMF) development environment to construct SCA composite applications
- Sub components
  - SCA Composite Meta Model: Ecore SCA meta model from OSOA specifications 1.0
  - SCA Editors: tree editor + textual editor
  - SCA Composite Designer: graphical editor for SCA composites
- Committers
  - Stéphane Drapeau (Stephane.Drapeau@obeo.fr) – Leader
  - Etienne Juliot (Etienne.Juliot@obeo.fr)

Features

- SCA meta model (Ecore file generated from Open SOA XSD schemas) that can be used standalone
  - http://wiki.eclipse.org/STP/SCA_Component/SCA_Composite_Meta_Model
- Additional validation rules
  - http://wiki.eclipse.org/STP/SCA_Component/SCA_Composite_Meta_Model#Additional_validation_rules
- Drag and drop of implementations, interfaces, and bindings of the workspace
- Link with developer editors (Java, BPEL, ...)
- SCA elements defined by Apache Tuscany are supported

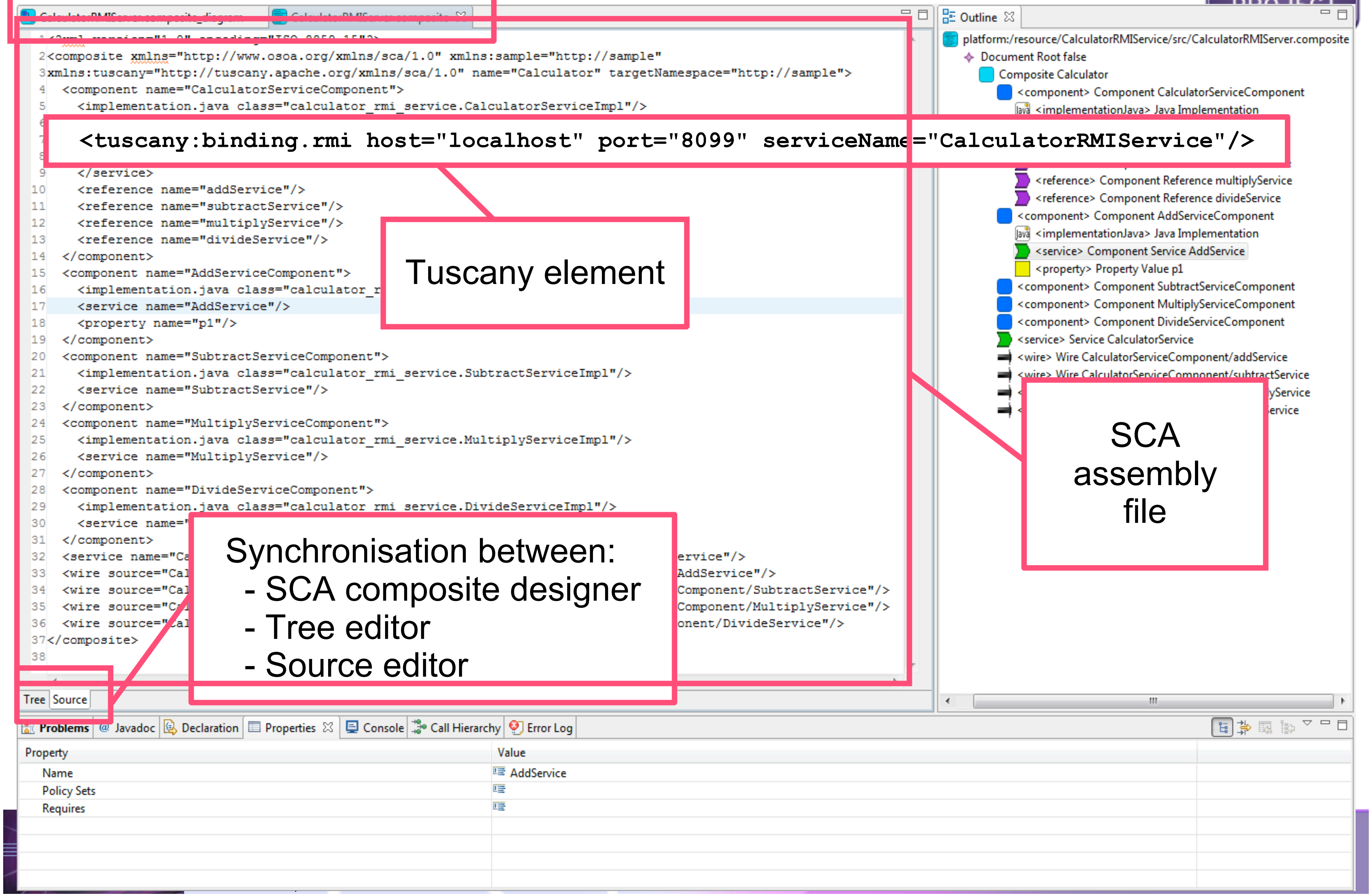
Features



Composite Calculator

Core	Property	Value
	Autowire	false
	Constraining Type	
	Include	
	Local	false
	Name	Calculator
	Policy Sets	
	Requires	
	Target Namespace	http://sample

Features



```
<?xml version="1.0" encoding="ISO-8859-15"?>
<composite xmlns="http://www.osoa.org/xmlns/sca/1.0" xmlns:sample="http://sample"
xmlns:tuscany="http://tuscany.apache.org/xmlns/sca/1.0" name="Calculator" targetNamespace="http://sample">
  <component name="CalculatorServiceComponent">
    <implementation.java class="calculator_rmi_service.CalculatorServiceImpl"/>
    </service>
    <reference name="addService"/>
    <reference name="subtractService"/>
    <reference name="multiplyService"/>
    <reference name="divideService"/>
  </component>
  <component name="AddServiceComponent">
    <implementation.java class="calculator_r
    <service name="AddService"/>
    <property name="p1"/>
  </component>
  <component name="SubtractServiceComponent">
    <implementation.java class="calculator_rmi_service.SubtractServiceImpl"/>
    <service name="SubtractService"/>
  </component>
  <component name="MultiplyServiceComponent">
    <implementation.java class="calculator_rmi_service.MultiplyServiceImpl"/>
    <service name="MultiplyService"/>
  </component>
  <component name="DivideServiceComponent">
    <implementation.java class="calculator_rmi_service.DivideServiceImpl"/>
    <service name="
  </component>
  <service name="Ca                                              ervice"/>
  <wire source="Cal                                              AddService"/>
  <wire source="Cal                                              Component/SubtractService"/>
  <wire source="C                                                Component/MultiplyService"/>
  <wire source="Cal                                              onent/DivideService"/>
</composite>
```

```
<tuscany:binding.rmi host="localhost" port="8099" serviceName="CalculatorRMIService"/>
```

Tuscany element

platform:/resource/CalculatorRMIService/src/CalculatorRMIServer.composite
- Document Root false
  - Composite Calculator
    - <component> Component CalculatorServiceComponent
      - <implementationJava> Java Implementation
      - <reference> Component Reference multiplyService
      - <reference> Component Reference divideService
    - <component> Component AddServiceComponent
      - <implementationJava> Java Implementation
      - <service> Component Service AddService
      - <property> Property Value p1
    - <component> Component SubtractServiceComponent
    - <component> Component MultiplyServiceComponent
    - <component> Component DivideServiceComponent
    - <service> Service CalculatorService
    - <wire> Wire CalculatorServiceComponent/addService
    - <wire> Wire CalculatorServiceComponent/subtractService
    - yService
    - ervice

SCA assembly file

Synchronisation between:
- SCA composite designer
- Tree editor
- Source editor

Property	Value
Name	AddService
Policy Sets	
Requires	

APIs

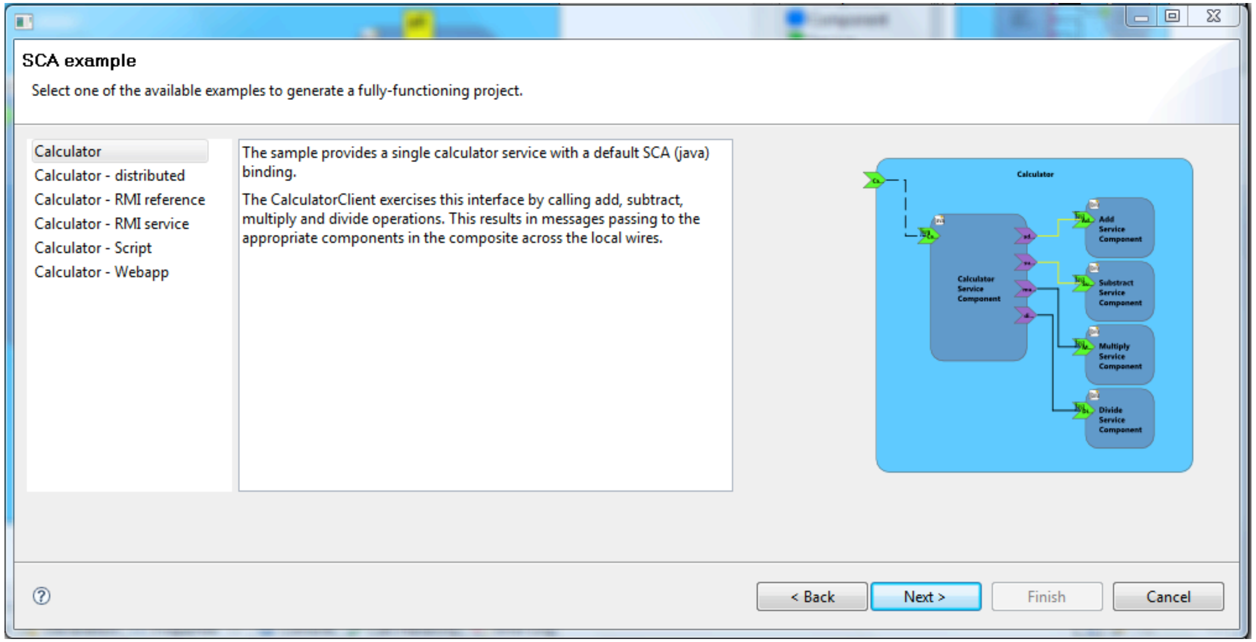
- Metamodels and corresponding implementations

Documentation

- User tutorial
  - First steps with the SCA Composite Designer

  http://wiki.eclipse.org/images/6/61/FirstStepsWithTheSCADesigner.pdf

- User samples
  - Set of ready to use SCA projects



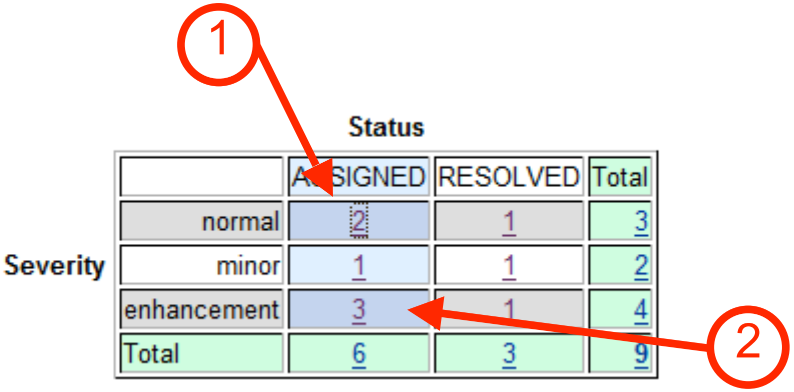
SCA example

Select one of the available examples to generate a fully-functioning project.

The sample provides a single calculator service with a default SCA (java) binding.

The CalculatorClient exercises this interface by calling add, subtract, multiply and divide operations. This results in messages passing to the appropriate components in the composite across the local wires.

Bugzilla



		Status		
		ASSIGNED	RESOLVED	Total
Severity	normal	2	1	3
	minor	1	1	2
	enhancement	3	1	4
	Total	6	3	9

- Item 1: one bug is a proposal for an XML editor and the other is a proposal to change the status of the SCA component in a sub-project of STP
- Item 2: 3 enhancements that are in progress (proposed by a contributor)
- Note : this figure is subject to changes (this snapshot has been taken on May 19, 2008)

Tool usability

- Localization

Standards

- The SCA Composite Designer is compliant with the SCA specifications 1.0

End of Life

- As this is the first release there are no specific end of life concerns

Communities

- Talks made in:
  - 6th franco-mexican school on distributed systems, 11-2007. Service Component Architecture: build systems using SOA
  - Solutions Linux 2008, 01-2008 (in French): Outils Eclipse d'aide au développement SCA
  - EclipseCon 2008, 03-2008. Building easily and quickly an SCA composite
- Articles
  - Programmez ! , 07-2008 (in French): Découverte de SCA avec Eclipse STP
- STP newsgroup

Schedule

- M4: 2008-01-08
- M5: 2008-02-20
- M6: 2008-04-09
- M7: 2008-05-07
- RC1: 2008-05-21
- RC2: 2008-05-28
- RC3: 2008-06-04
- RC4: 2008-06-11
- Ganymede: 2008-06-18

IP Issues

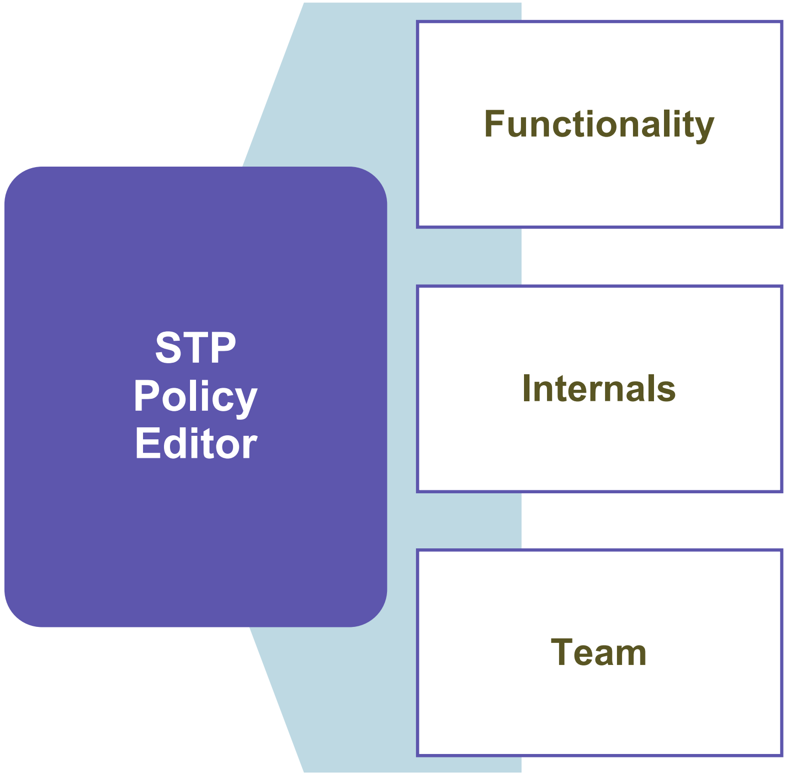
- About files and licenses in place
- No dependency on external JAR or libs
- No contribution was integrated for this release

Project plan

- June 2008 (Ganymede)
  - SCA specifications 1.0 from OSOA
  - Support of Tuscany SCA elements
  - Tutorial that explains how to use SCA tools
- December 2008
  - Full support of the additional validation rules
  - Support of Frascati SCA elements
  - SCA XML and Form Editors
  - Integration with the following components: STP Policy Editor, STP Service Creation and STP SOA System
- March 2009
  - Support of SCA specifications that should be published in December 2008 by OASIS
  - Support of POJO and EJB introspections

Policy Support in Eclipse STP
www.eclipse.org/stp

STP Policy Editor
- a generic, extensible editor for WS-Policy



STP Policy Editor

Functionality

- - editing of WS-Policy files
- - validation
- - design and source views
- - extendable with custom policy assertions

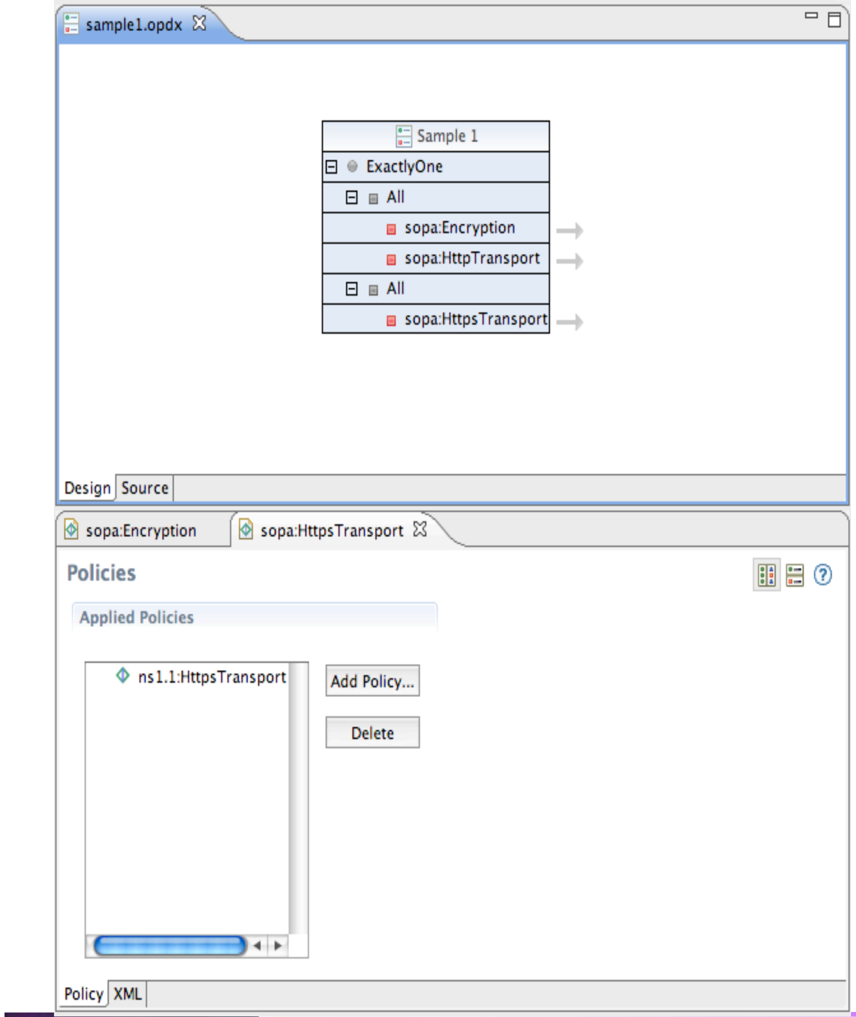
Internals

- - supports basic WS-Policy structure and validation
- - intended to be extensible via plug-ins, schemas, configuration (WIP)

Team

- - merges contributions from IONA and SOPERA
- - active development by both contributors

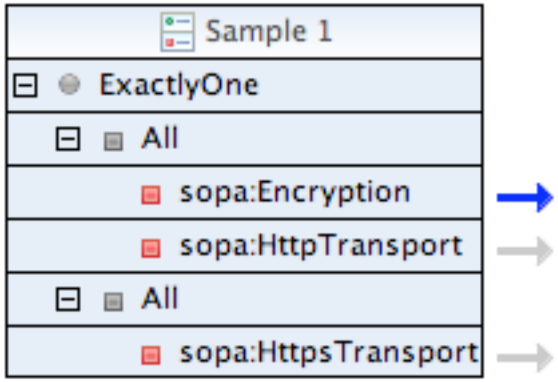
The Policy Editor integrates two editor components to provide a task-centered GUI



- The policy editor provides two editor windows:
- The high level editor shows the complete structure of the policy
- The detail editor shows one selected policy assertion together with all attributes

The high level editor manipulates the structure of the policy

The high level editor manipulates the structure of the policy



Sample 1

- ExactlyOne
  - All
    - sopa:Encryption
    - sopa:HttpTransport
  - All
    - sopa:HttpsTransport

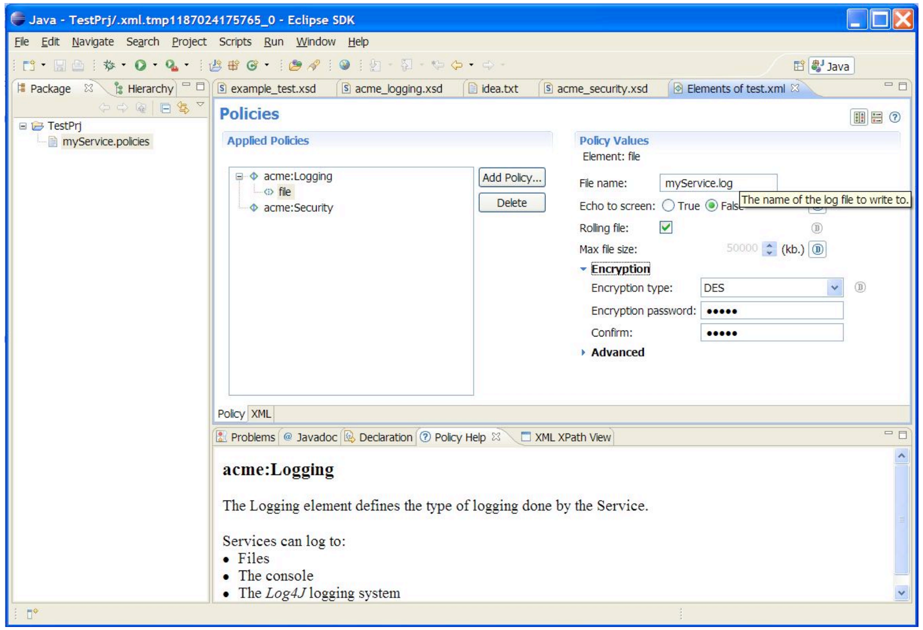
From the high level editor, you can

- add and remove compositors
- add and remove individual assertions
- switch to the detail editor to work with an individual assertion

The details editor is used to edit individual assertions

Details Editor

- Similar in look & feel to PDE Extension Point editor
- Can edit the details of WS-Policy assertions as well as other types of XML files that contain embedded elements.
- Editor dynamically synthesizes a GUI based on the schema definition of the policy assertions.
- GUI works with most standard XML Schema definitions
- Based on XEF (also part of STP)



Element: file

File name: myService.log

The name of the log file to write to.

Echo to screen: ○ True ● False

Rolling file: ☑

Max file size: 50000 (kb.)

Encryption

Encryption type: DES

Encryption password: •••••

Confirm: •••••

Advanced

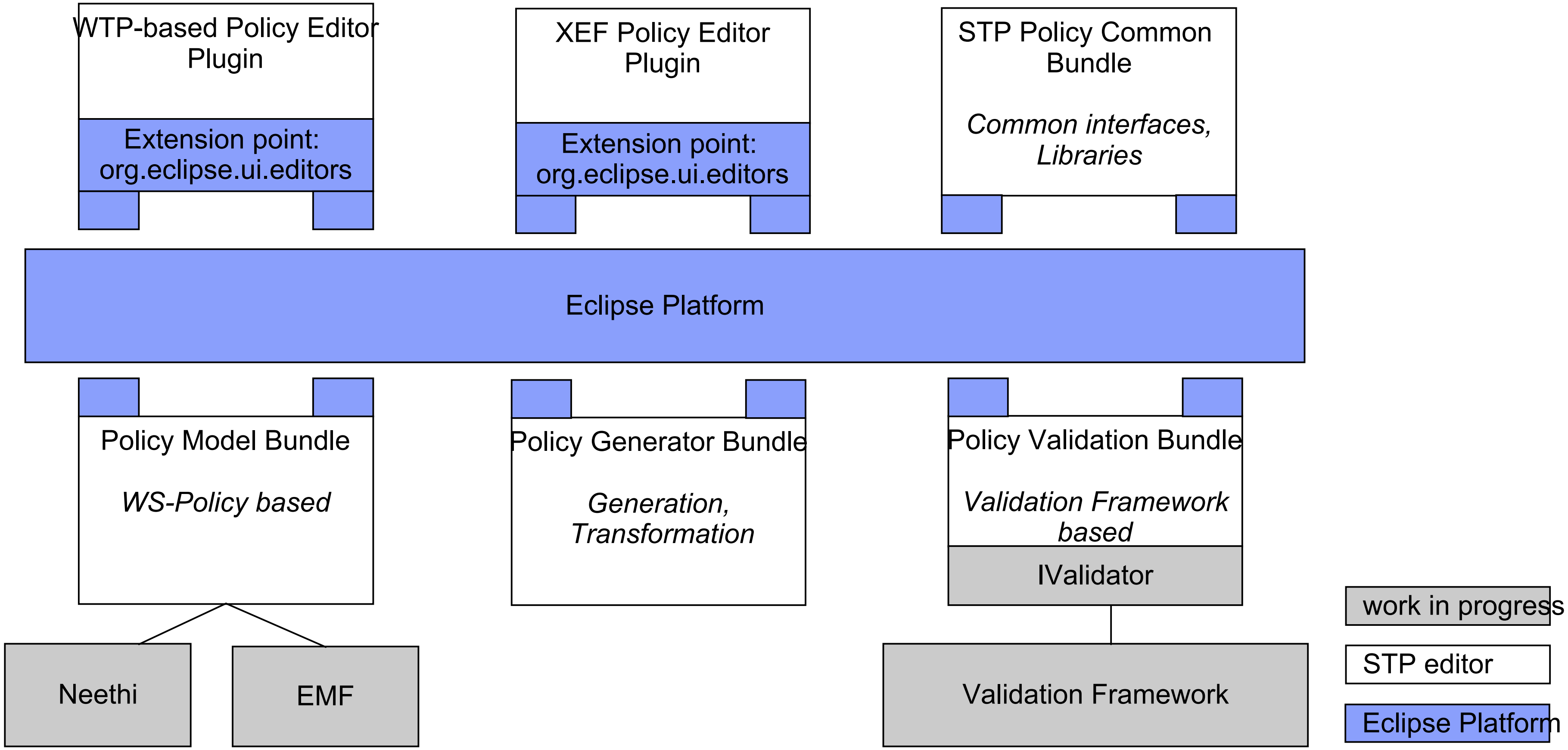
acme:Logging

The Logging element defines the type of logging done by the Service.

Services can log to:

- Files
- The console
- The *Log4J* logging system

The implementation is split into multiple plug-ins according to functionality



Contacts

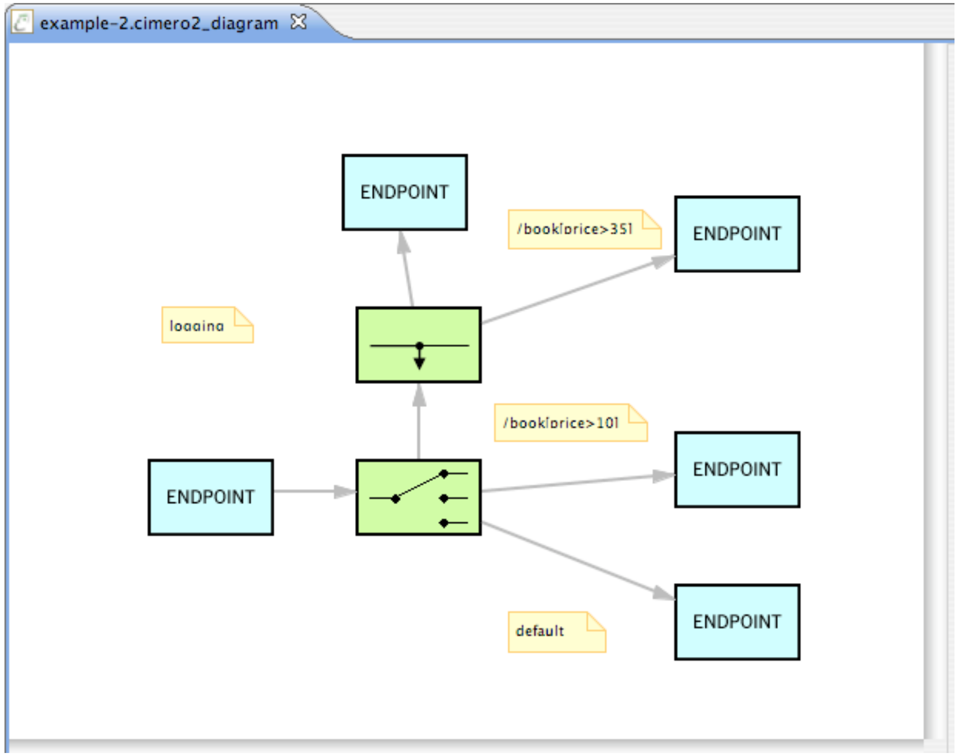
- David Bosschaert (davidb@iona.com)
- Jerry Preissler (gerald.preissler@sopera.de)
- Andrei Shakirin (andrei.shakirin@sopera.de)

Enterprise Integration Designer

Ganymede Release

Enterprise Integration Designer

- Diagrammatic editor for Enterprise Integration Patterns
- Provides a code and configuration generation framework
- Provides a runtime environment extension framework
- Initial runtimes ServiceMix3, PeTaLs, Camel



Enterprise Integration Designer

- Committers from Bull, EBM Websourcing and IONA
- Relatively new component, not ready for 1.0 yet, APIs are still in the process of construction
- Information is available at
  - http://wiki.eclipse.org/STP/EID_Component
- Component is undergoing active development as extension APIs are being developed and tested
- Future plans include
  - extension points for reverse-engineering configuration
  - integration with the STP Intermediate Model
  - extension points for shared component storage