AM03

What‘s New in RSA and RSM 7.5

Kim Letkeman

Senior Technical Staff Member

Rational Modeling Platform

kletkema@ca.ibm.com

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

IBM

Rational. software

Good News, Bad News Story …

- The **good** news is that RSA and RSM 7.5 are **massively** improved in every way imaginable …

Good News, Bad News Story …

- The **bad** news is that RSA and RSM 7.5 are **massively** improved in every way imaginable …

Good News, Bad News Story …

- The **bad** news is that RSA and RSM 7.5 are **massively** improved in every way imaginable …
  - It takes 250+ slides to explain the improvements
  - This presentation is therefore a **summary** of the high points
    - The deck contains all of the details, with most of the slides hidden for the presentation to give us a fighting chance

AM03

What‘s New in RSA and RSM 7.5

Highlights

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

IBM

Rational. software

RSA 7.5 Highlights

- Packaging clarified
  - RSA for Websphere, RSA standard and RSM
- Diagram editing
  - Improved layout, editing, printing, palette
- UML
  - Editing, compliance, tooling and improvements to sequence and activity diagramming
- Reporting
  - BIRT integration, web publishing, traceability

RSA 7.5 Highlights (2)

- Team development
  - Morph deltas, sparse logical model merge, text view mode, delta tree filters, sorters and groupers, pane visibility settings
- Model-driven development
  - Integrate to WBM 6.1.X and SOMA ME
  - Java modeling, visualization and transformation improvements
  - Support for JPA and EJB 3.0
    - Profiles for both
    - New forward and reverse transformations for both
  - SOA transformations enhanced
  - New services transformations for java and session beans

RSA 7.5 Highlights (3)

- Model-driven development authoring tools and solutions
  - Visual profile development, query-based tooling, per diagram palettes
  - Intent-based transform config wizard, cmd line automation, transform instances
  - M2M transform mapping types, predef rules, UML support all enhanced
  - M2T (JET) authoring and engine both streamlined dramatically
  - Patterns tooling enhanced, DevWorks assets migrated
- RSM / RSD 7.0.5 feature migration
  - Configurable user interface, viewpoints, reminders, rich text
  - Improved UML 2.1 compliance
  - BIRT reporting UML / EMF, model analysis and metrics, UPDM
  - Significant UML model and diagram editing improvements

RSA 7.5 Highlights (4)

- RSM / RSD 7.0.5 ***extensibility*** feature migration
  - M2M, M2T authoring enhancements, JET syntax aware editor, patterns framework enhancements, significant open source releases for GMF, EMF, UML2, etc
  - UML profile tooling generation (palettes, menus, diagram assistants, capabilities, …)
    - Build UML-based DSLs
  - Reminders framework, queries framework
  - Viewpoints, configurable modeling UI
  - Model analysis/metrics
- Deployment modeling
  - Extensibility framework
  - Topologies

AM03

What‘s New in RSA and RSM 7.5

Modeling Platform

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

IBM

Rational. software

Diagramming Improvements

- Multi-platform printing with WYSIWYG preview
- Significant layout improvements
  - Routing avoidance, rectilinear everywhere, border nodes
- Significant diagram editor improvements
  - Save As Image in PDF format, also detect and avoid OOM on large diagrams
  - Palette has a new look
  - Group / Ungroup, Snap to grid, Text Alignment, Line Styles

Diagram Improvements: Printing Support

- Multi-platform printing is now supported.
  - Linux
  - Solaris
  - Windows

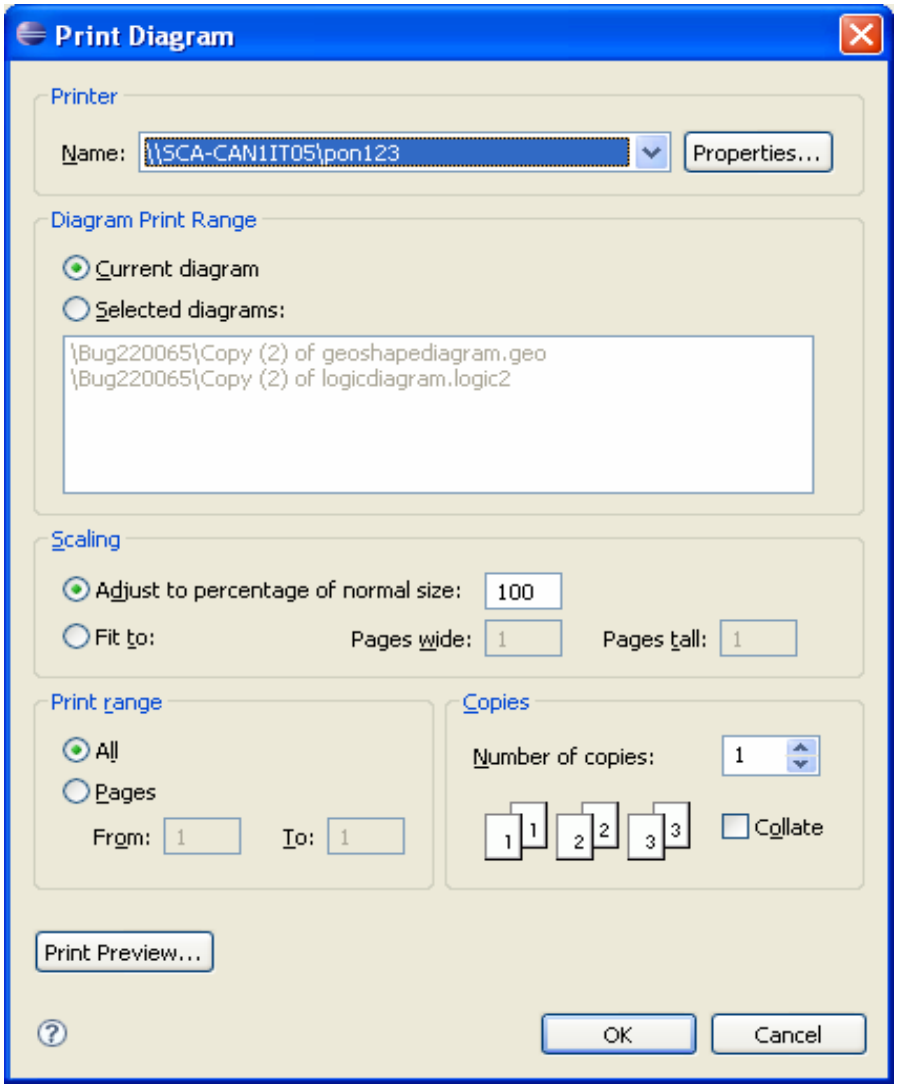


Diagram Improvements: Print Preview

- Print preview is now **WYSIWYG**.
- Adjustable zoom settings supported.
- Scale to fit supported.

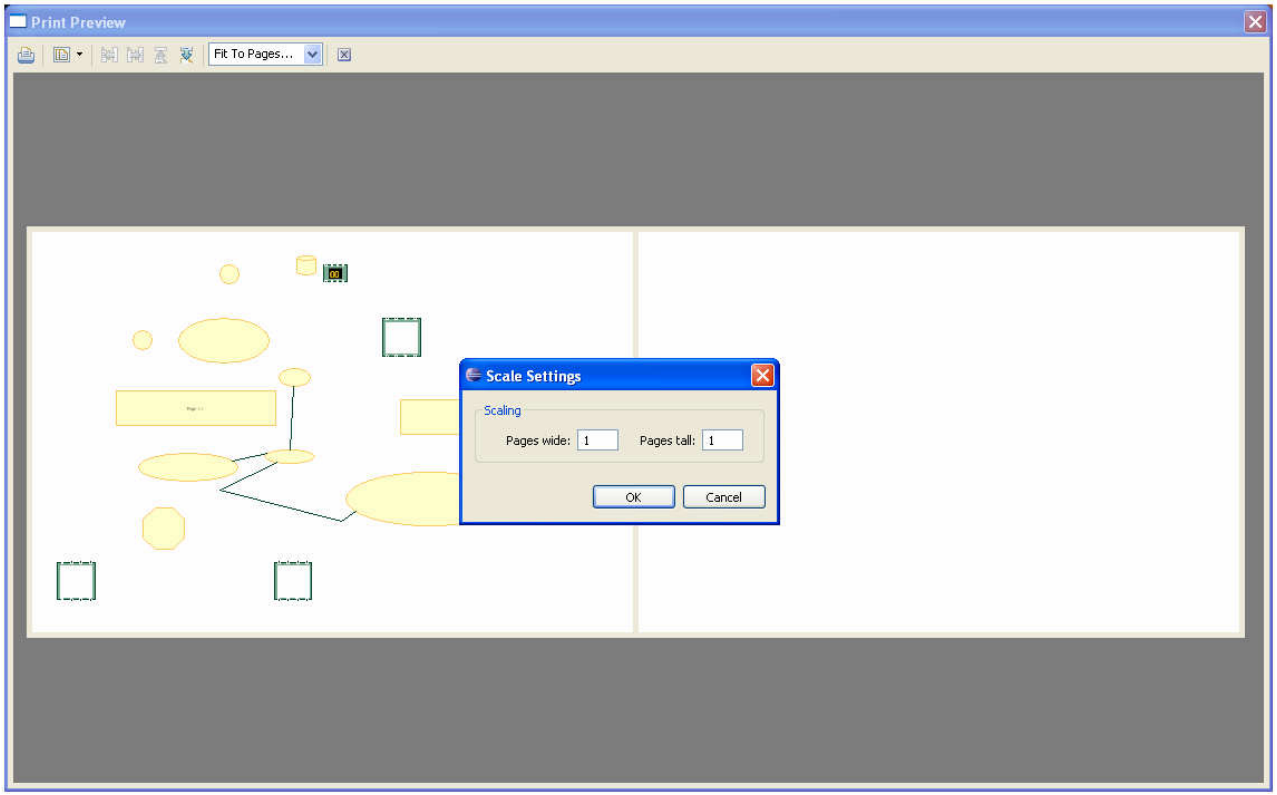
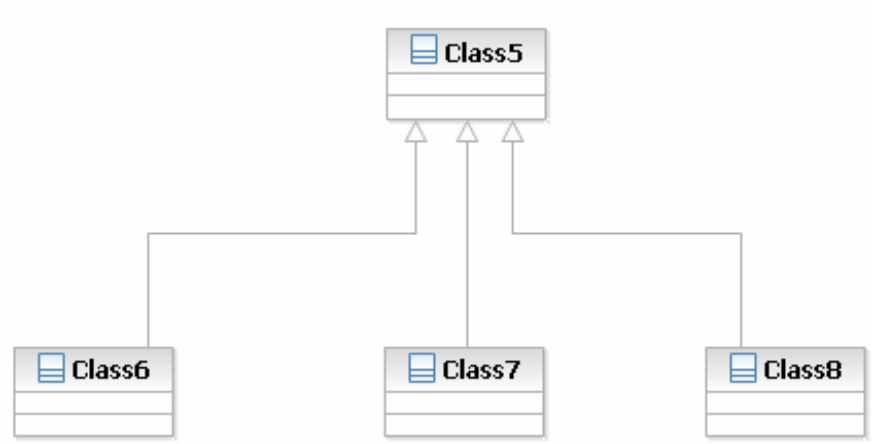


Diagram Improvements: Connection Routing

- Rectilinear Routing
  - Line segments movement fixed
  - Can move bend points
  - Layout enhancements

NEW



OLD

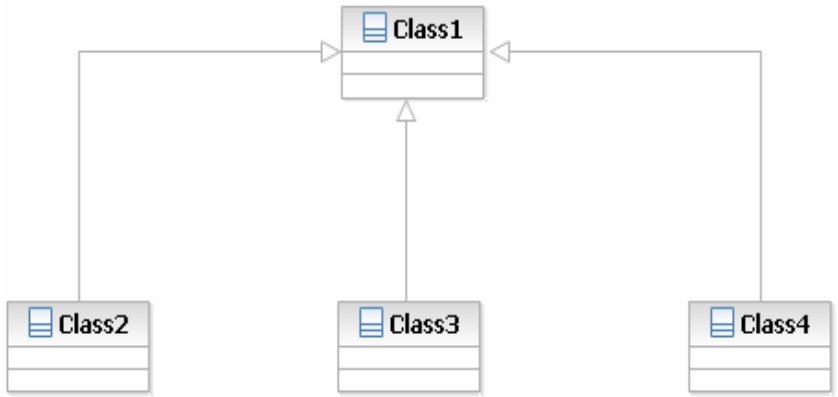


Diagram Improvements: Connection Routing (Cont.)

- Routing of connections between border items

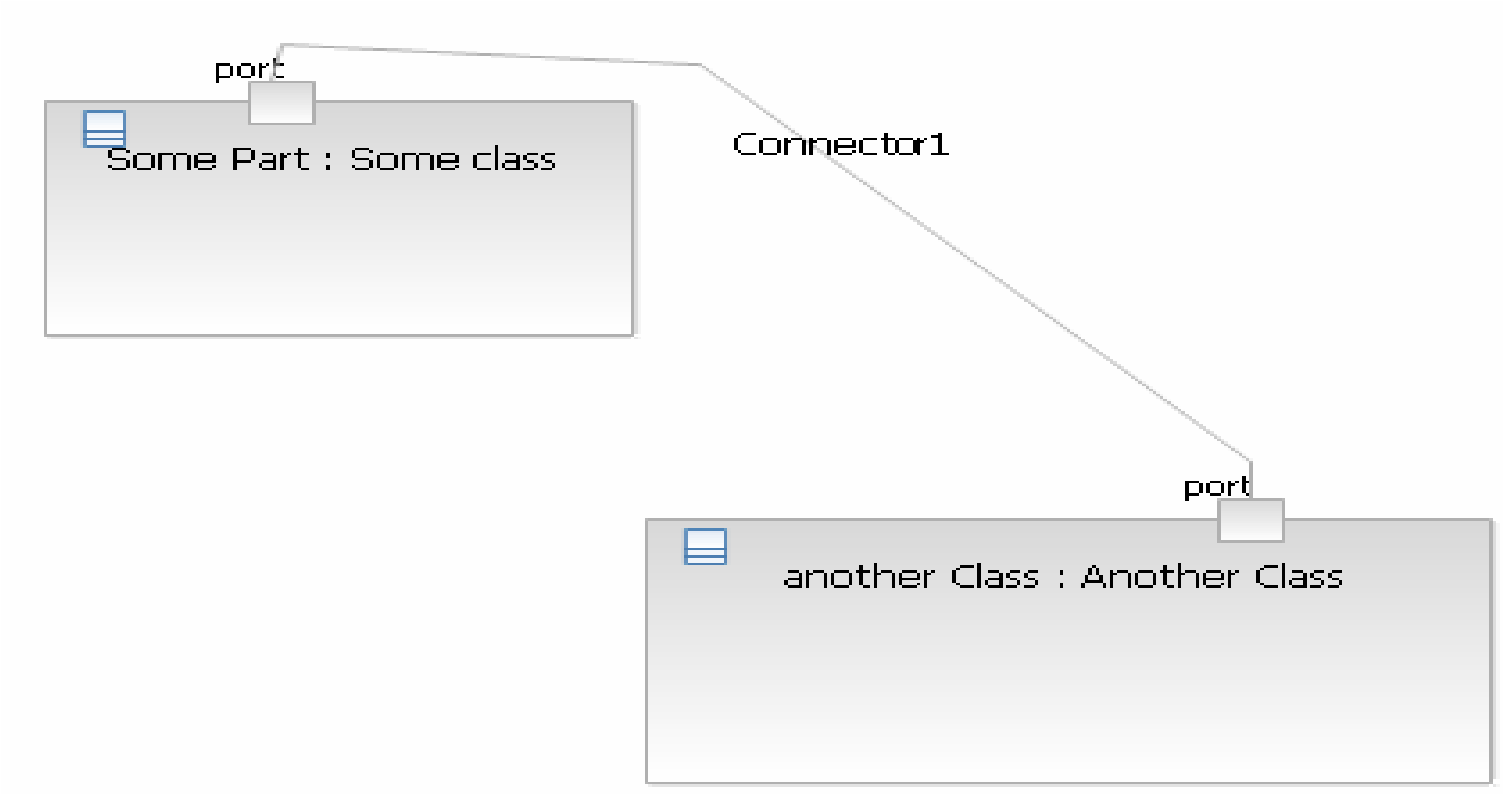


Diagram Improvements: Arrange All

- Graph layout results are precisely delivered to the diagram.
- **Before**

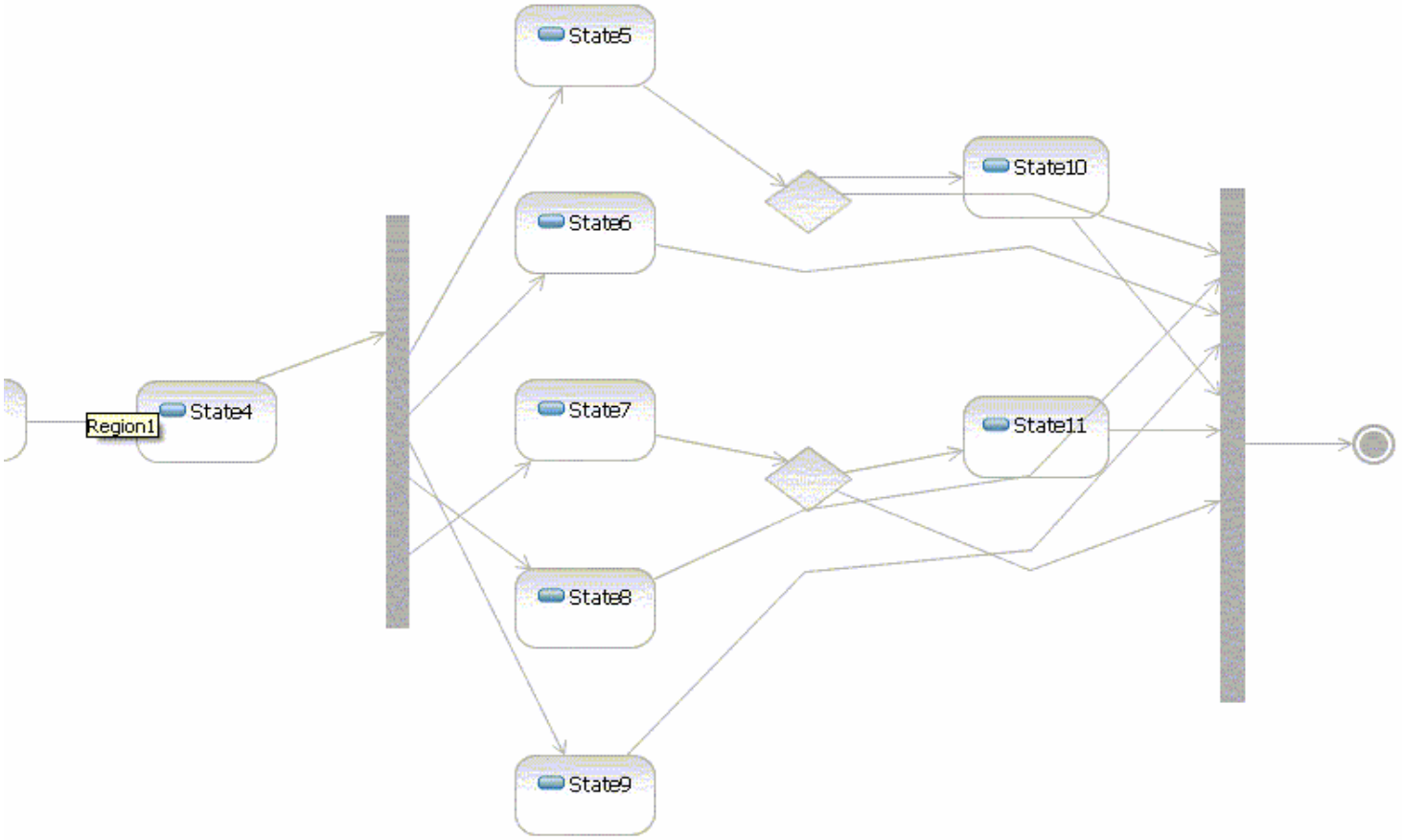


Diagram Improvements: Arrange All (Cont.)

- **After**

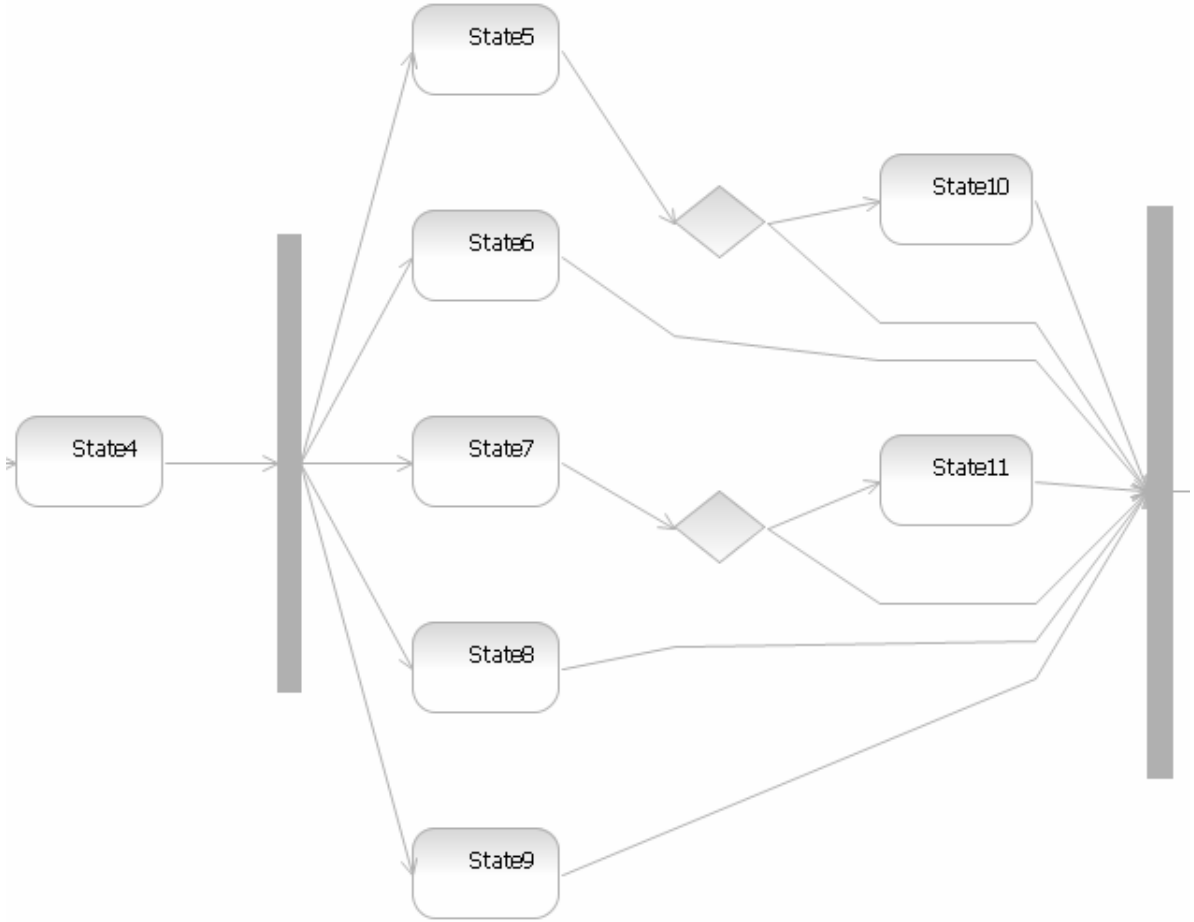


Diagram Improvements: Arrange All (Cont.)

- Support for rectilinear connections
- Support for distinct connection end points

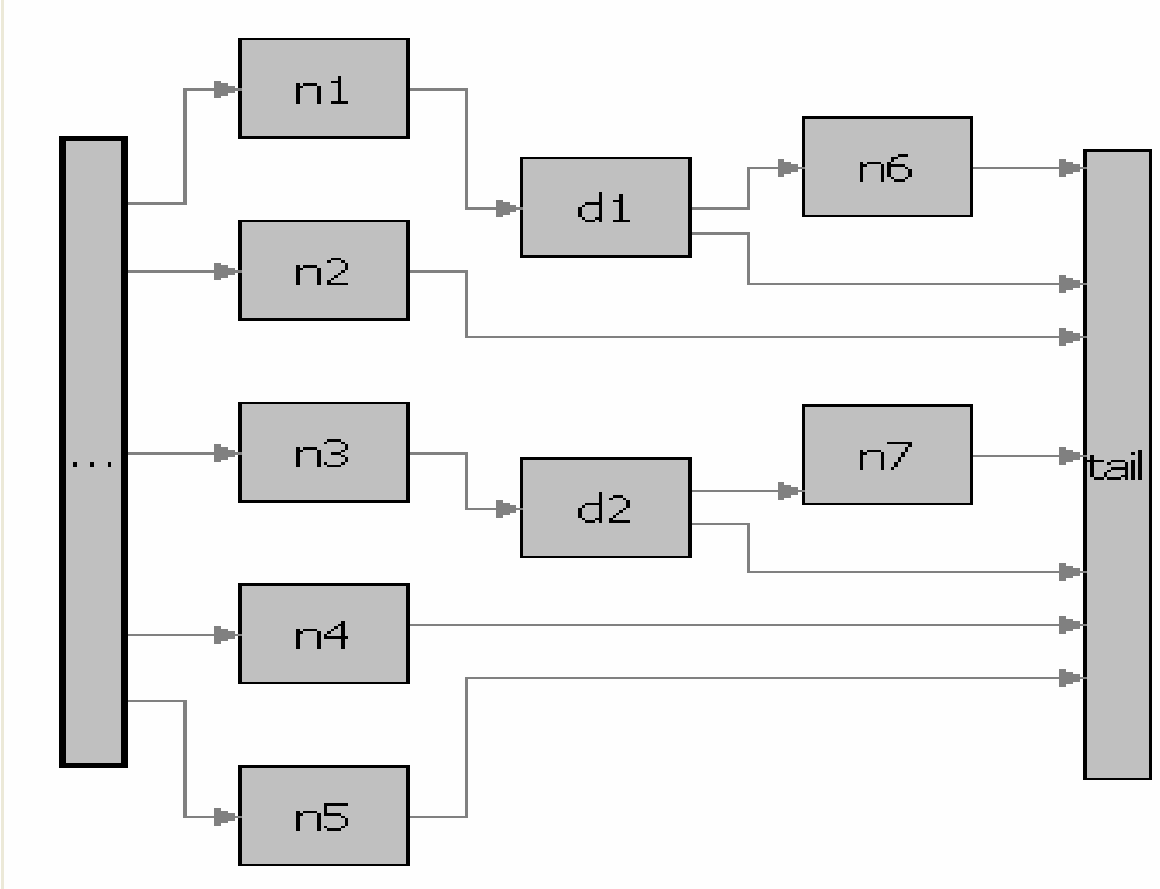


Diagram Improvements: Arrange All (Cont.)

- Basic support for border nodes

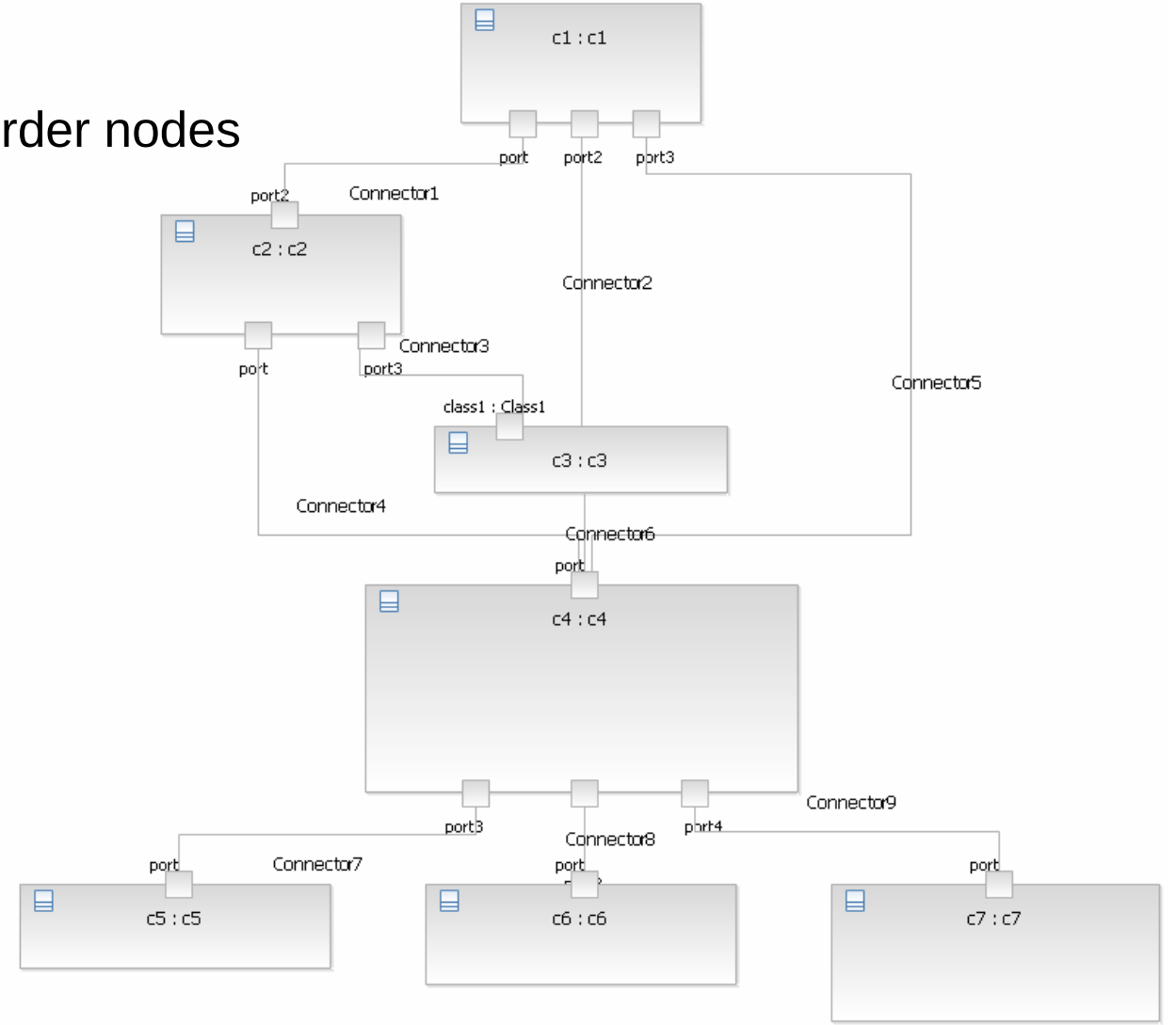


Diagram Improvements: Save as image

- PDF format now supported by save as image file command.
- Export to HTML
  - Large diagrams can fail with out of memory (OOM) error.
  - OOM now detected, allowing the diagram to be split into pieces.
  - The diagram can be viewed in an image map in the generated HTML file.

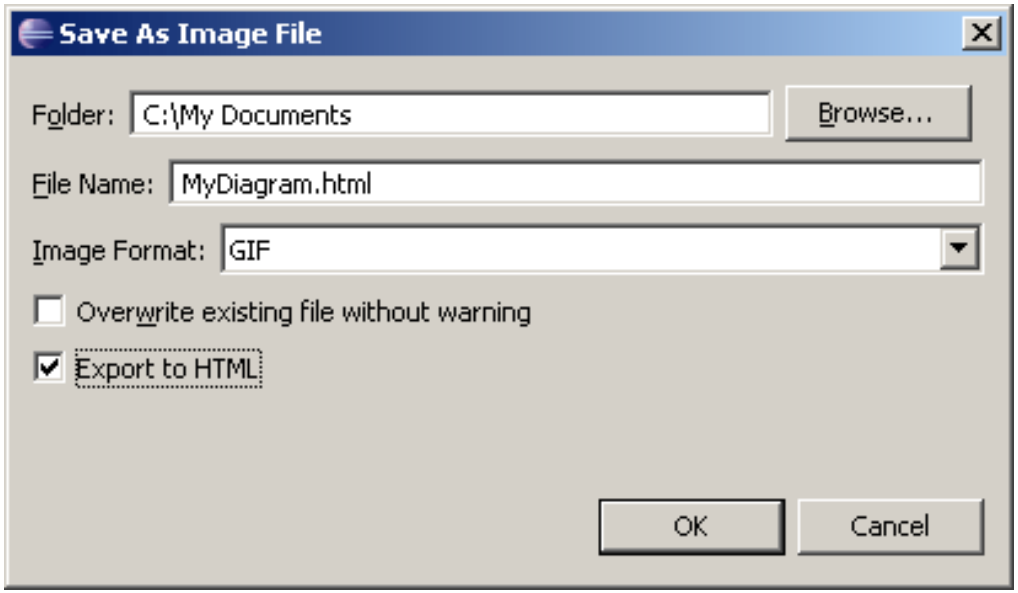


Diagram Improvements: The Palette's New Look

- New design for the diagram palette.
- What has changed:
  - Colors
  - Mouse hover and selected.
  - Pin in and Pin out.
  - Improved icons.
  - Dock and expanded look.
  - Toolbar group
  - New stacks, including flyouts and pinned stack.

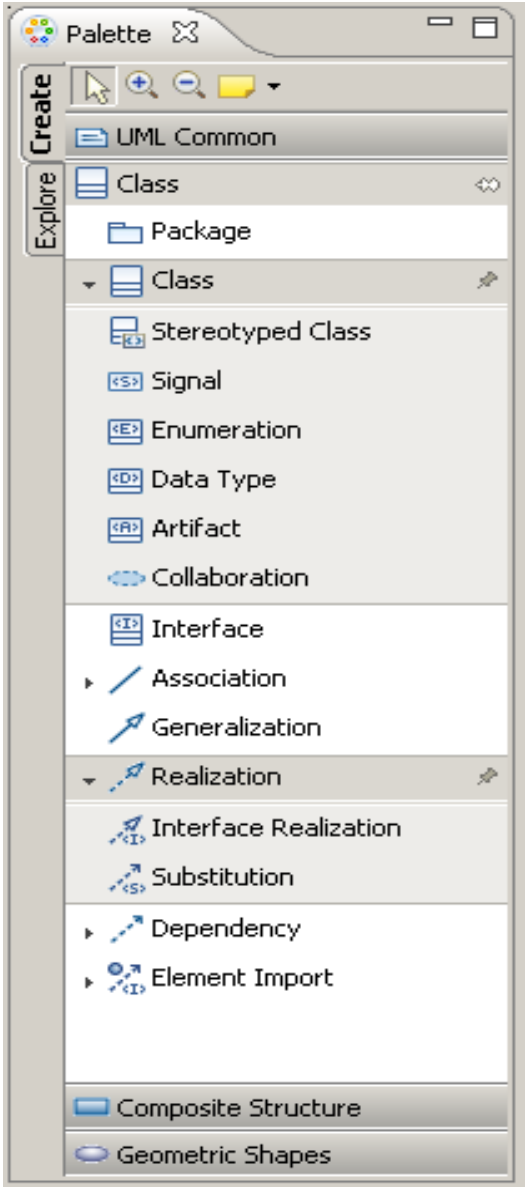


Diagram Improvements: Group / Ungroup

- Group and ungroup is now supported.

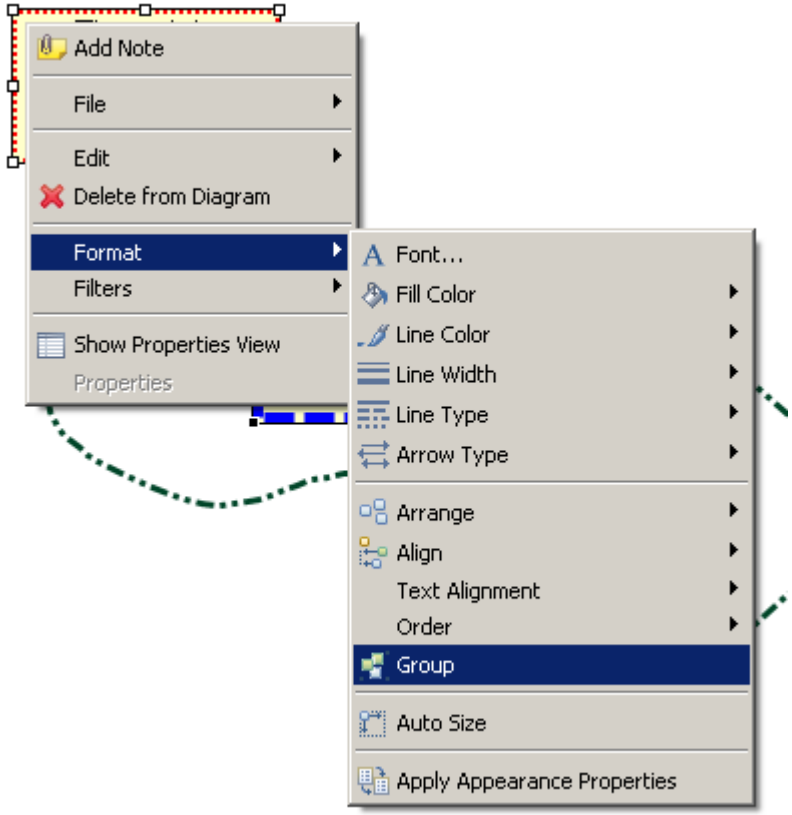


Diagram Improvements: Snap to enhancements

- Snap to shape now supported
  - Align shapes by side or center.
- Connection bendpoints now can snap to grid
- Shapes can now be moved using the arrow keys on the keyboard.

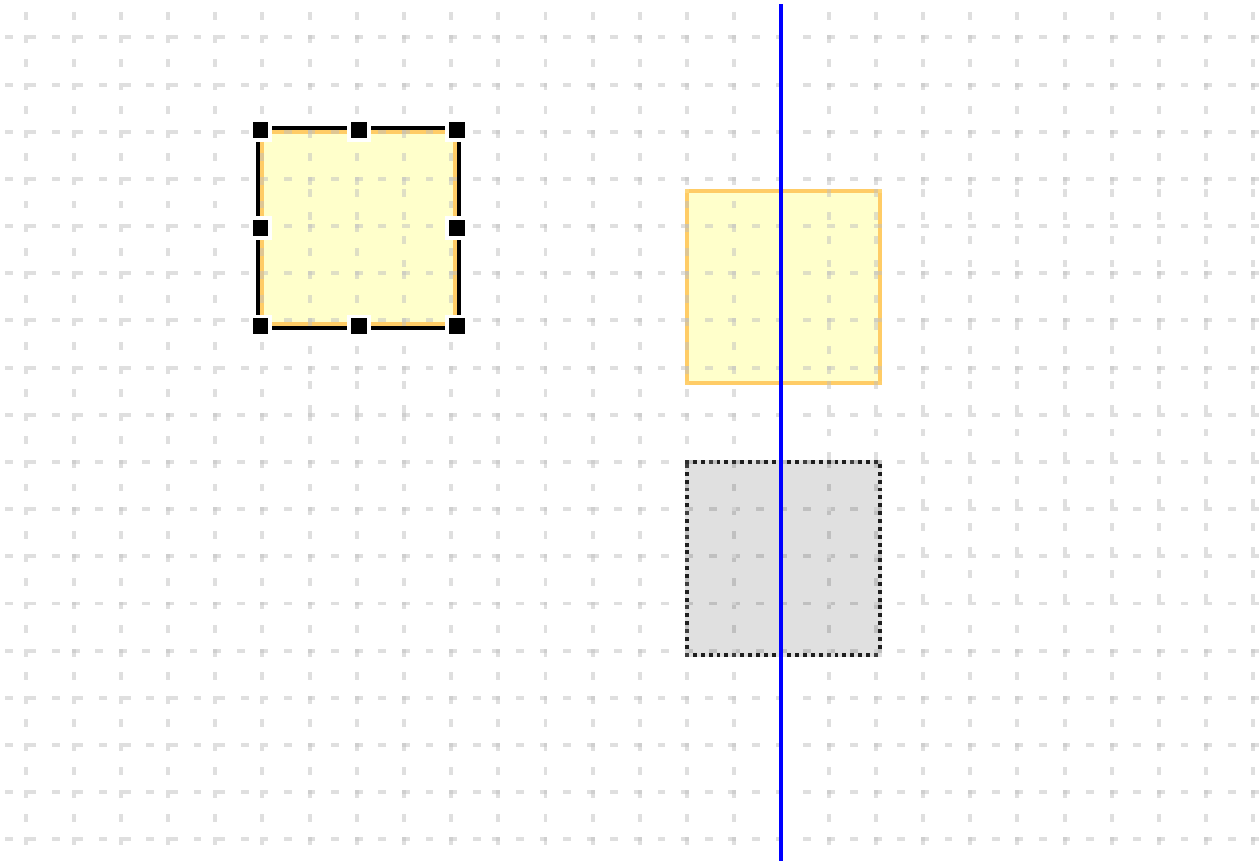


Diagram Improvements: Customize Palette

- New customize palette dialog.
- Rename and move palette entries
- Hide palette entries.

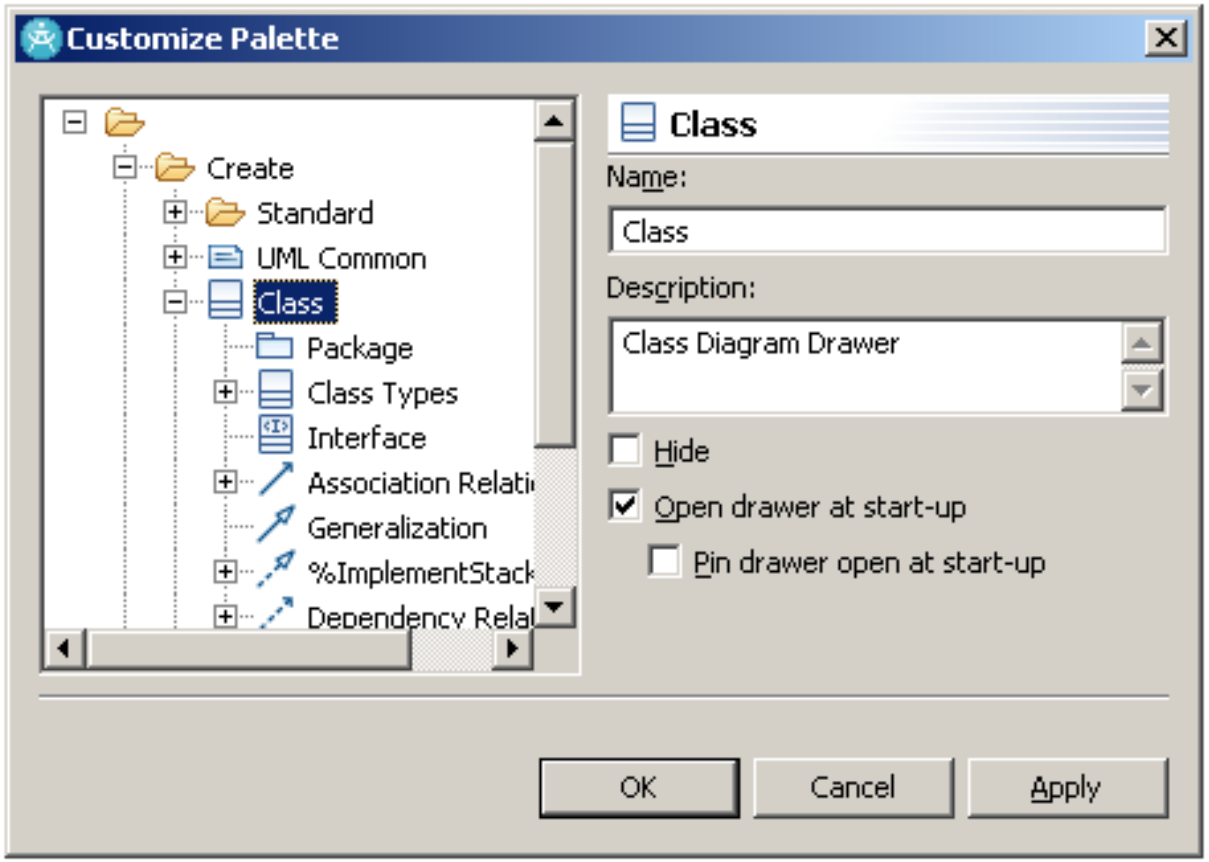


Diagram Improvements: Text Alignment Style

- Text can be left, right or center aligned.

Align Left

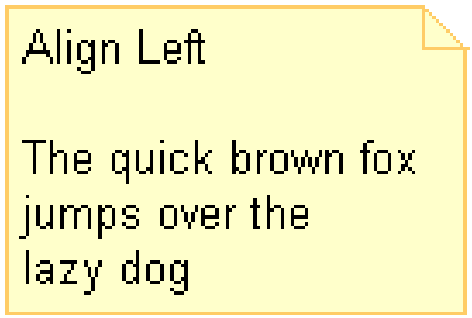
The quick brown fox jumps over the lazy dog

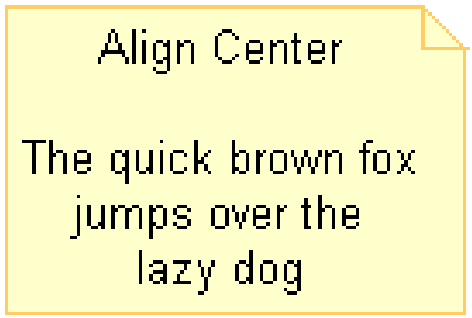
Align Center

The quick brown fox jumps over the lazy dog

Align Right

The quick brown fox jumps over the lazy dog





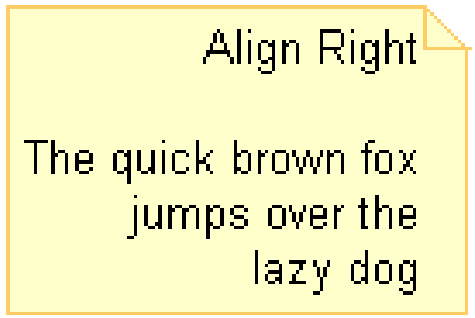
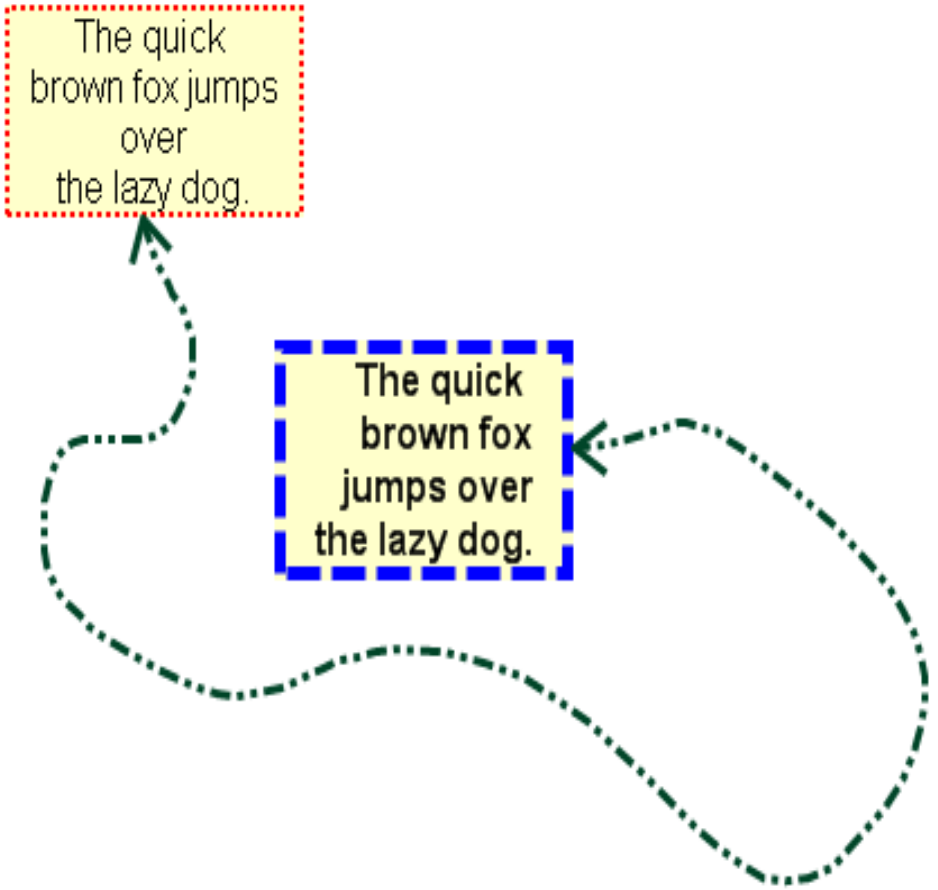


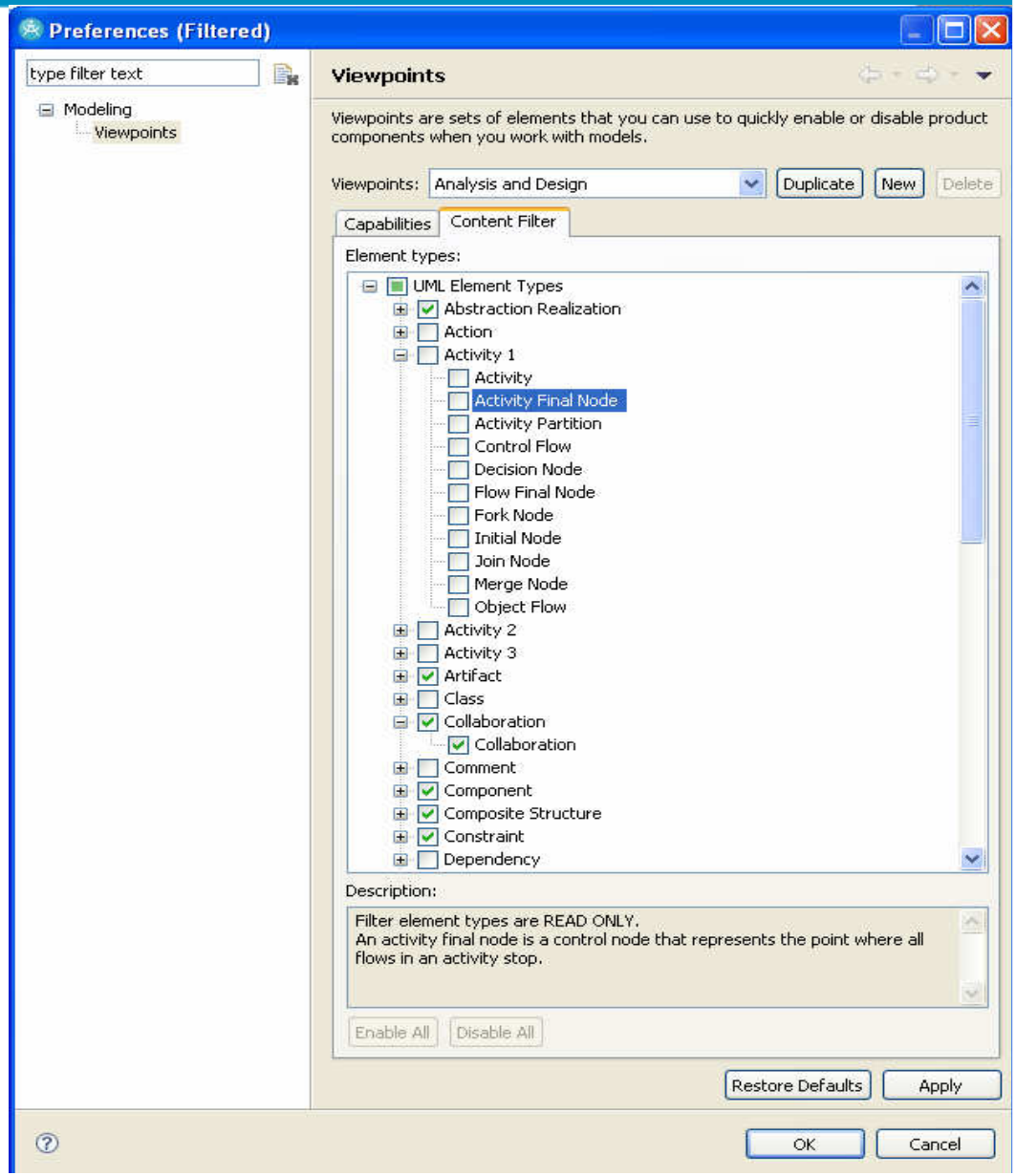
Diagram Improvements: Line Styles

- Line type
  - Can now change from a solid line.
  - Dash, dot, dash dot, etc.
- Line thickness
  - Lines can be more than one point.
- Arrow heads
  - Can be added to geometric lines.
  - Solid arrow, open arrow.



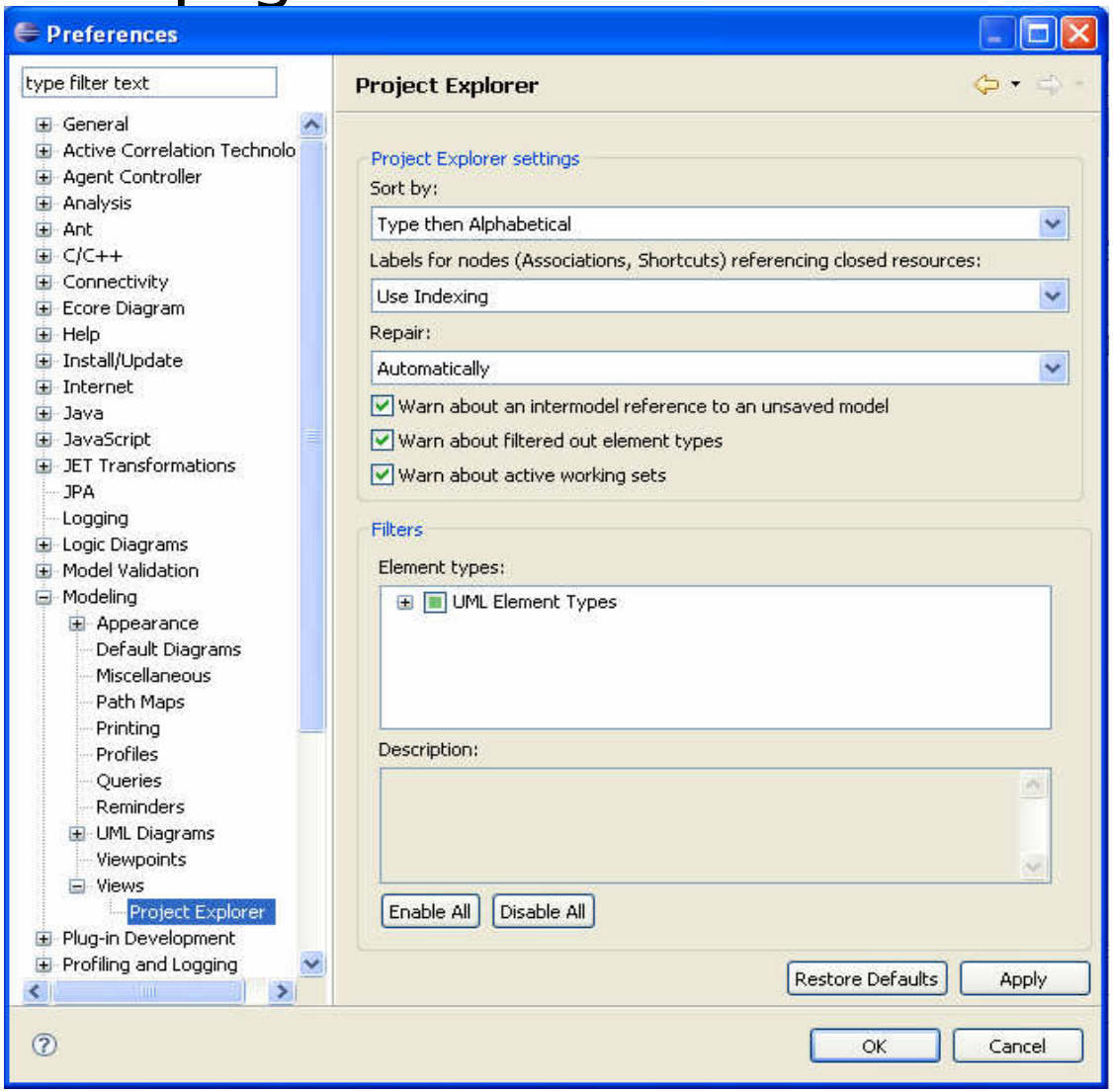
Viewpoints with Content

- In addition to capabilities, Viewpoints allow reduction of any content
- Can contribute custom content reduction to Viewpoints
- UML Modeler will reduce Project Explorer content when switching between Viewpoints
- Consider creating viewpoints to match roles, with more or less complex user interfaces



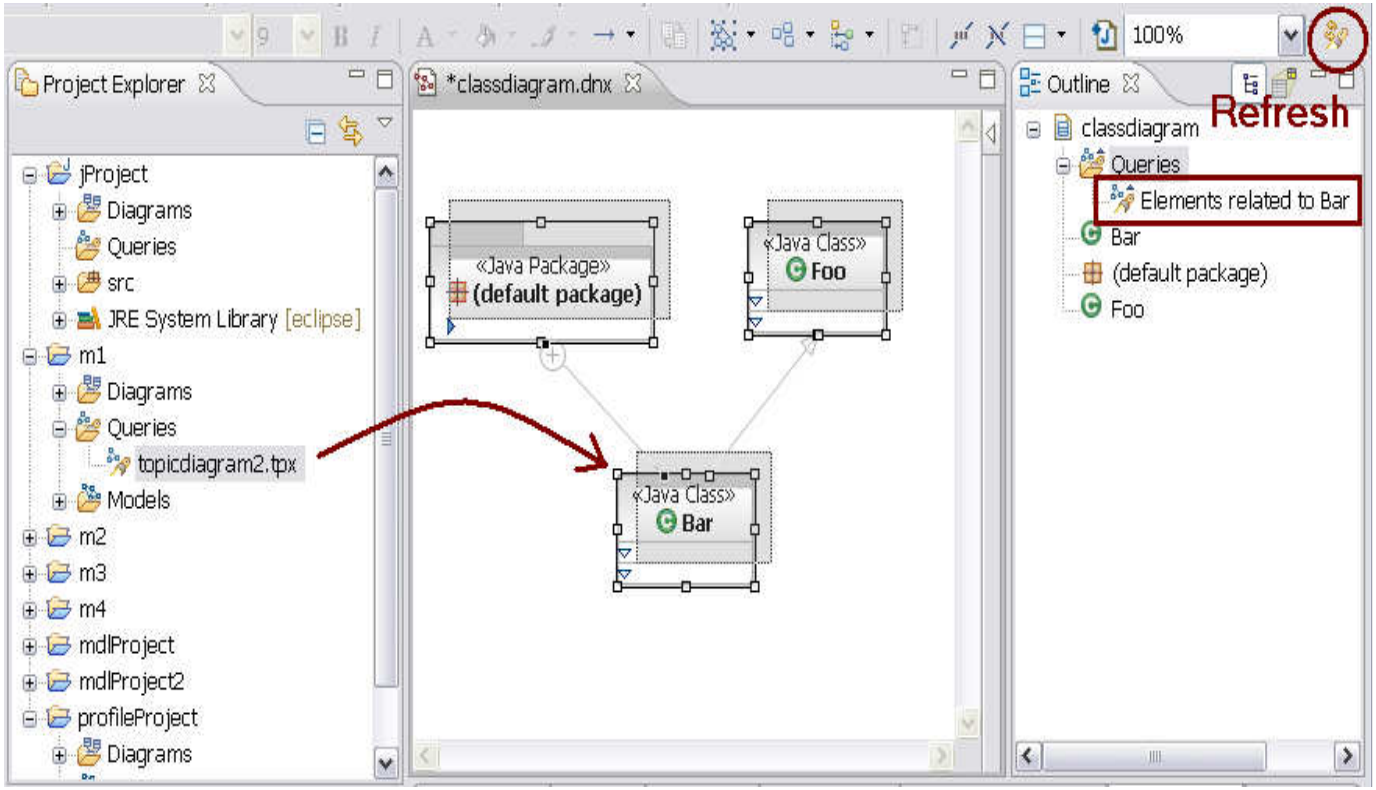
Project Explorer preference page

- Improved filtering, UML elements are displayed in a tree structure
- Preference for warning about model move refactoring to active working set



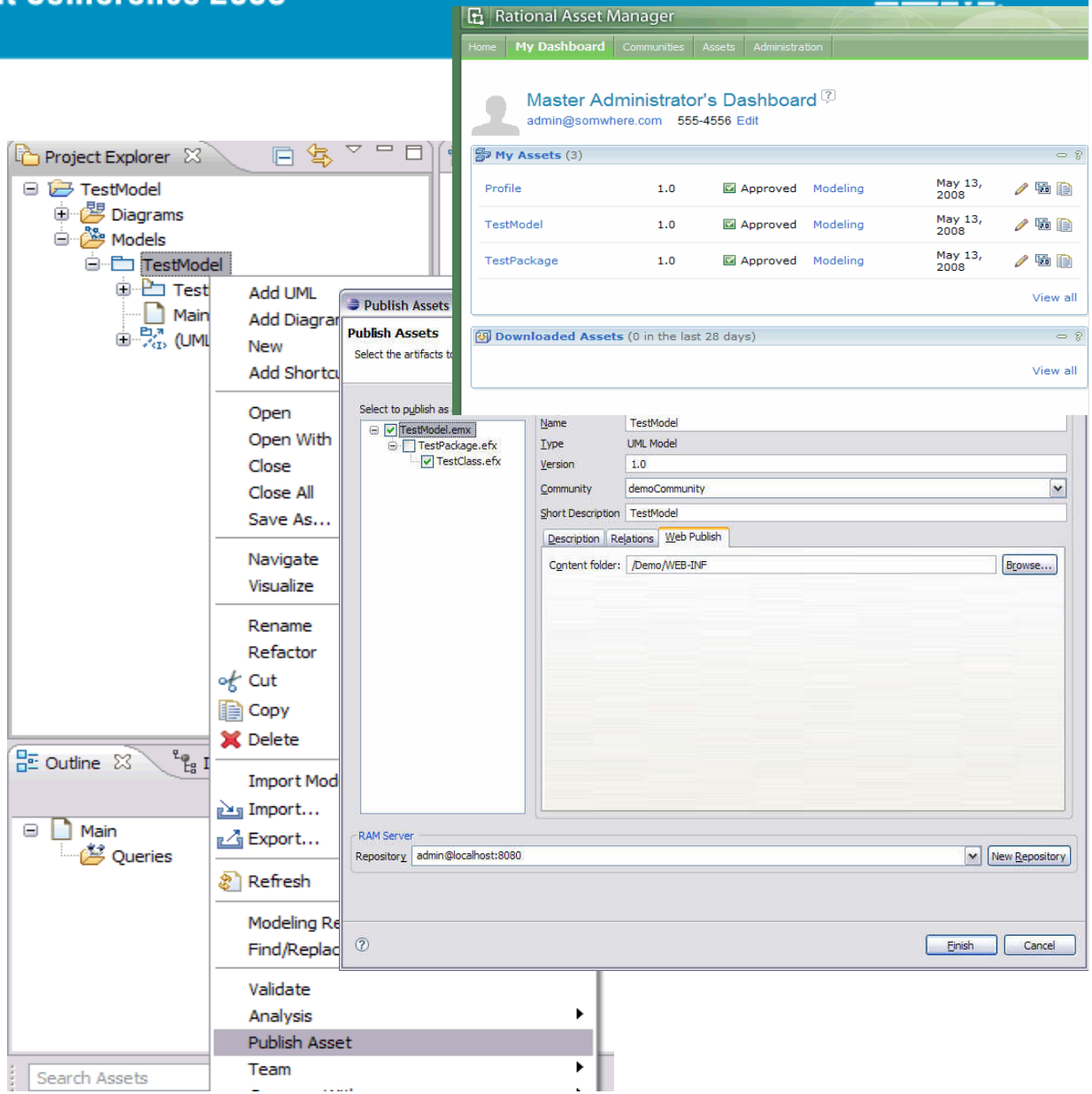
Querying

- Topic diagram wizard shows topic selection page before the context selection page, which is much more user friendly
- Can drag and drop queries (.tpx files) from workspace on a modelling or visualization diagram to show the results
- Queries such as UPDM can now be refreshed on UML diagrams
- Executable queries are created in the workspace as *.tpx files
- Topic diagram provides a preview of the query results

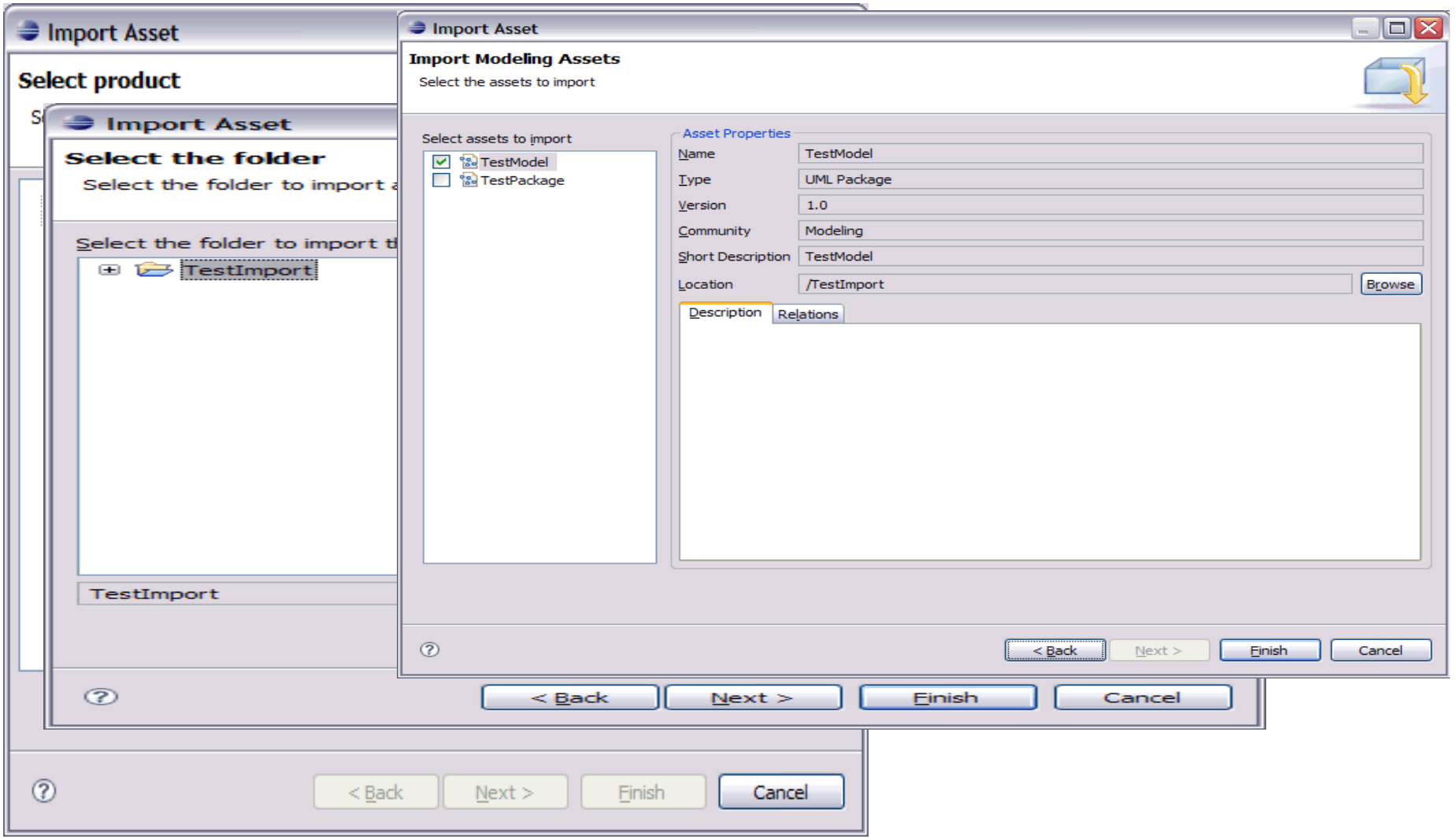


RAM Integration

- Publish Models, Fragments and Profiles to RAM repository
  - Asset integrity – publishes all required models
  - Separate assets for fragments in a model
  - Publish navigable web version of model.
- Import published assets
  - Project independent import
- Modify and Republish
  - Version maintenance – tracking and up-versioning



RAM Integration: Import Assets



AM03

What‘s New in RSA and RSM 7.5

UML Modeling

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

IBM

Rational. software

May 2008

UML Modeling

- Editing improvements
  - Rich text support, improved over 7.0.X
  - Refactoring enhancements: include closed models, preview, extendable
  - Content assist
  - Select element dialog can see into closed models!
  - Group / ungroup support
  - Activity diagram enhancements / simplification
  - Sequence diagram enhancements
  - Visualization of stereotype and profile applications

UML Modeling (2)

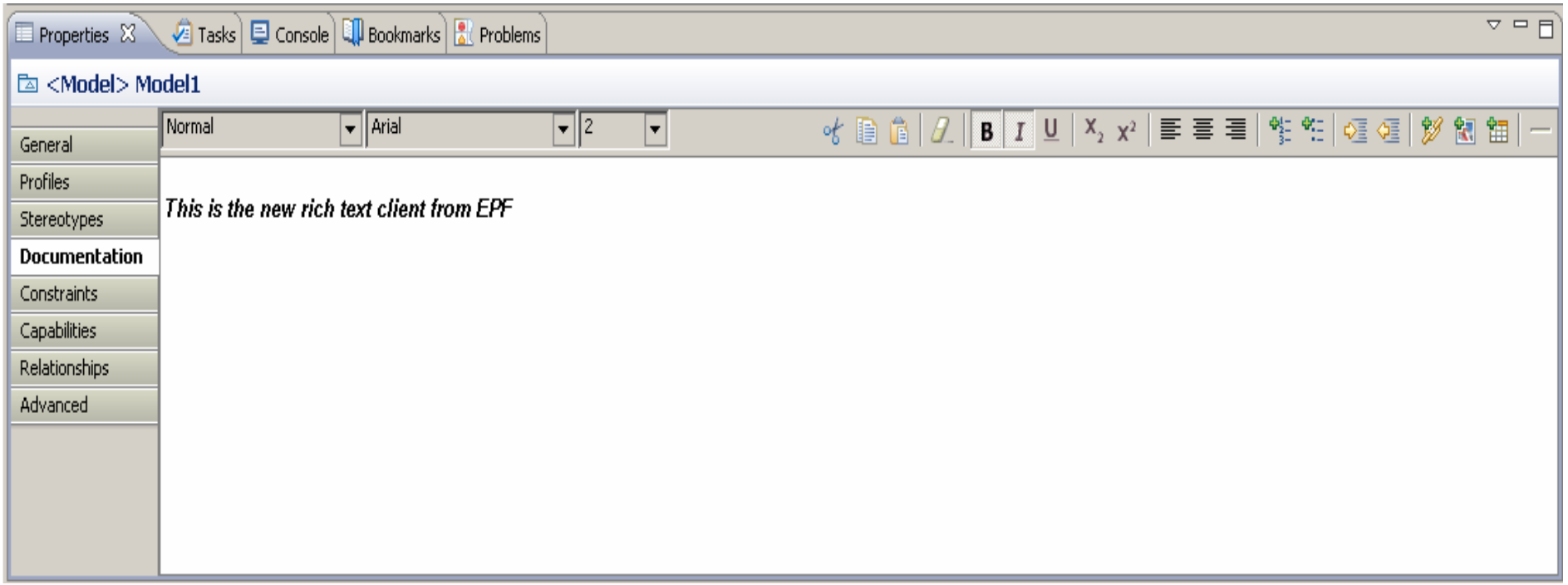
- UML compliance improvements
  - Operation exceptions easily added
  - Modeling multi-valued attributes
  - Associating rule with validation failure to improve clarity
  - Improved relationship consumability
  - Enumeration literals

UML Modeling (3)

- Tooling improvements
  - Closed models and fragments
  - Packages as model roots
  - Find and replace improvements
  - UML tools for palette explore tab
  - Workspace URL support
  - Preferences enhancements
  - Use existing views setting

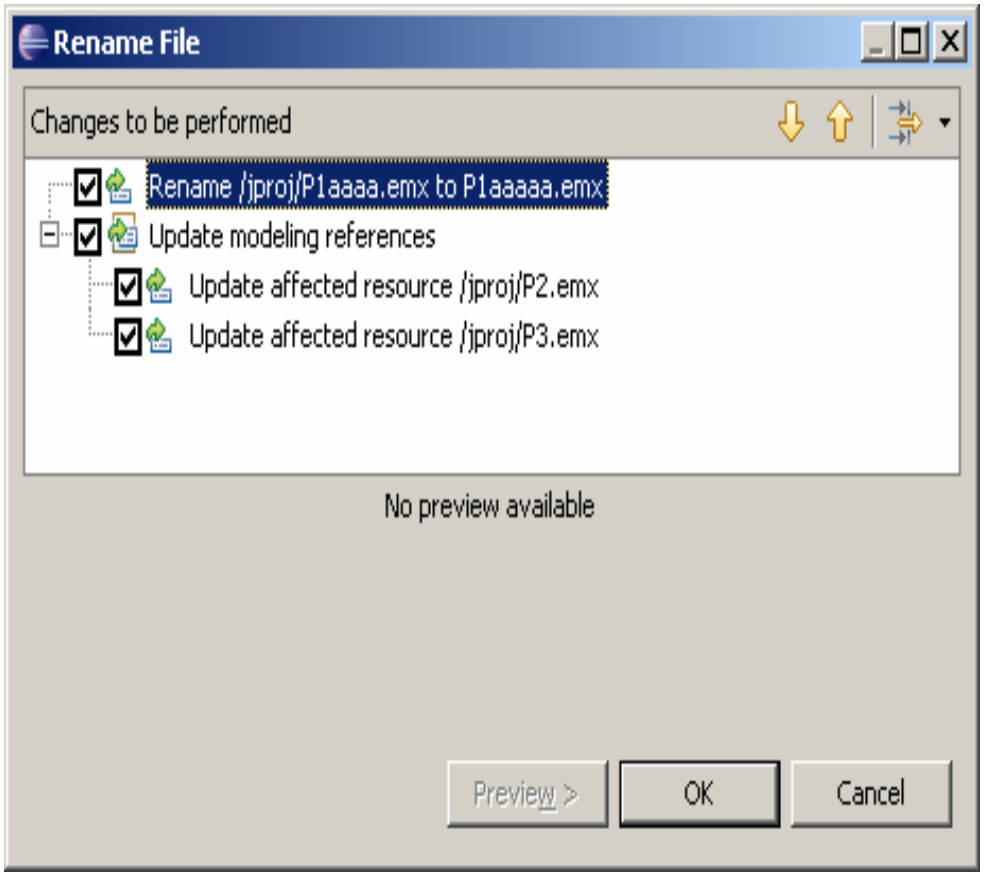
Rich Text Support

- New in RSA 7.5
- First appeared in RSM / RSD 7.0.5; improved in RSA 7.5
- Available for documentation, notes and comments



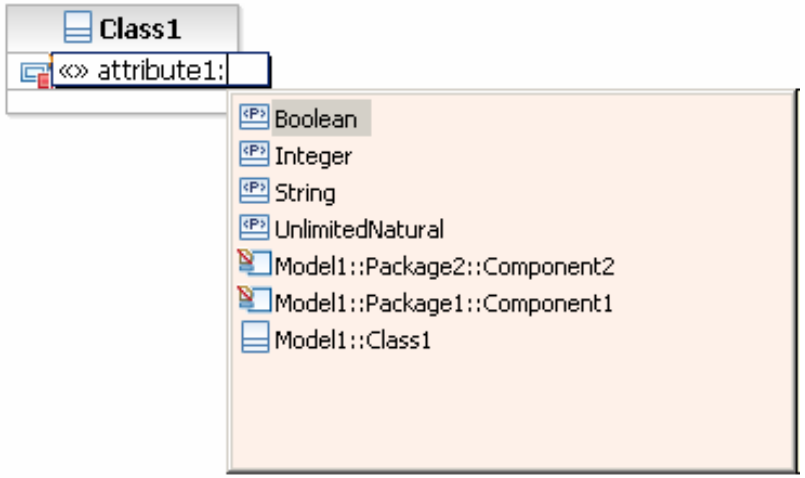
Refactoring enhancements

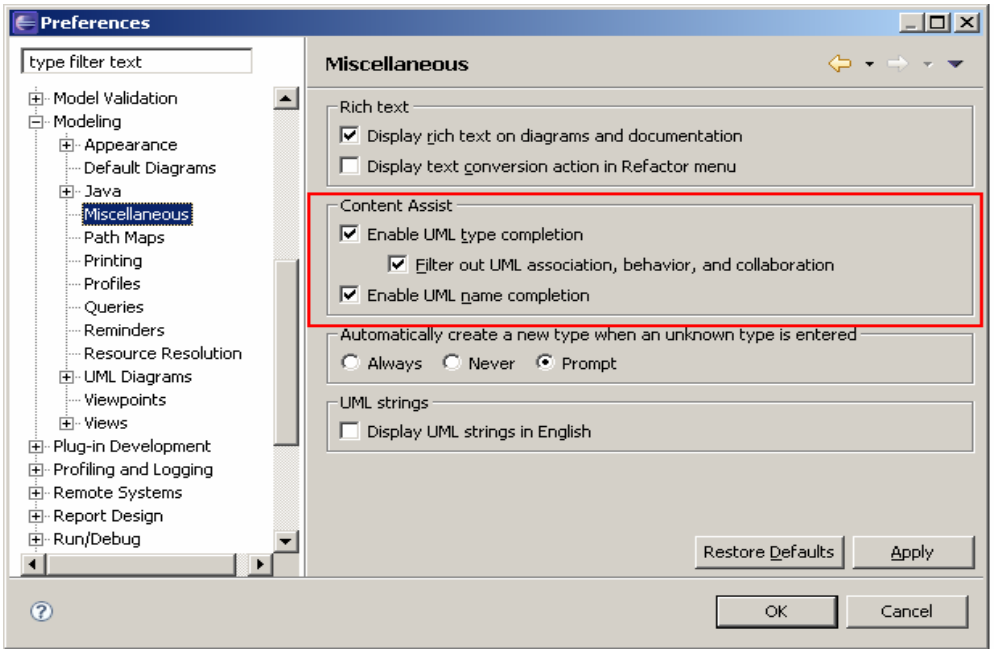
- Added Refactor -> Delete (since RSA 7.0.0.7; RSM/RSD 7.0.5.2)
- Refactor -> Rename of operation looks in *closed models*
- Preview of resources!
- **Extensibility**
  - Support create fragment and absorb fragment participant
  - Added refactoring participant example



Content Assist

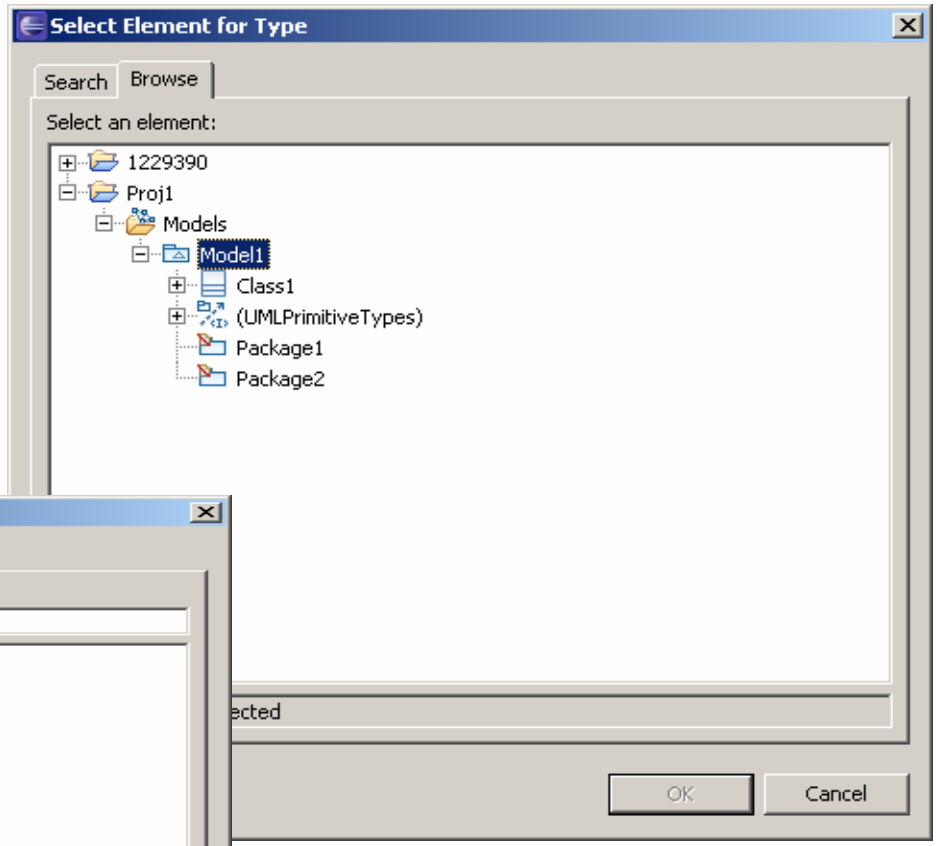
- Display types from model in context and all its imports, including types from closed resources
- Visibility rules are respected
- Certain types are filtered by default (controlled via preference)

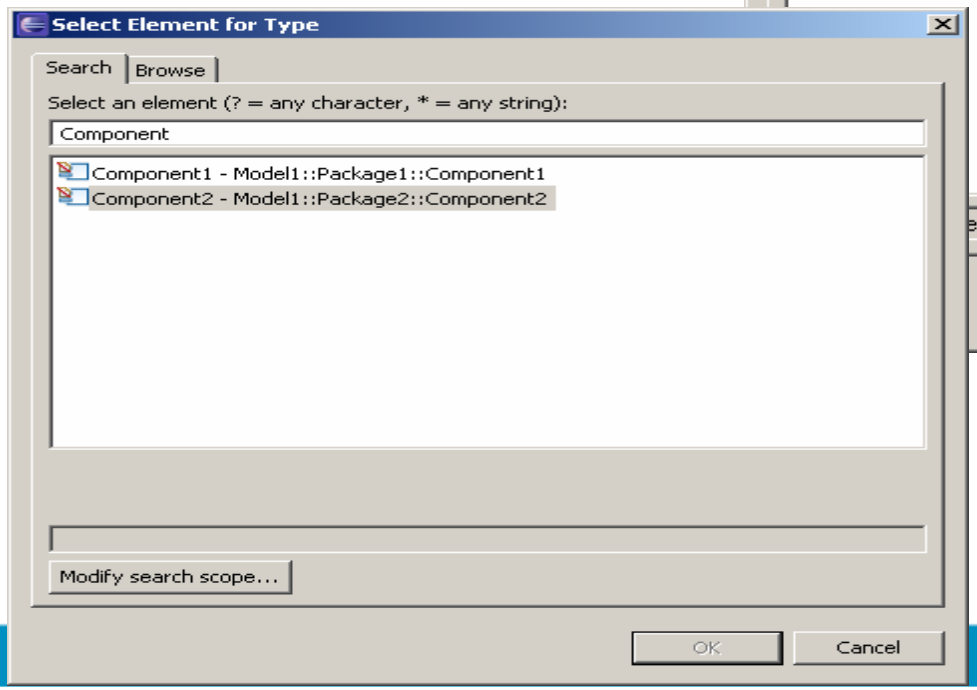




Select Element Dialog

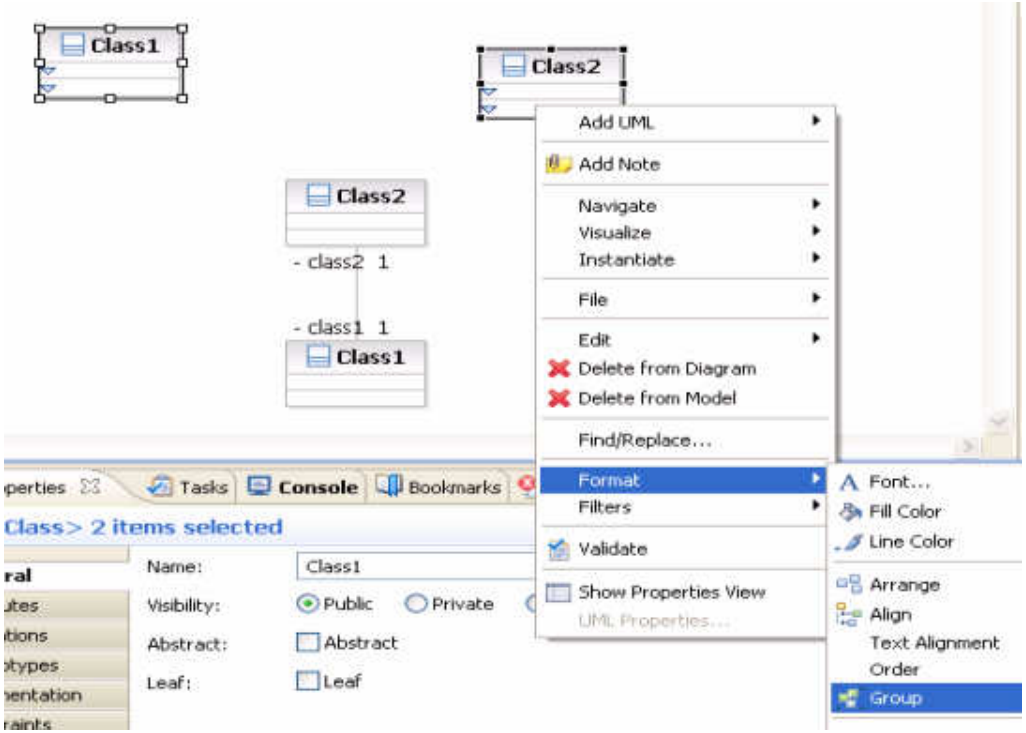
- *Closed model files* are now queried in the search tab, and displayed in the browse tab.
  - Can load the files in the browse tab.
- Improved filter consistency for both tabs





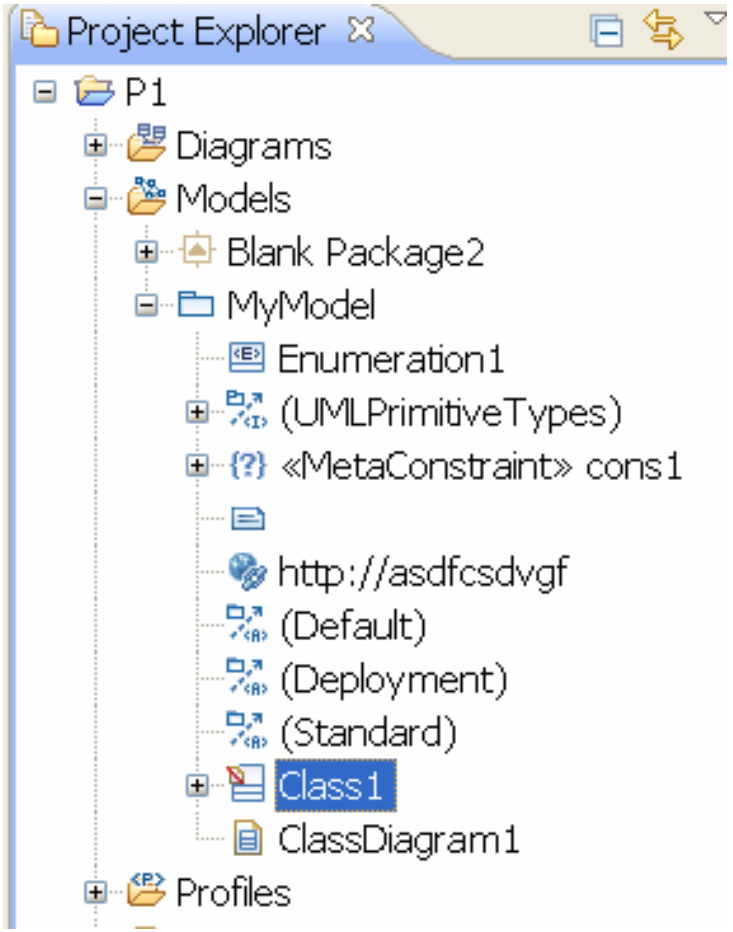
Group / Ungroup support

- Group / Ungroup Support
  - Modeler shapes can be grouped together and operations such as cut/copy, move, delete, can be performed on them.
- Enhanced SRE/SHR support for grouped elements.



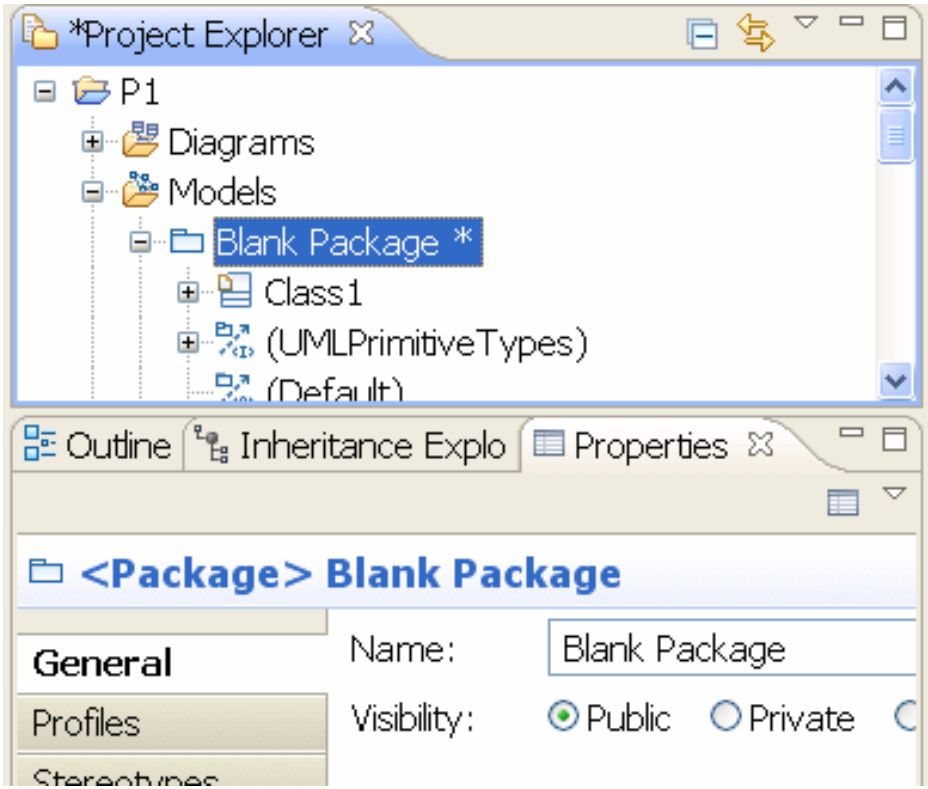
Closed Model Files

- *Closed models and fragments* now opened through expand gesture in Project Explorer



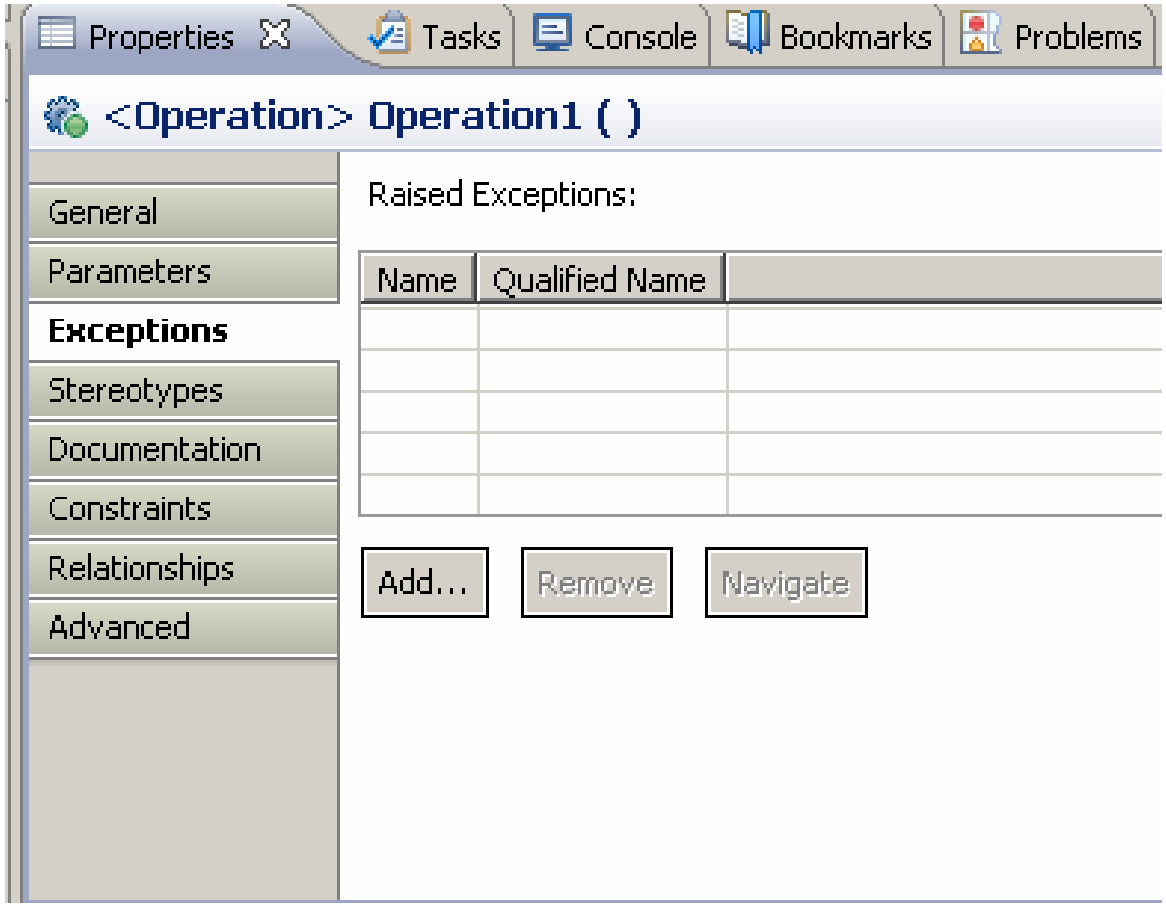
Package as Model Root

- Root package can be *morphed* into a model
- Model can be *morphed* into a root package
- These are sharable packages and can be imported into multiple models!
  - Opens new options for model partitioning



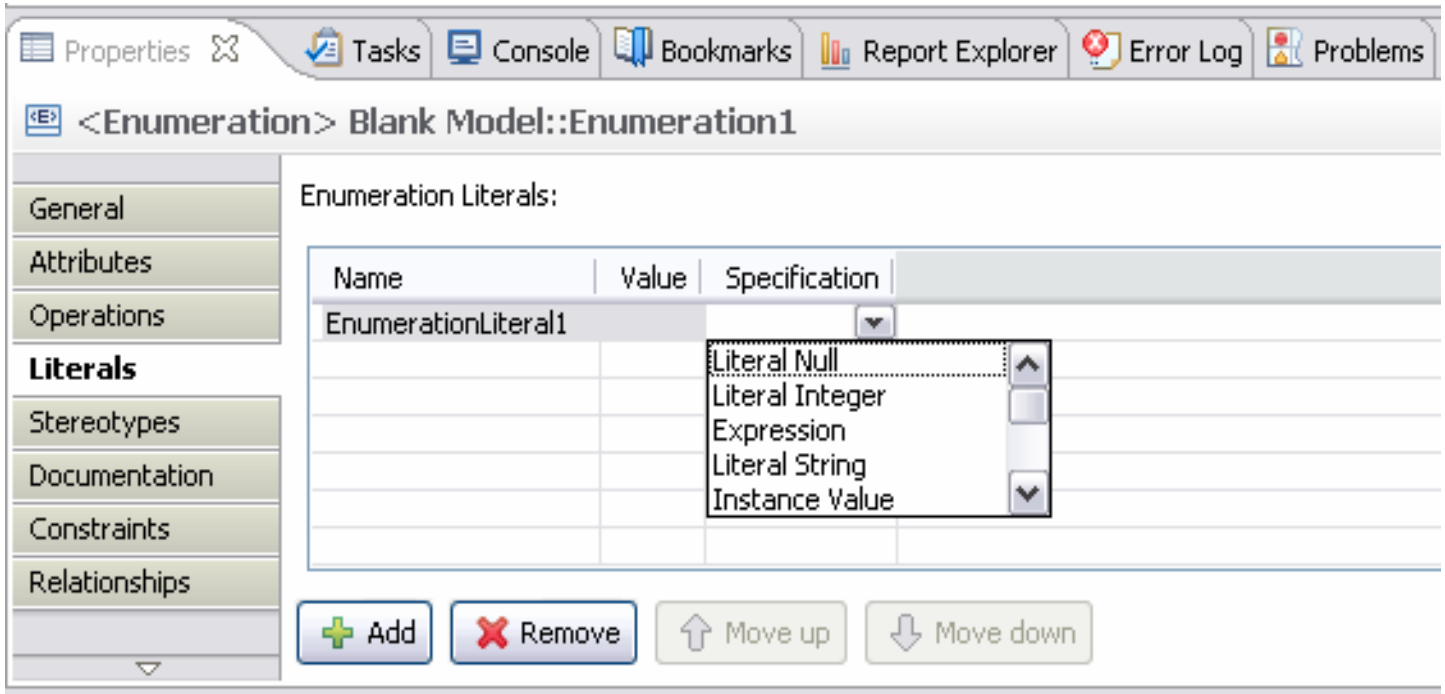
Operation exceptions

- Exceptions can now be easily added to an operation



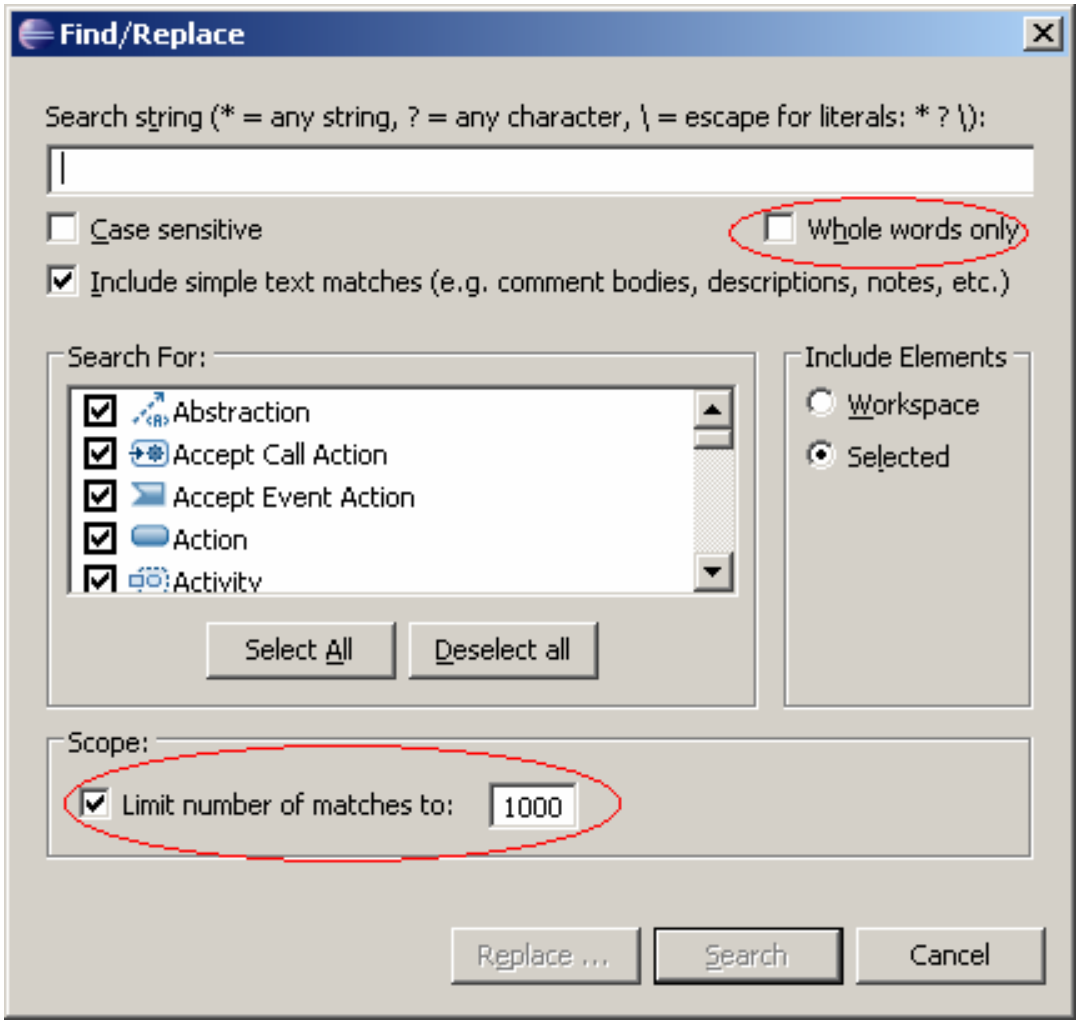
Enumeration Literals

- Simplified the editor
  - Define name, value and specification
  - Supports add, delete and reorder



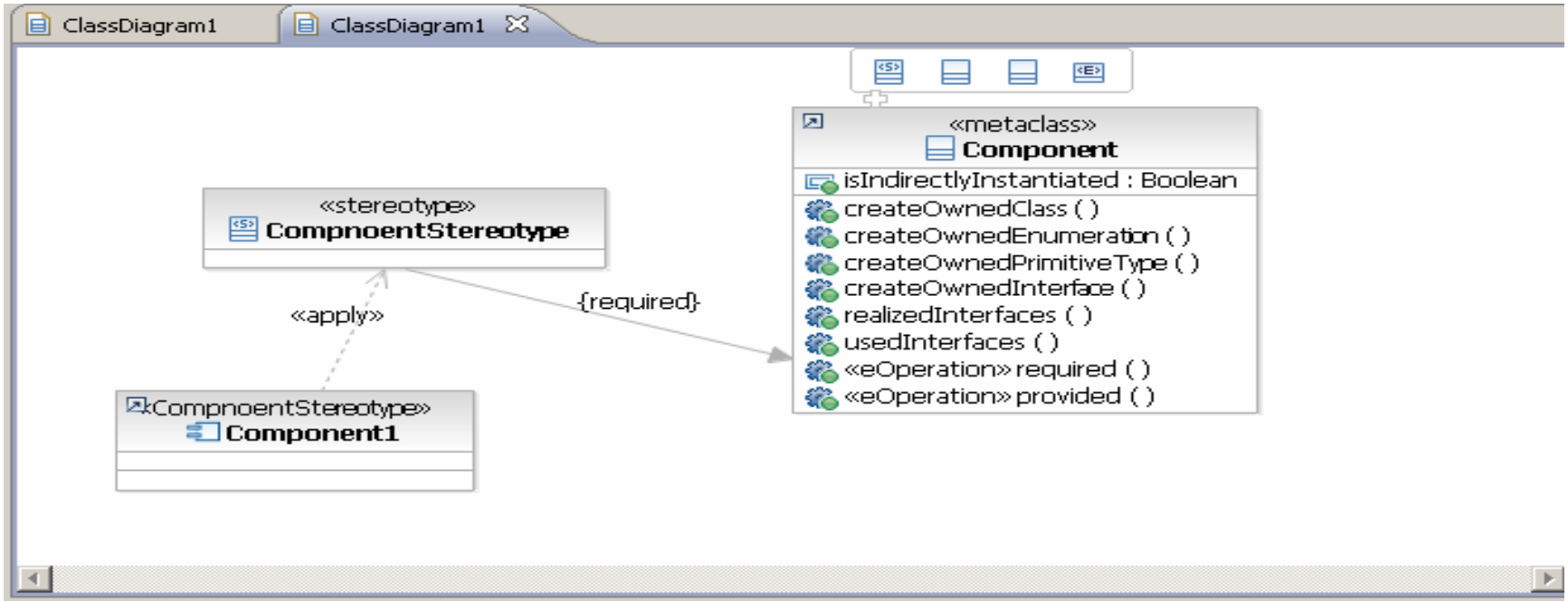
Find and Replace improvements

- Improved Find and Replace
  - "Whole words only" option now supported
  - Can now limit the number of search results, improving performance
  - Search and Replace also improved



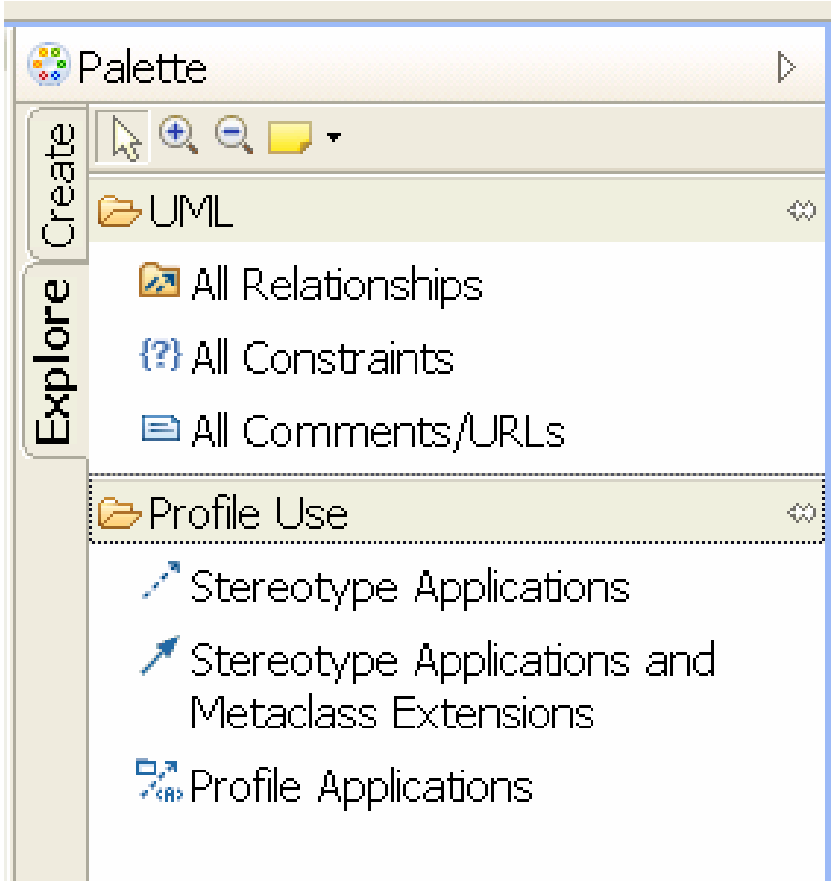
Visualization of Stereotype and Profile Applications

- Can be visualized from palette
- Profile applications can be visualized from Show/Hide relationships dialog
- Stereotype applications can visualize extended metaclasses



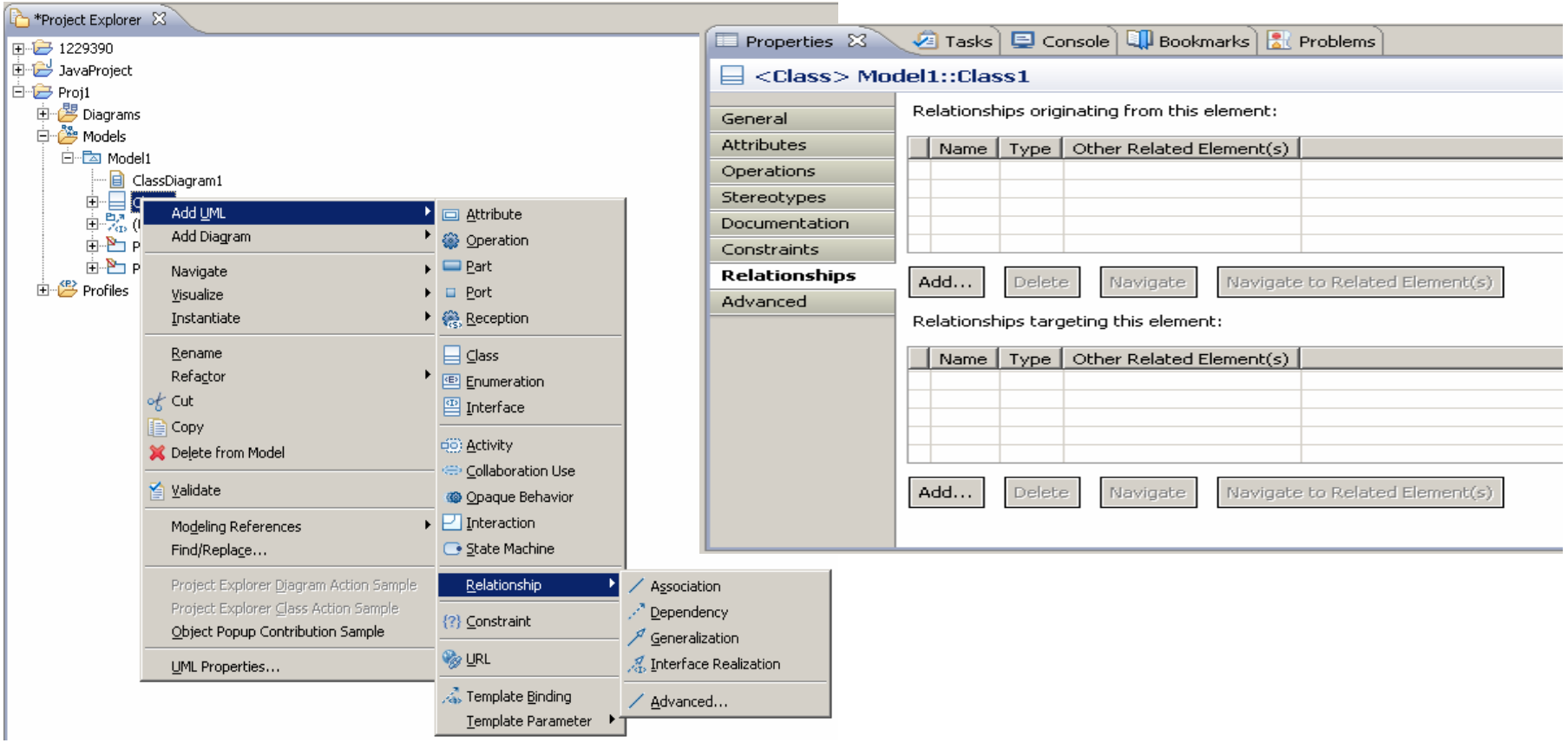
UML Tools for Palette Explore Tab

- Provided tools for the explore tab in palette to query relationships, constraints, and comments / URLs.
- Visualization of stereotype and profile applications also available through these tools



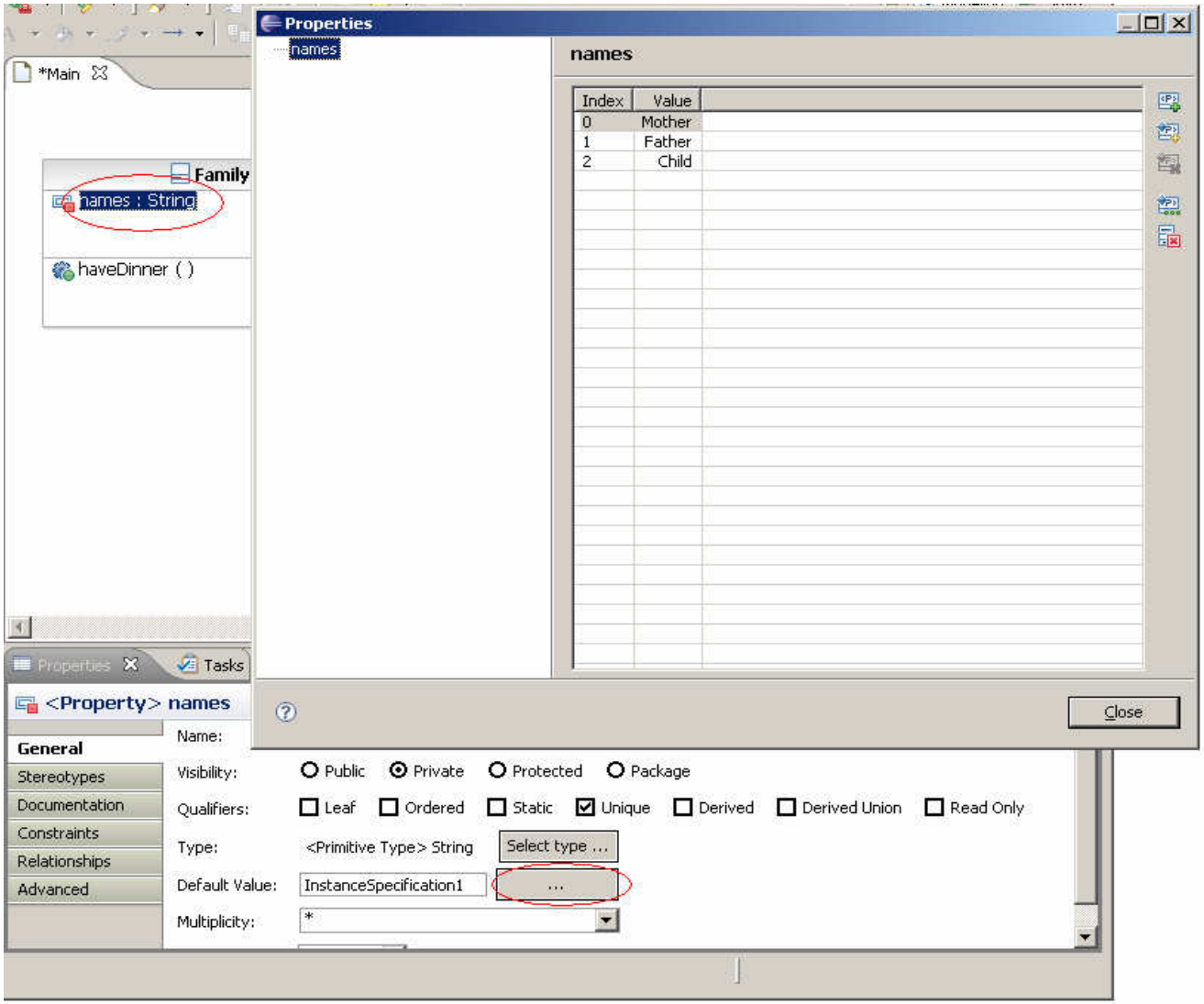
Improved Relationship Usability

- Create relationship directly from Project Explorer
- Element relationships now displayed in Properties View



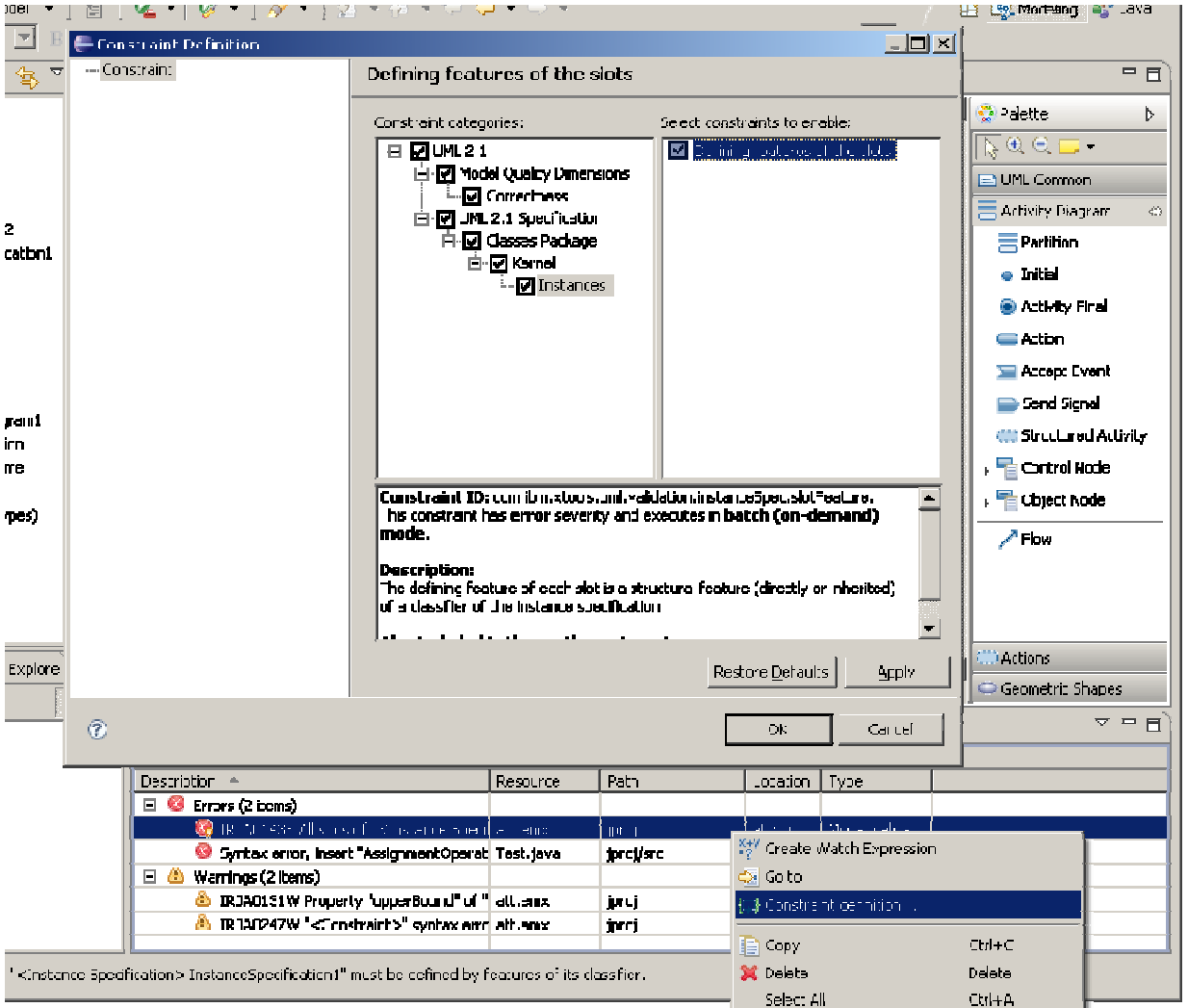
Model Multi-Valued Attributes

- Directly model multi-valued attributes



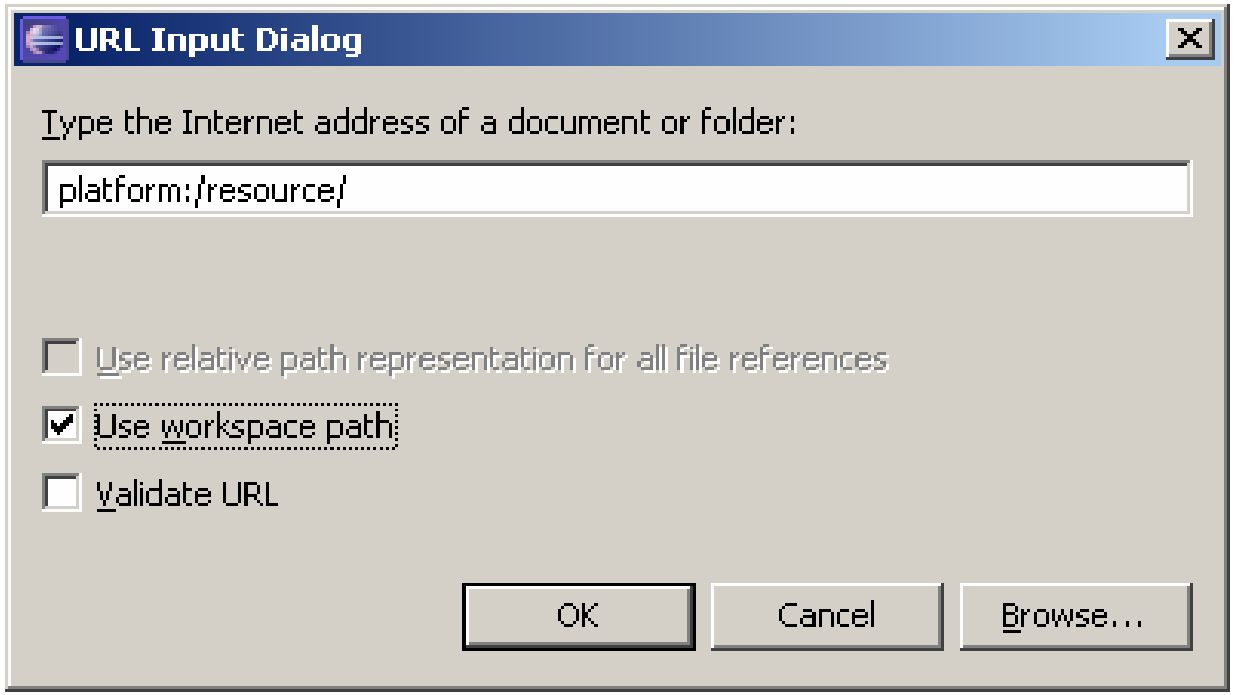
Associate Rule with Validation Failure

- Easily identify a rule causing a validation failure
  - Constraint definition … action
  - From Problems and Tasks view



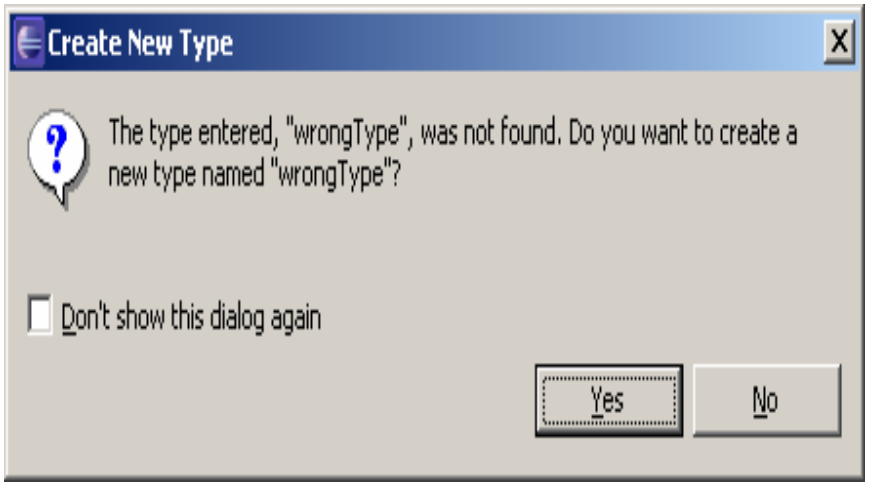
Workspace URL Support

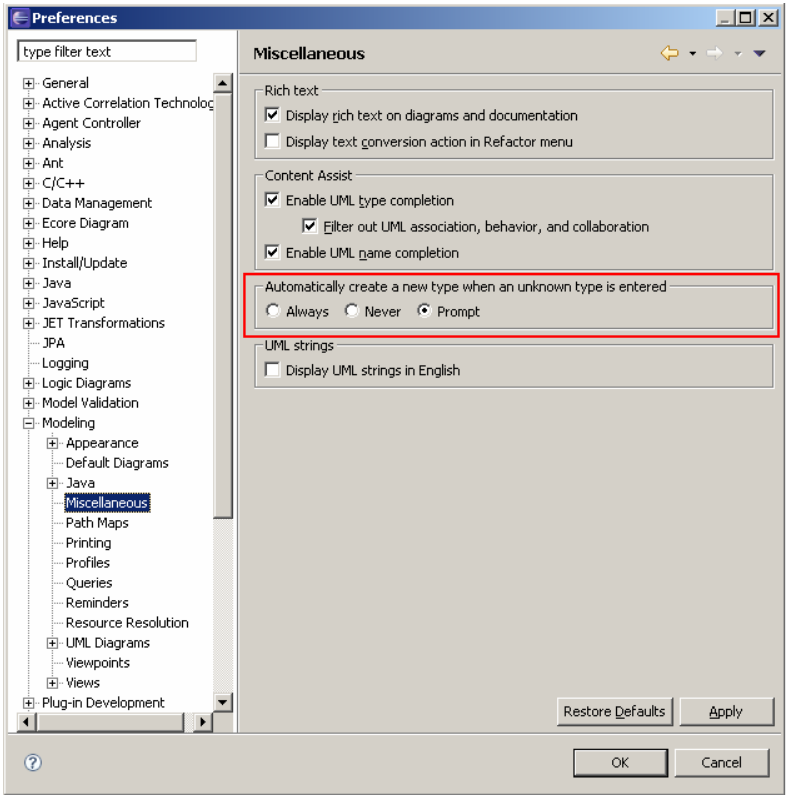
- Can create a URL using workspace format



Automatic type creation

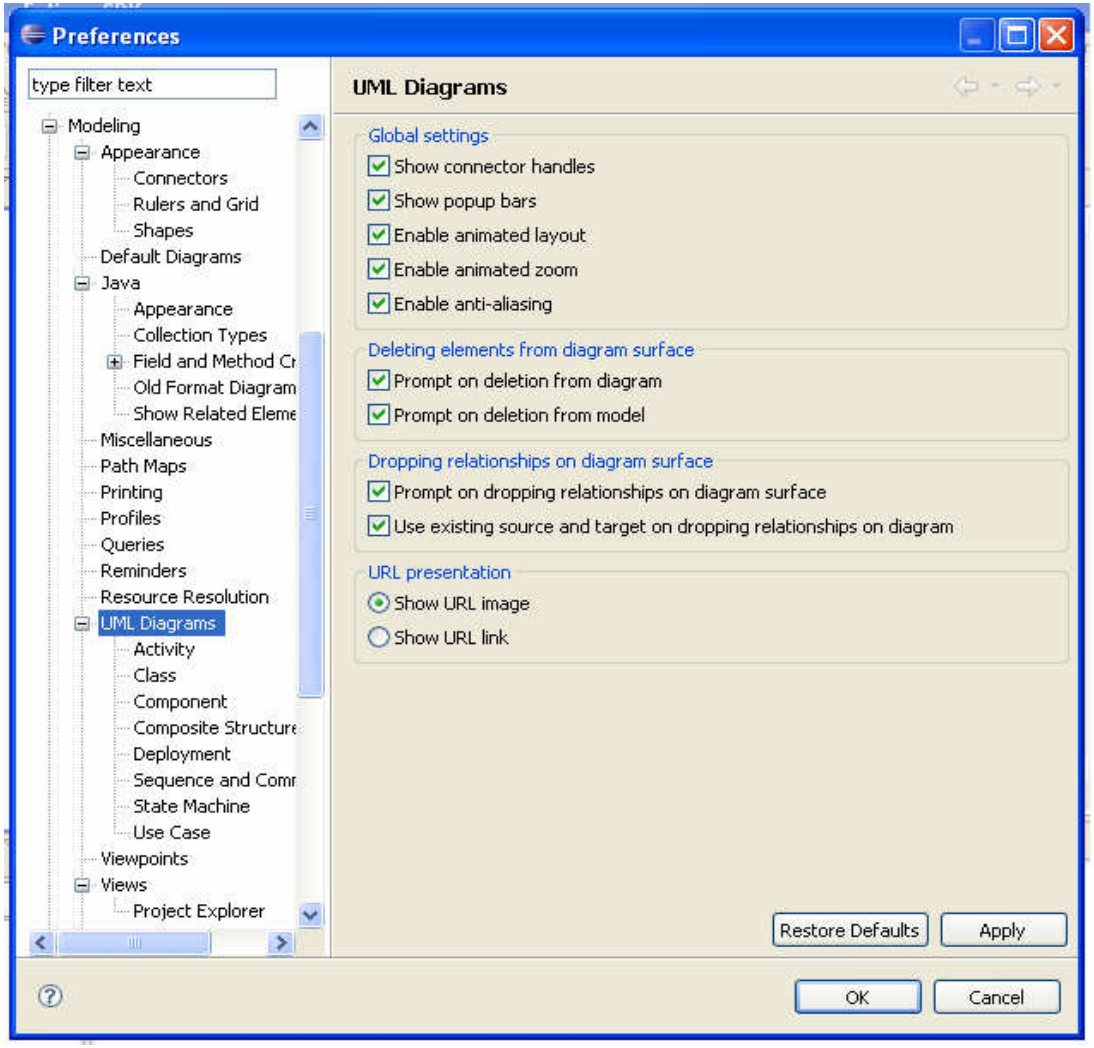
- User is now prompted on automatic type creation when an unknown type is specified. The prompt is controlled via preference.





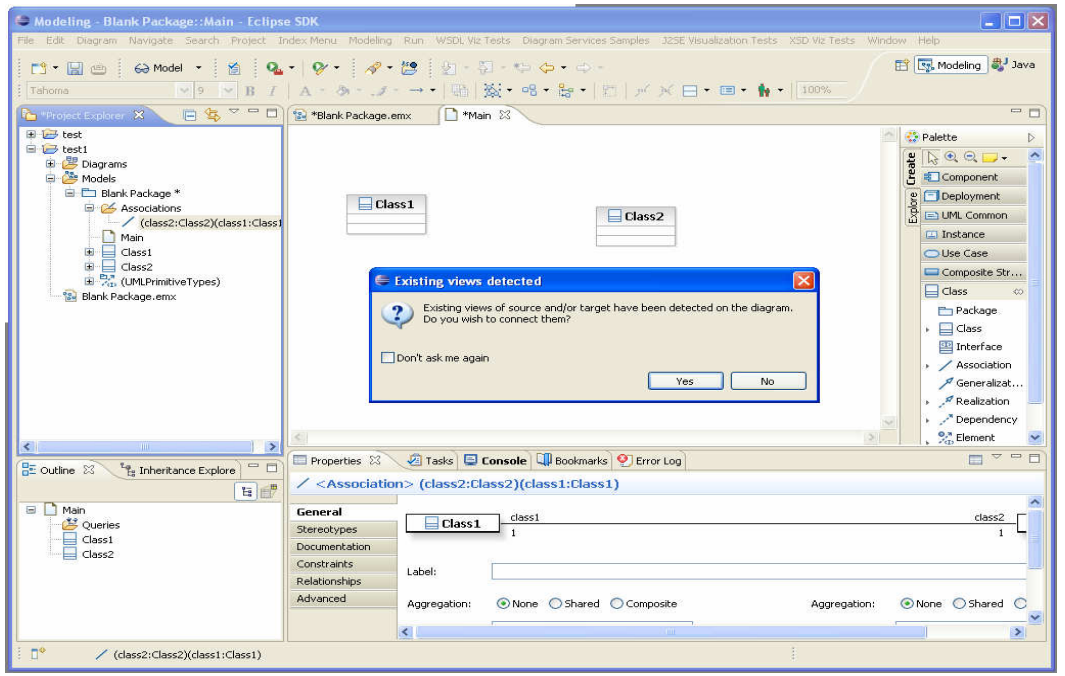
Modeler Preferences

- Reorganized to minimized the number of preference pages, added new display prefs
  - Classifier Compartment
  - Association
  - Node (shape) Compartment
  - Component Compartment
  - Use Case



Use existing views setting

- Use Existing Views
  - When a relationship is dropped into a modeler diagram with an existing source and/or target views, users could choose to use these views or create new ones.
  - This option is controlled by a preference setting.

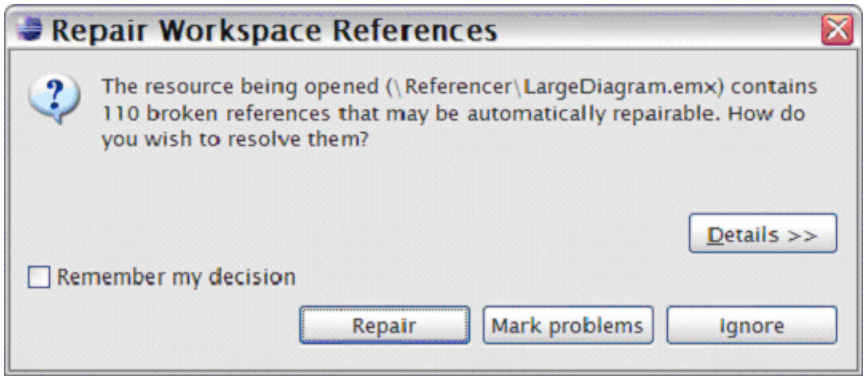


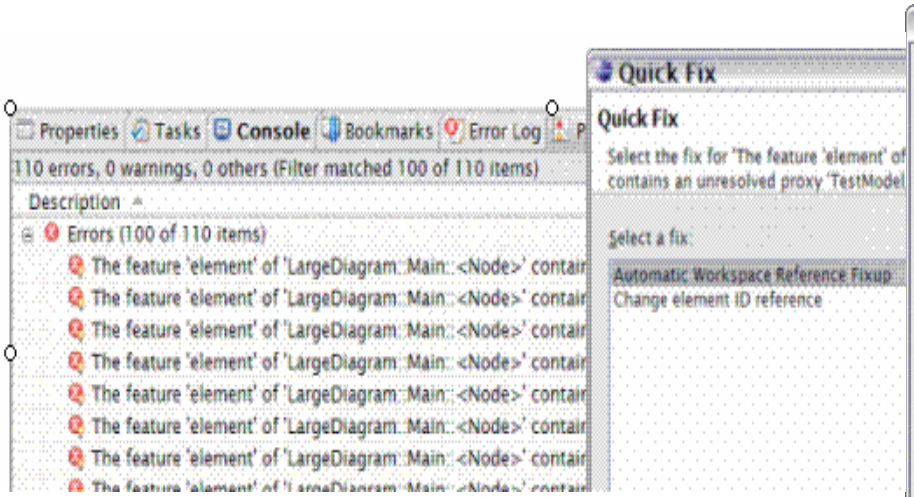
Model Integrity and Management Improvements

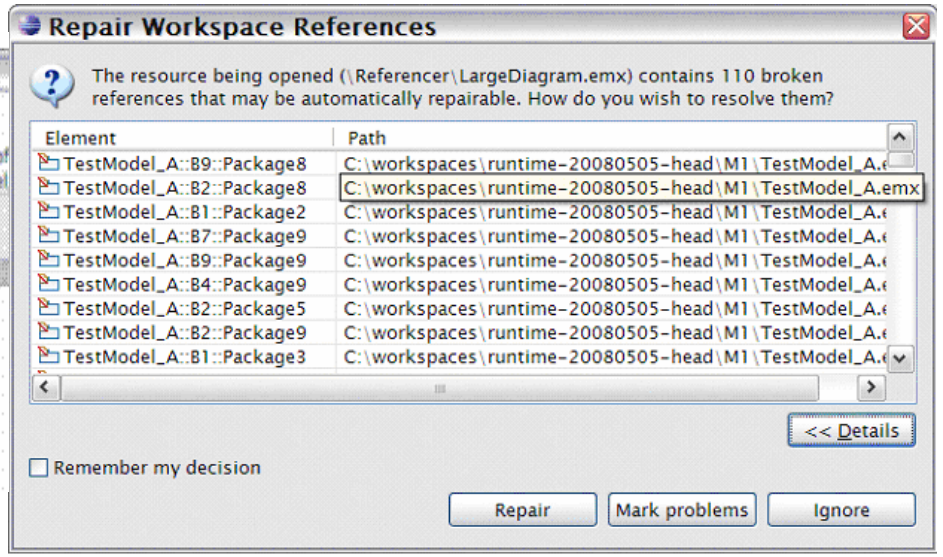
- Automatic reporting and repair of unresolved references
- Repairing or removing broken profile applications
- Model fragment improvements

Automatic reporting and repair of broken references

- New broken references dialog replaces missing resources dialog
- Problems also reported in the Problems View with a quick fix

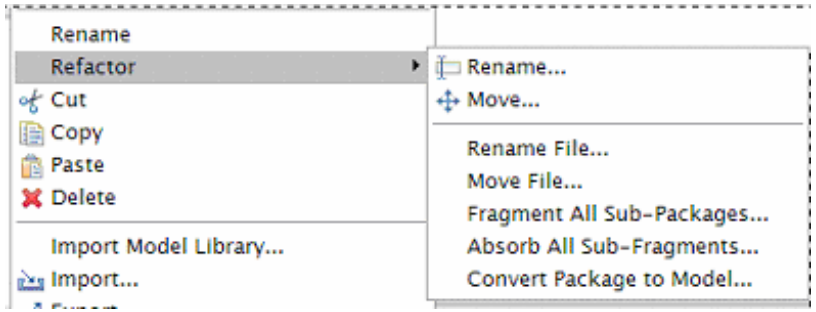


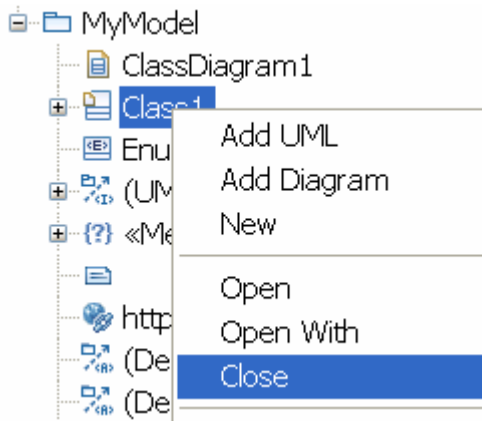


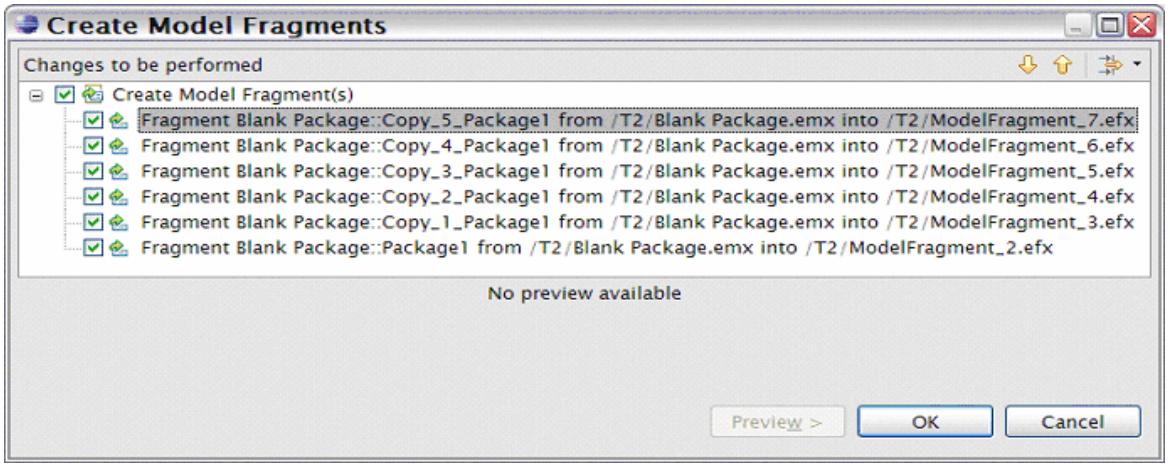


Model Fragments

- Fragment actions are now part of the refactoring menu
- Fast explosion / recombination of model
  - fragment all sub-packages
  - absorb all sub-fragments
- **Fragments can be closed**

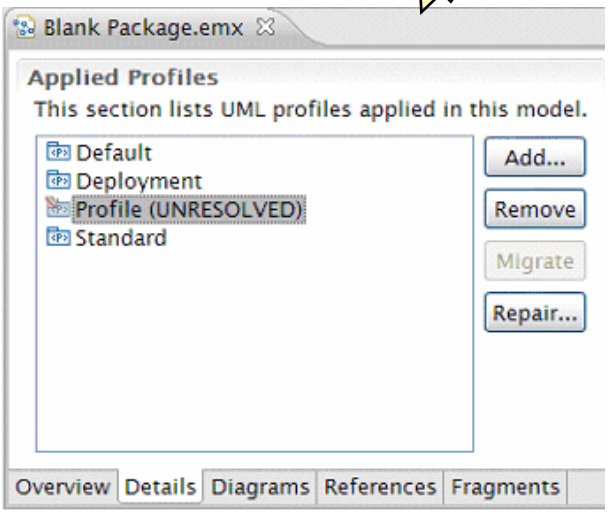


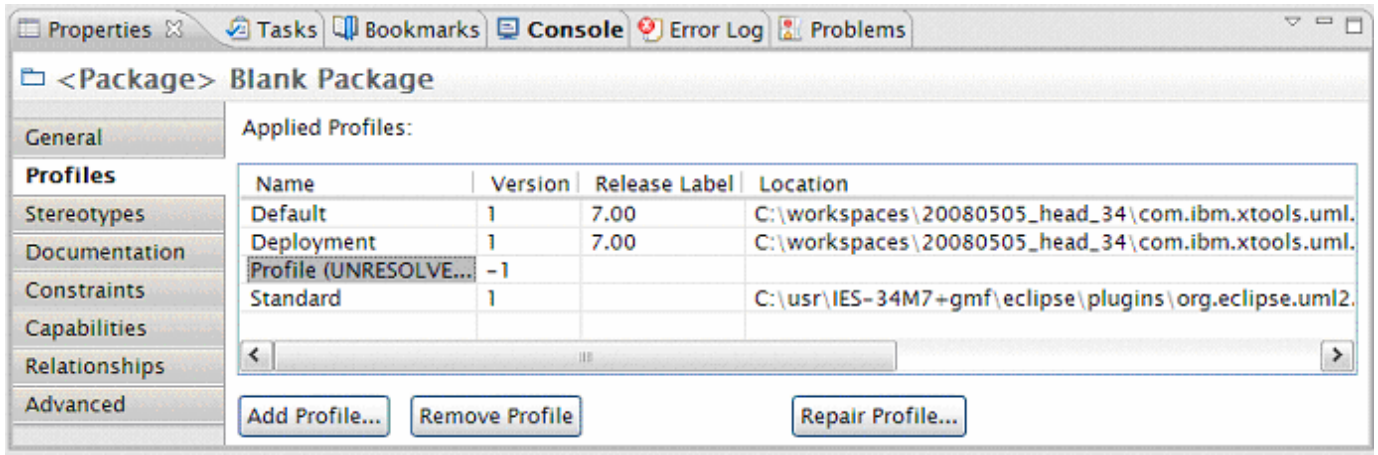




Repair or Remove Broken Profile Applications

- Remove button is now enabled for unresolved profiles
- Repair button is added for unresolved profiles





Sequence Diagram Improvements

- Copy/Paste support
- Refactor support
- Message Arguments UI
- Support for non-Integer types in guard condition
- Gates support

Sequence Diagram Copy/Paste/Refactoring

- Copy/Paste
  - Copy/Paste sequence diagram elements within and between sequence diagrams
  - Copy/Paste Interactions within Project Explorer
  - Context-aware copy operation forming Copy Atomic Set
  - Context-aware paste operation reusing existing lifelines and elements
- Refactoring
  - Refactor sections of a sequence diagram into a new sequence diagram
  - Reference the new sequence diagram through InteractionUse
  - Reusing Sequence Diagram Copy/Paste implementation

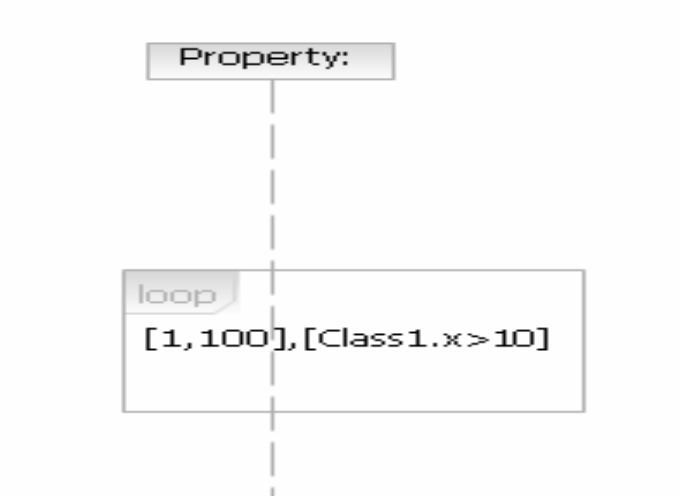
Message Argument GUI Enhancement

- User friendly GUI to specify message argument values
  - Message Property Sheet: add new property section for message arguments and values
  - Add and Delete argument values
  - Change argument type, For example, from LiteralString to LiteralNull
  - Change argument value: different cell editor for argument with different types (LiteralString, LiteralInteger, LiteralBoolean etc.)
  - Reorder existing argument values
  - Populate argument values for all message arguments

- Enhanced existing message signature and argument rendering on Sequence Diagram.

Editing Guard Condition in Sequence Diagram

- Added support to specify boolean expression in the guard condition of loop combined fragment
- Added support to specify opaque expressions for minint and maxint in the guard condition of a loop combined fragment

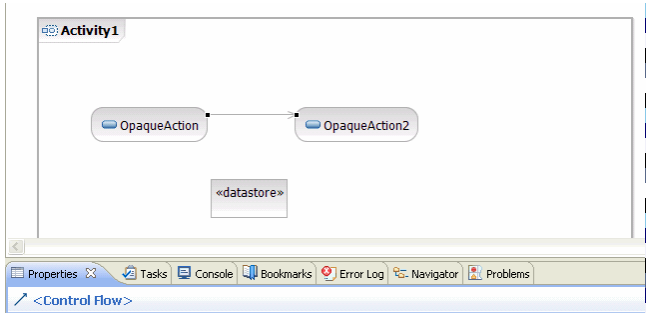


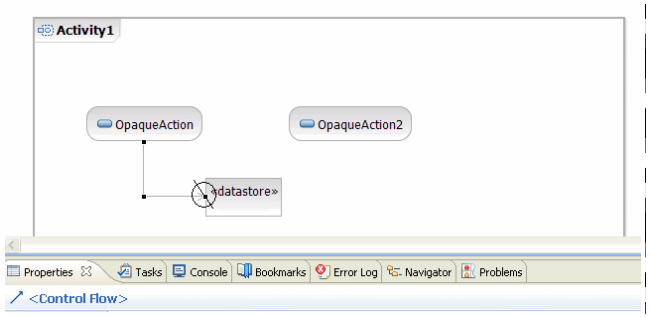
Activity Diagram Improvements

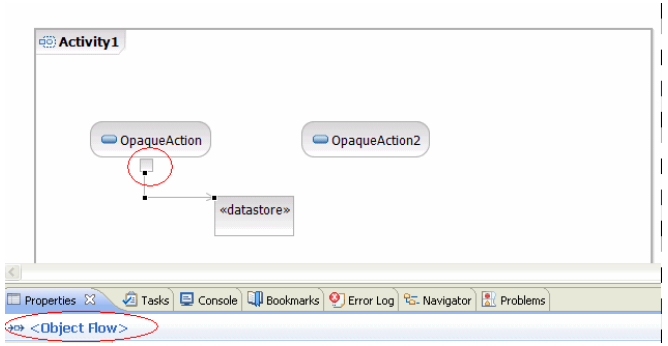
- Generic Flow (Object vs Control Flow)
- Hiding Pins
- Layout Enhancements

Generic Flow

- Added a Flow tool to replace separate control and object flow tools
- Flow tool automatically creates the correct flow type
- Reorient flow between object nodes, control nodes and actions by
  - automatically completing the context (i.e. adding pins)
  - morphing object flows and control flows where required
    - note that compare and merge understand this change type







Hiding Pins

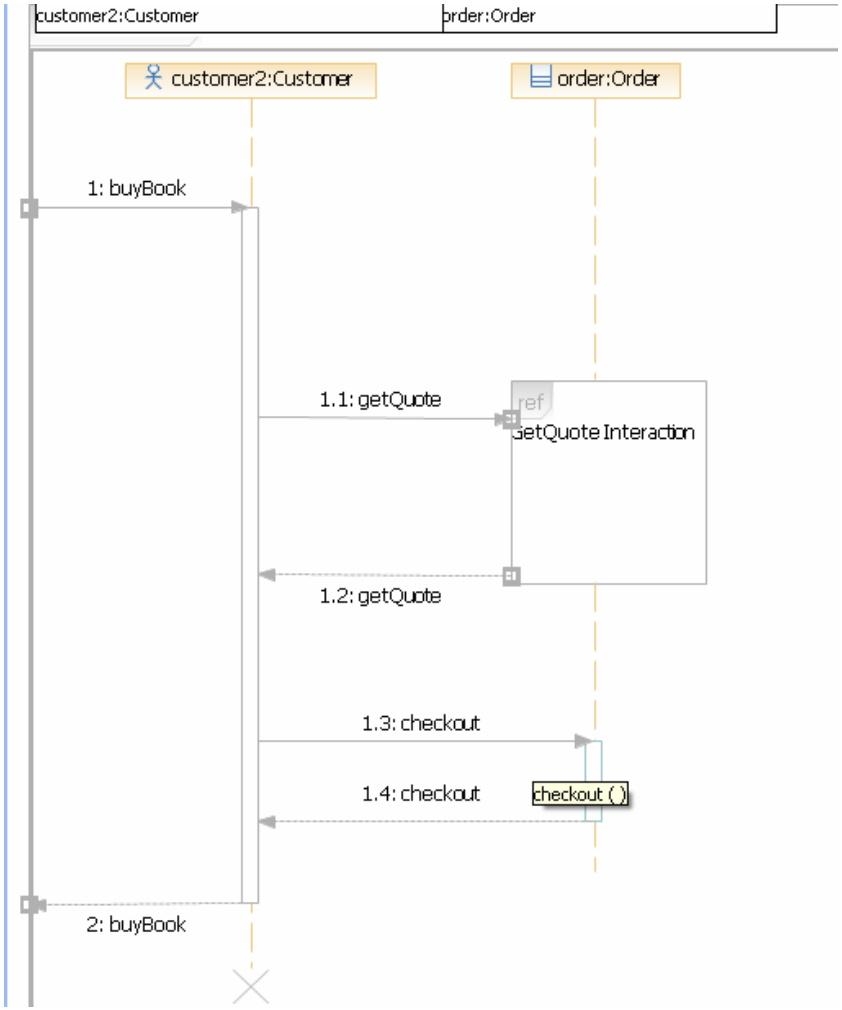
- Allowed users to hide pins on actions inside activity diagrams:
  - The complexity of the diagram is reduced by not showing pins on actions.
  - The user can choose to connect flows to existing hidden pins.
  - The user can toggle showing pins per activity diagram.

Layout enhancements

- Enhanced the Arrange All and Arrange Selection behavior in activity diagrams:
  - Invoke **Arrange Selection** on a selected partition to arrange only the elements inside that partition
  - Invoke **Arrange All** to arrange the entire activity diagram independently from the selected object
    - **Arrange All** produces better results as elements do not overlap partitions.

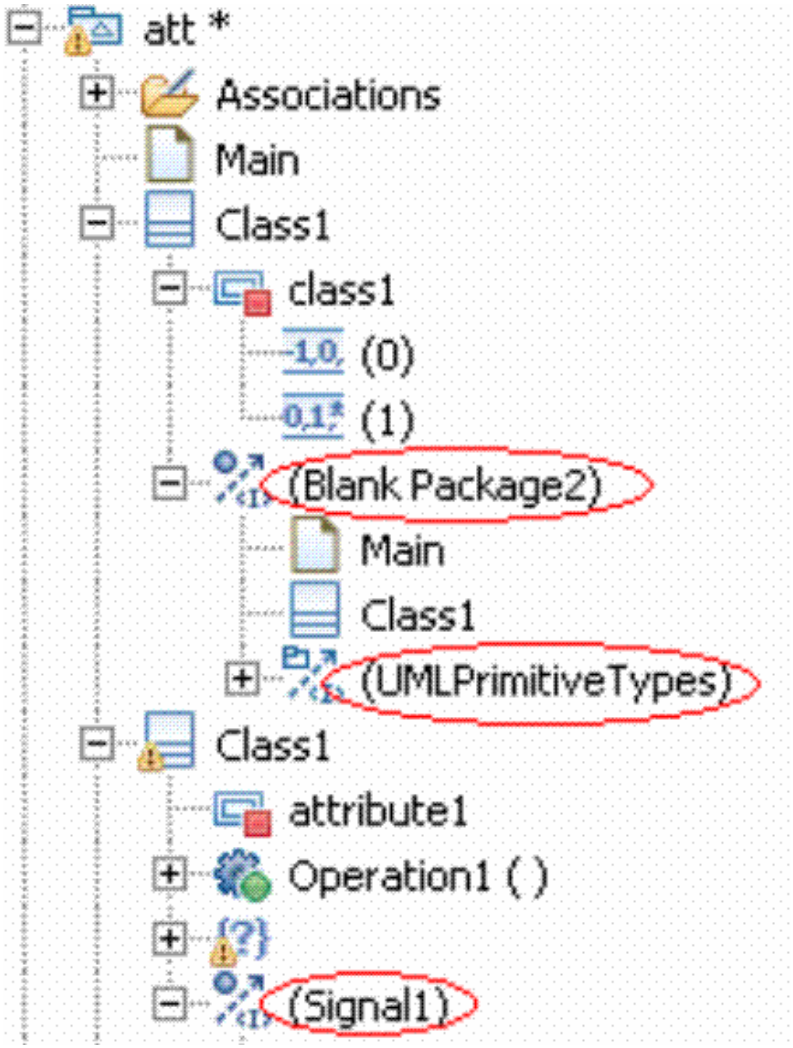
Sequence Diagram Gates

- Editor Support
  - Formal Gate on Interaction Frame
  - Actual Gate on Interaction Use
  - Create and Delete Gated Message
  - Move Gated Message Up and Down
  - Reorder Gated Message
  - Number Gated Message
- Validation Rules
  - Formal Gate and Actual Gate Matching
  - Dangling Gate



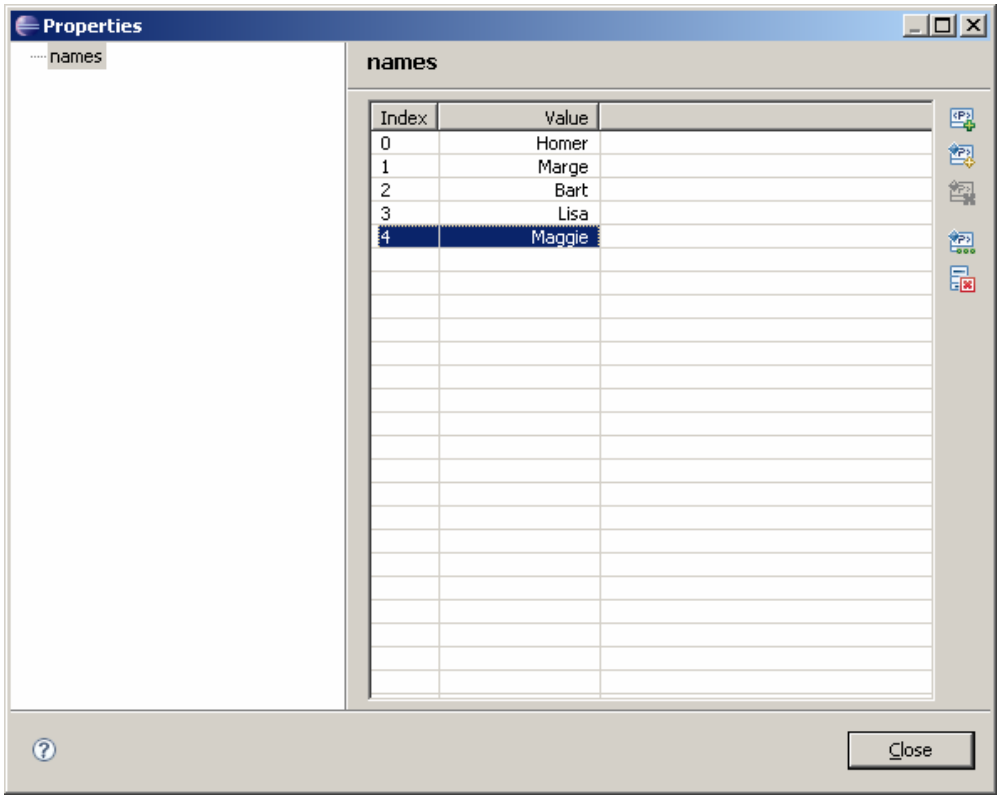
Expand in Place

- Expand in Place
  - Browse package imports and element imports directly from the Project Explorer or Select Elements Dialog
  - Acts like a shortcut
  - More convenient than having to open multiple models



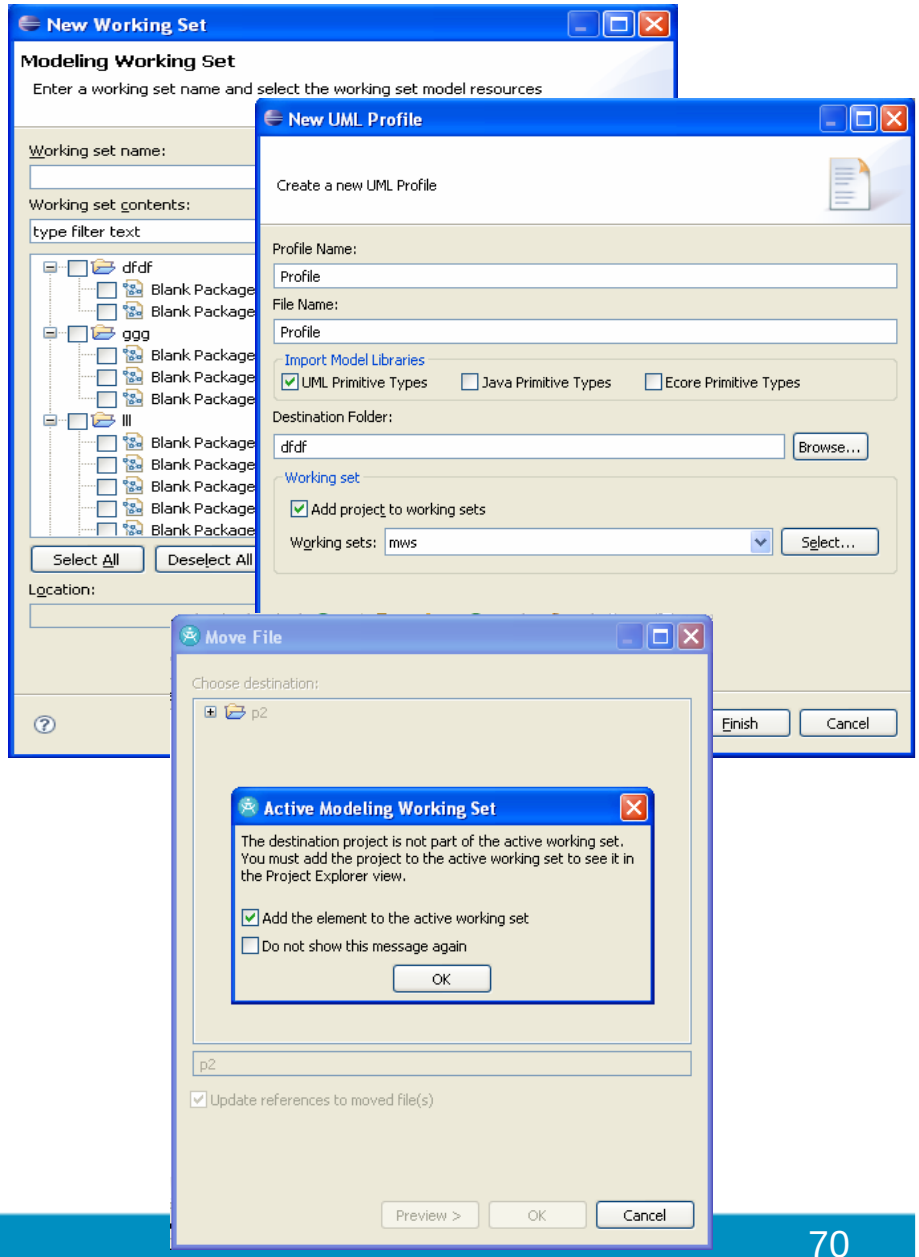
Working With Slot Values

- Enhanced support for UML primitive types
- Extra delete functionality (“deep” delete and “clear the slot” delete)
- Multi-line text editing now supported
- Bug fixes



Modeling Working Sets

- Add new or existing models to Modeling Working Sets
  - New Model/Profile Wizard
  - New Profile Project Wizard
  - New Modeling Working Set
- Support for Model Refactoring like Move File to Project to fix up the working sets



RSx Performance Improvements…

- 6% - 15% memory use reduction
  - larger models experience the higher percentage of memory savings
- All indexing operations run 20-50% faster
- Improved UI Response
- Ability to manually unload fragments facilitates reduction in runtime memory use
- 50% memory reduction time during startup
- Heap size increased from 1.5GB to ~2GB
  - handles much larger models

AM03

What‘s New in RSA and RSM 7.5

Reporting and Traceability

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

IBM

Rational. software

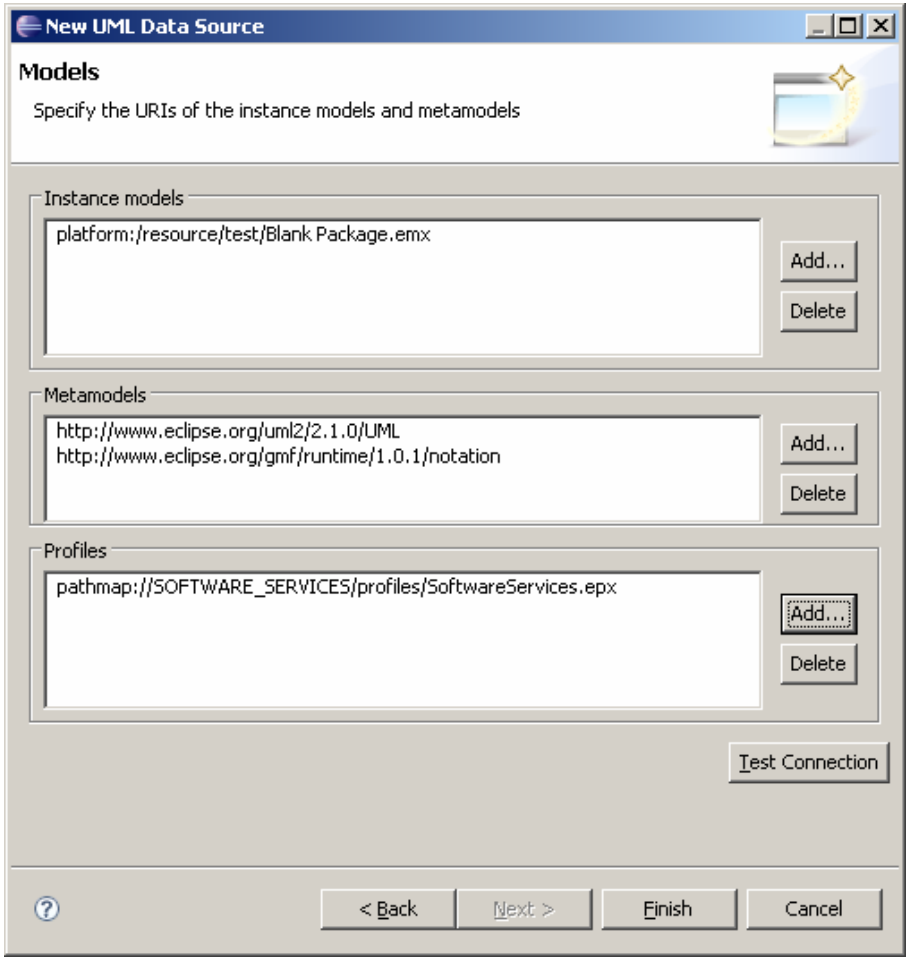
May 2008

Reporting & Traceability

- Birt
  - Improved ODA driver implementation
    - Performance
    - Updated row and column mapping pages
    - Support parameterized sets
  - Launch configuration
- WebPub
  - Interactive tree (dojo toolkit 1.1.0)
  - Property value links
- Traceability
  - Requirement & java type queries

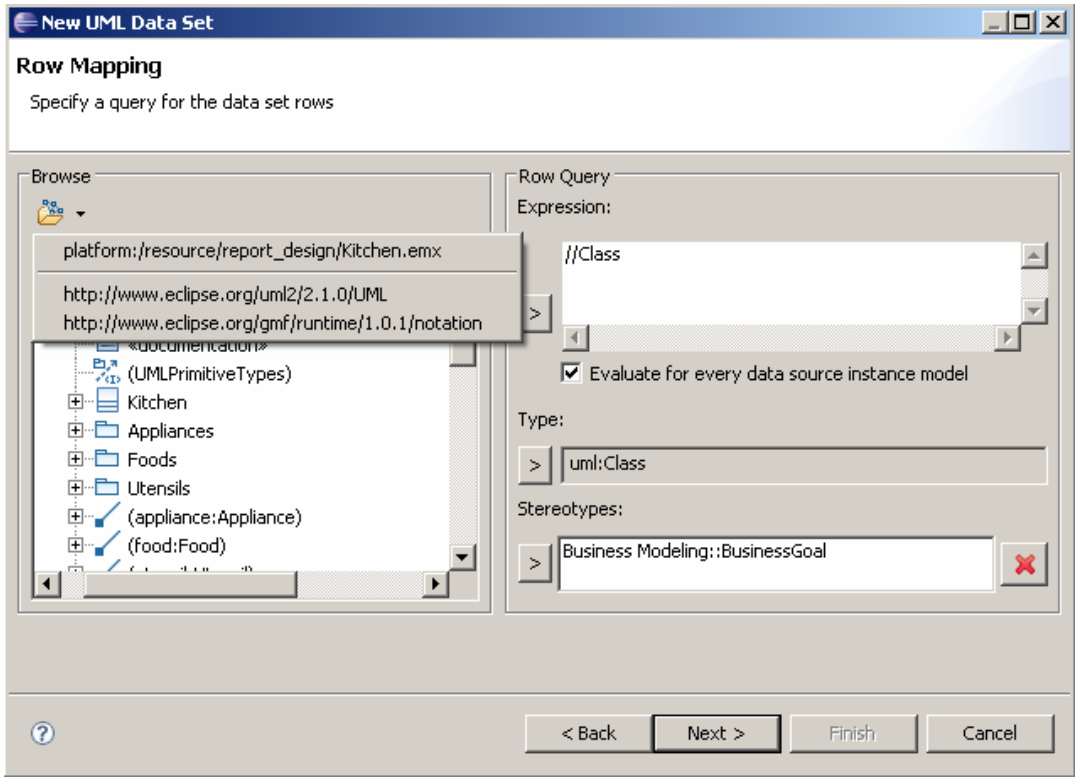
Improved EMF / UML ODA Drivers

- Improved layout, performance and usability of Data Source Wizard
  - Separate meta-model and profile sections
  - Reference meta-models by nsURI vs. ecore path
  - Faster ‘Test Connection’



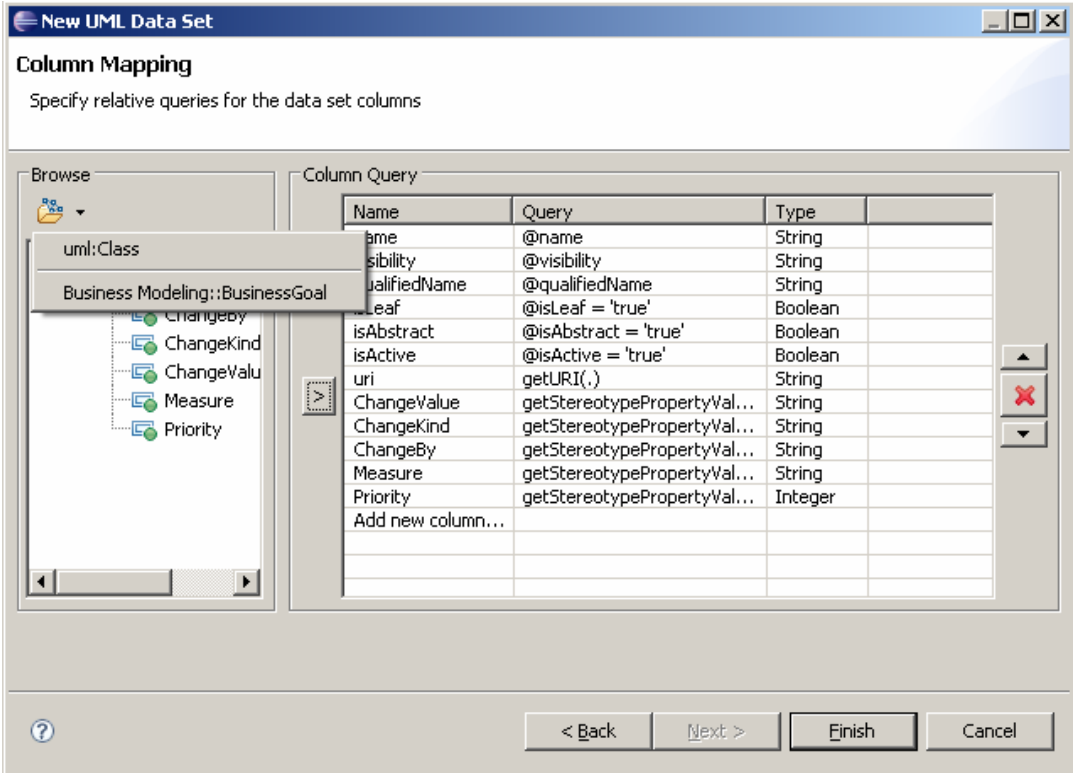
Improved EMF / UML ODA Drivers

- Improved layout, performance and usability of Data Set Row Mapping Page
  - Browse models on demand vs. all at once
  - Customized browse tree for every kind of model
  - Improved XPath proposals



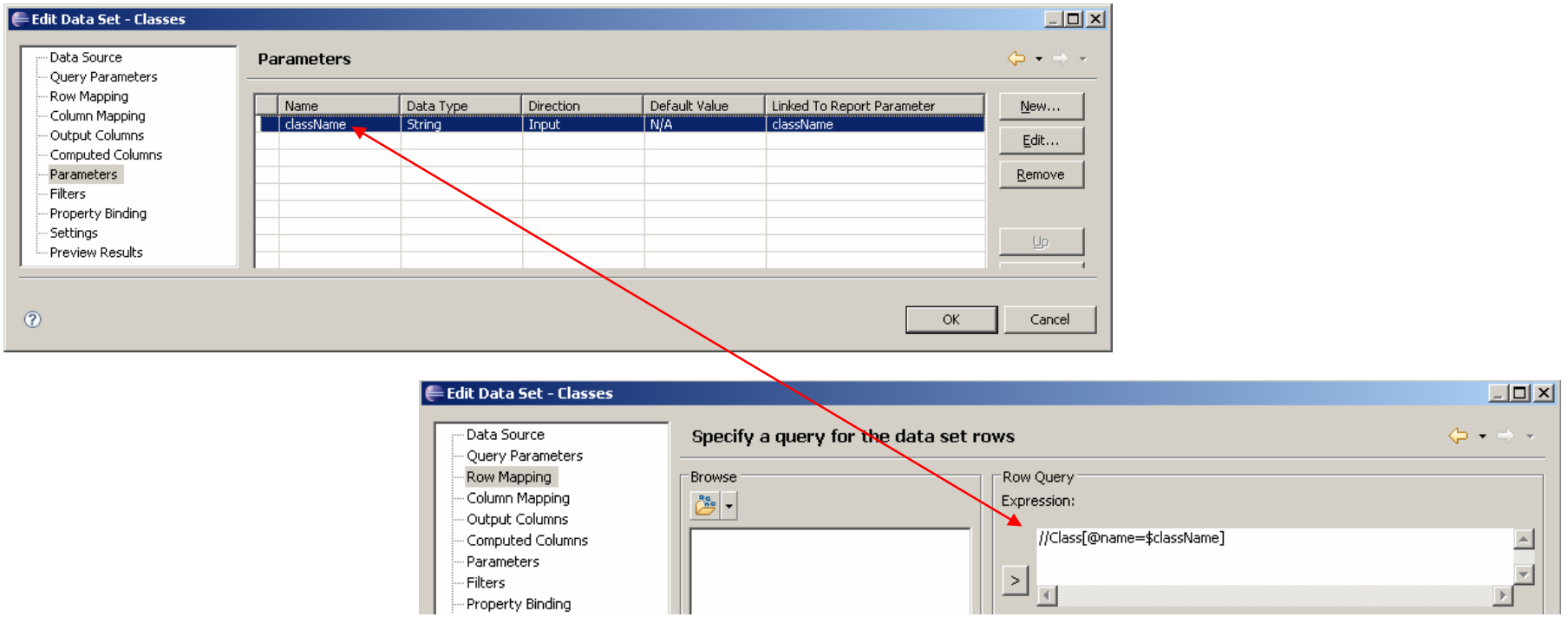
Improved EMF / UML ODA Drivers

- Improved layout, performance and usability of Data Set Column Mapping Page
  - Browse types on demand vs. all at once
  - Customized browse tree for types and stereotypes
  - Improved XPath proposals



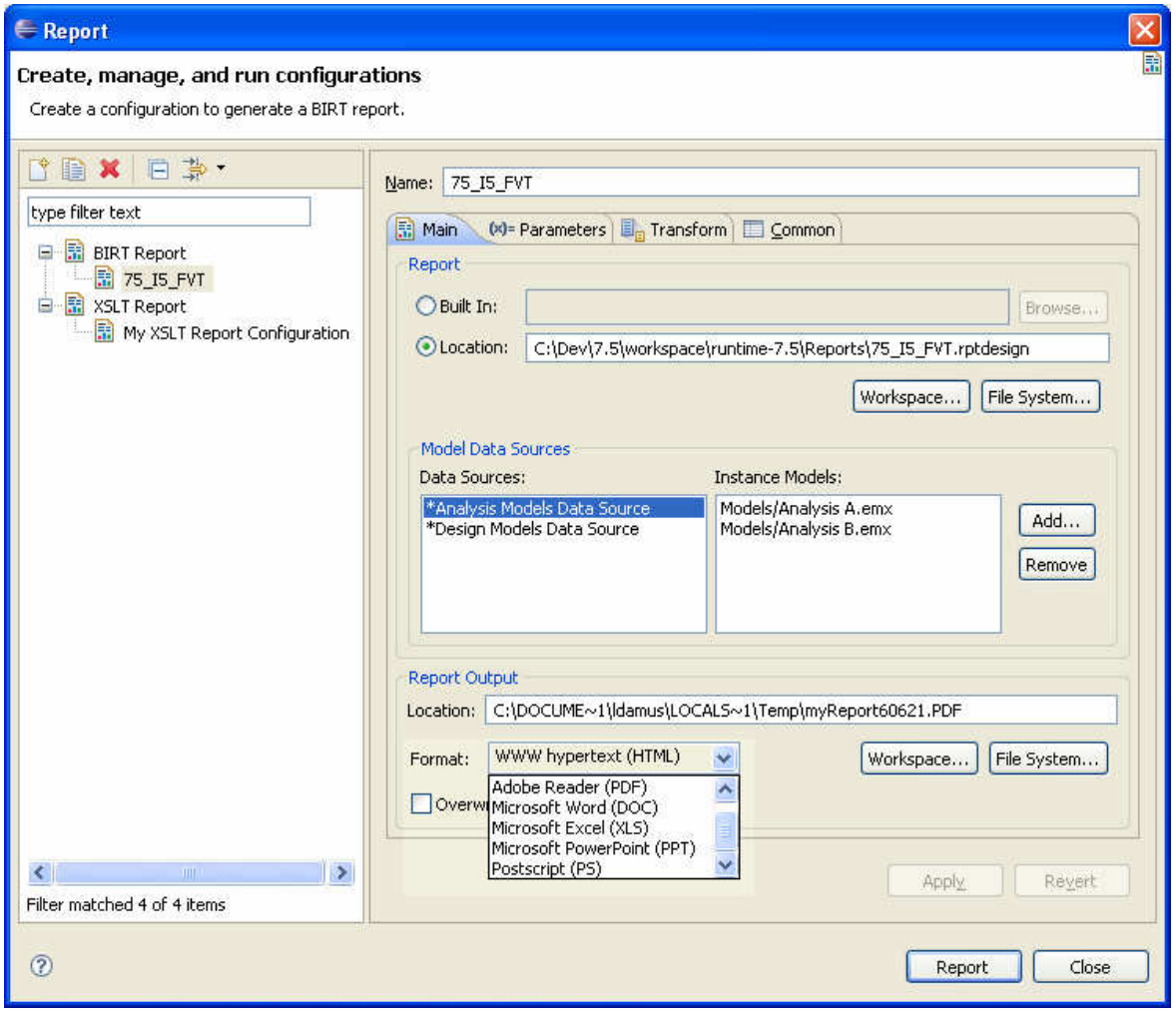
Improved EMF / UML ODA Drivers

- Define and include parameters in data sets and bind the parameters from columns in other data sets or from report parameters



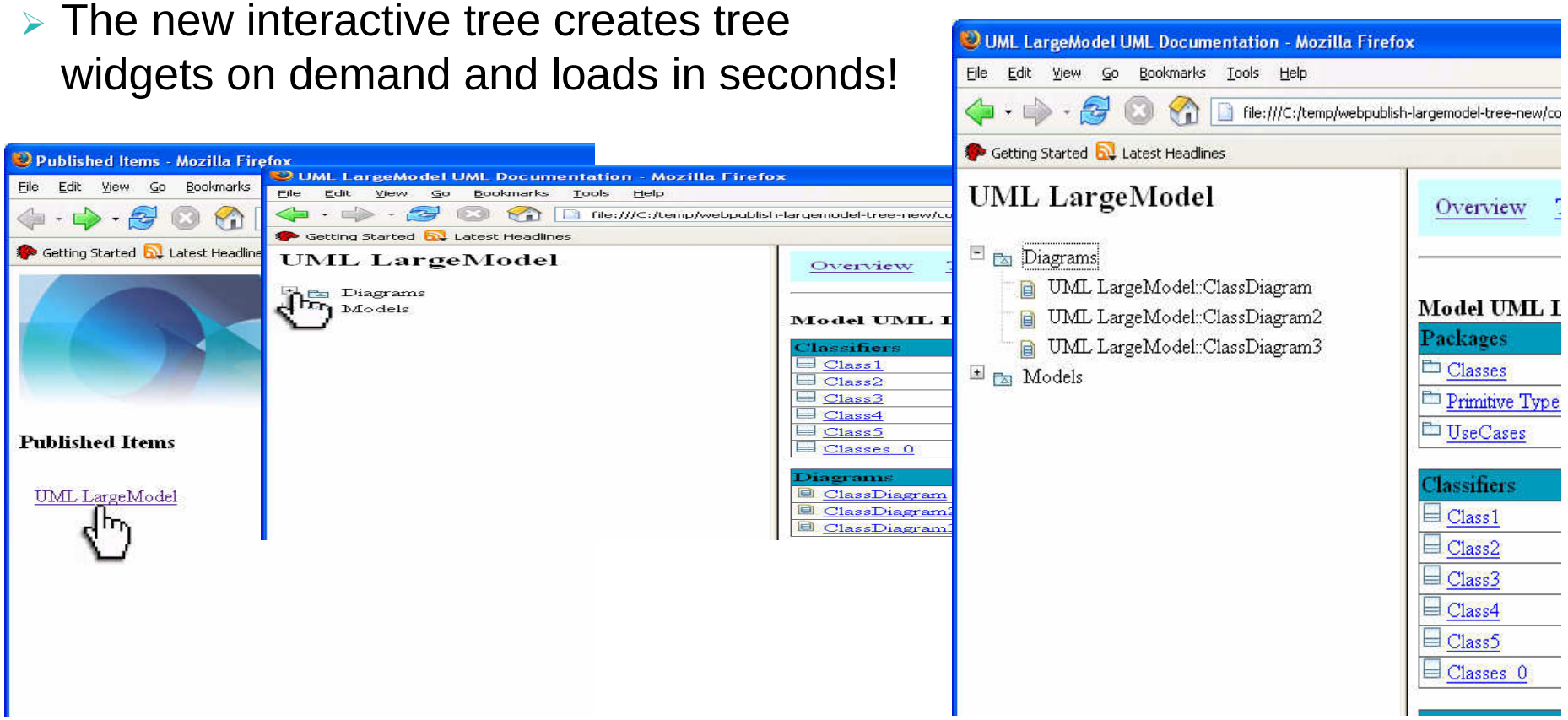
Model Report Launch Configurations

- BIRT and XSLT
- Persist settings such as report parameters and input models
- Generate with data from multiple models and data sources (BIRT)
- Report in word, PowerPoint, excel and postscript formats (BIRT)



Web Publish: Scalable Interactive Tree

- **Dojo toolkit 1.1.0**
- The new interactive tree creates tree widgets on demand and loads in seconds!



Web Publish: Property Value Links

- Property value links offer a better browsing experience…
  - link an attribute to its type
  - link a lifeline to the type it represents

Datatype	
Default	
Default Value	
Lower Value	<Literal Integer> 0
Multiplicity	*
Name	contacts
Name Expression	
Namespace	<Class> AddressBook
Opposite	
Owner	<Class> AddressBook
Owning Association	
Owning Template Parameter	
Qualified Name	Design::addressbook::AddressBook::c
Stereotype	
Template Parameter	
Type	<Class> Contact
Upper	*
Upper Value	<Literal Unlimited Natural> *
Visibility	Private

Overview Top Package Element

Summary: Attributes Operations Specifications Properties Detail: Attribute

addressbook

Class AddressBook

Attributes
<Class> Contact

Operations
addContact
removeContact

etc.

Specifications
Specification
Remove contact
Add contact

Properties:

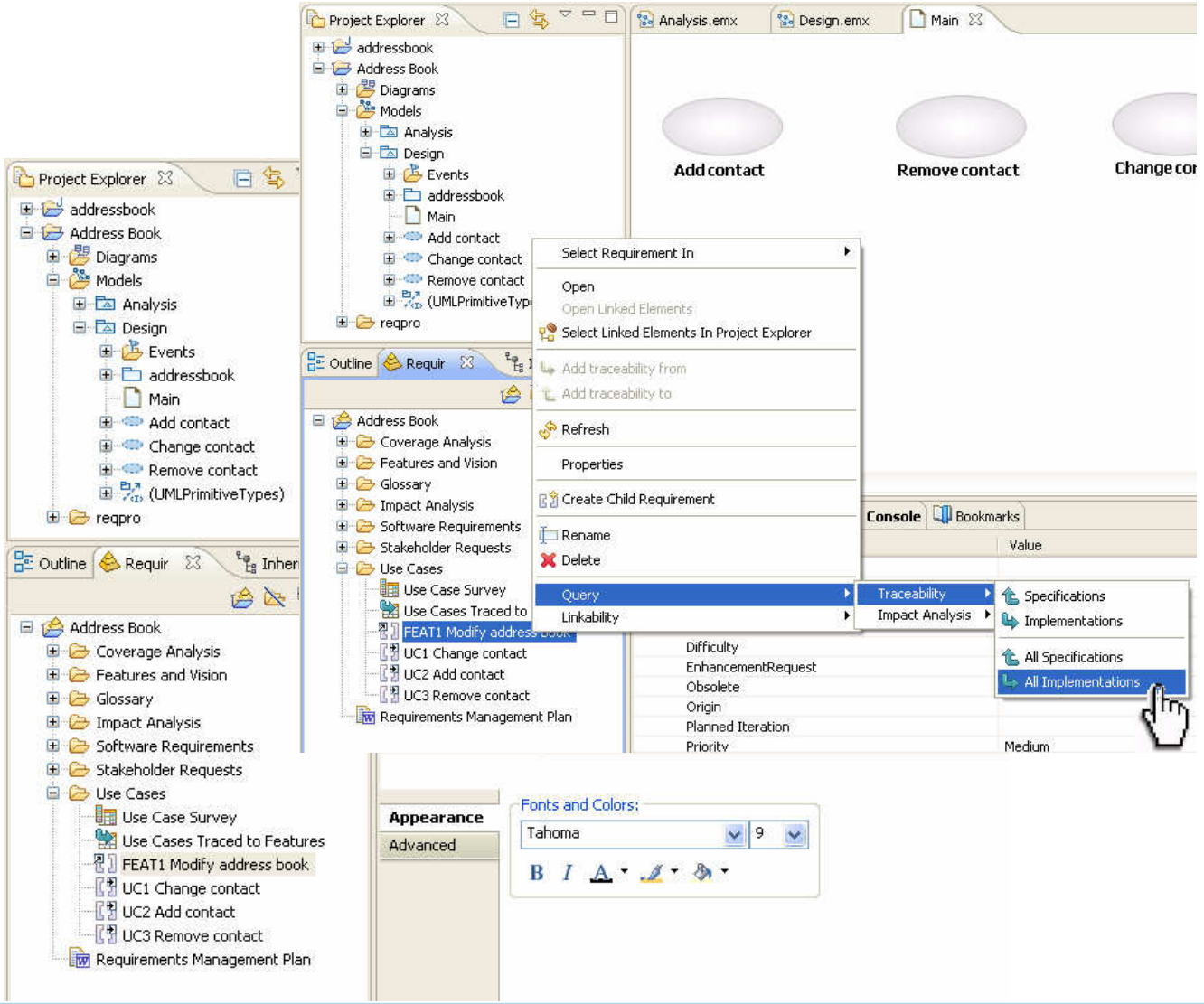
Alias	
Classifier Behavior	

Web Publish: URL Links

- Publishing URL links to external documentation improved
  - Absolute file URLs, when browsing web publish output locally
  - Relative file URLs, when committing web publish output to source control
  - Relative file URLs for files copied to the web publish output folder when moving it to a server

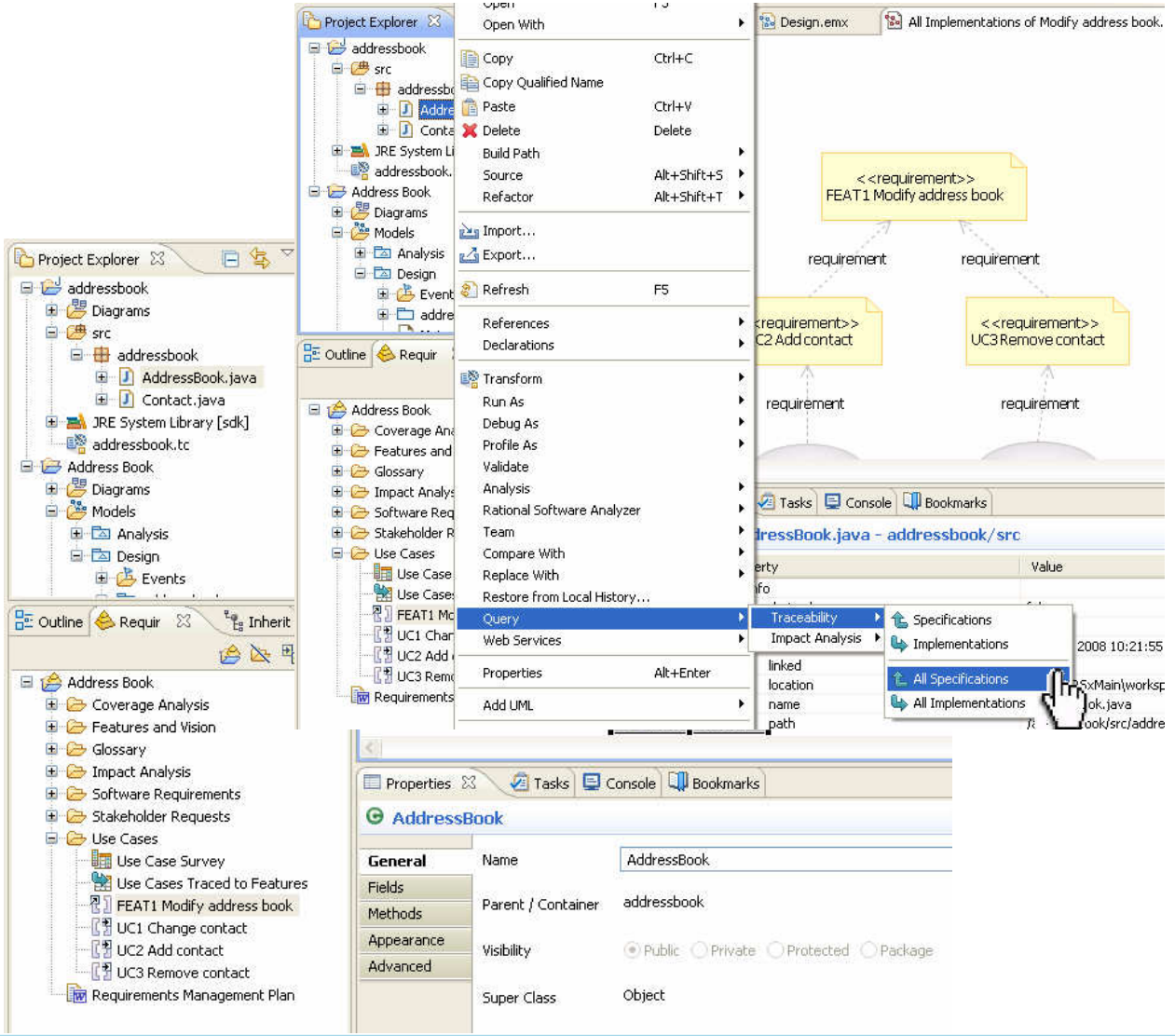
Traceability: Requirement Queries

- Traceability and impact analysis queries are available in the requirement explorer
- Explore queries work with requirement views
- The queries return requirement trace to and trace from relationships



Traceability: Java Type Queries

- Traceability and Impact Analysis queries are available in the Package Explorer



WBM Integration Enhancements

- Support for WBM 6.1.X
- Support for SOMA ME
  - Support global and local “human task”
  - Support global and local “business rule task”
  - Generate comments for the “process” and “task” description

AM03

What‘s New in RSA and RSM 7.5

Team Development and Compare and Merge

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

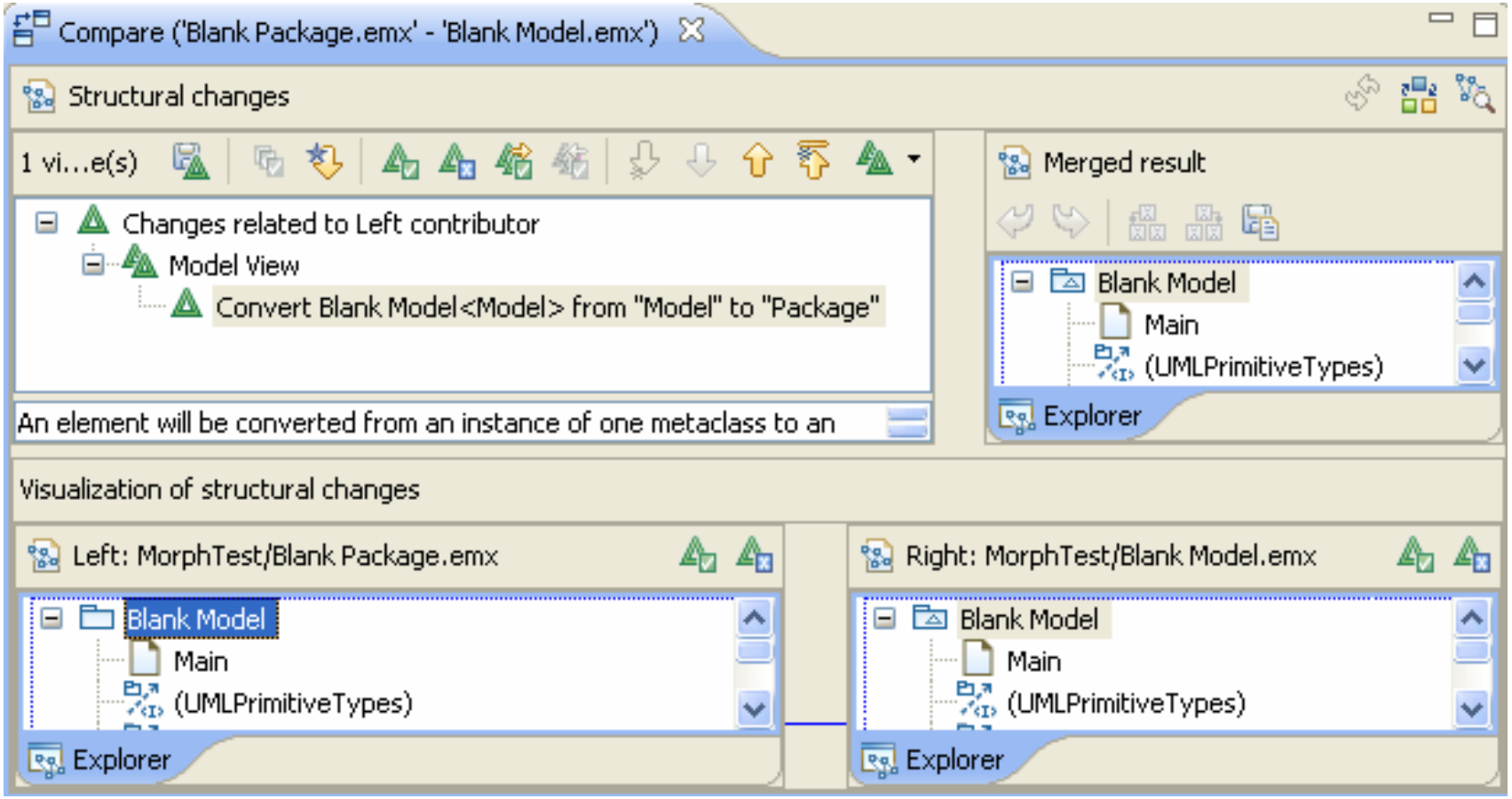
IBM

Rational. software

Rational Modeling Compare Support

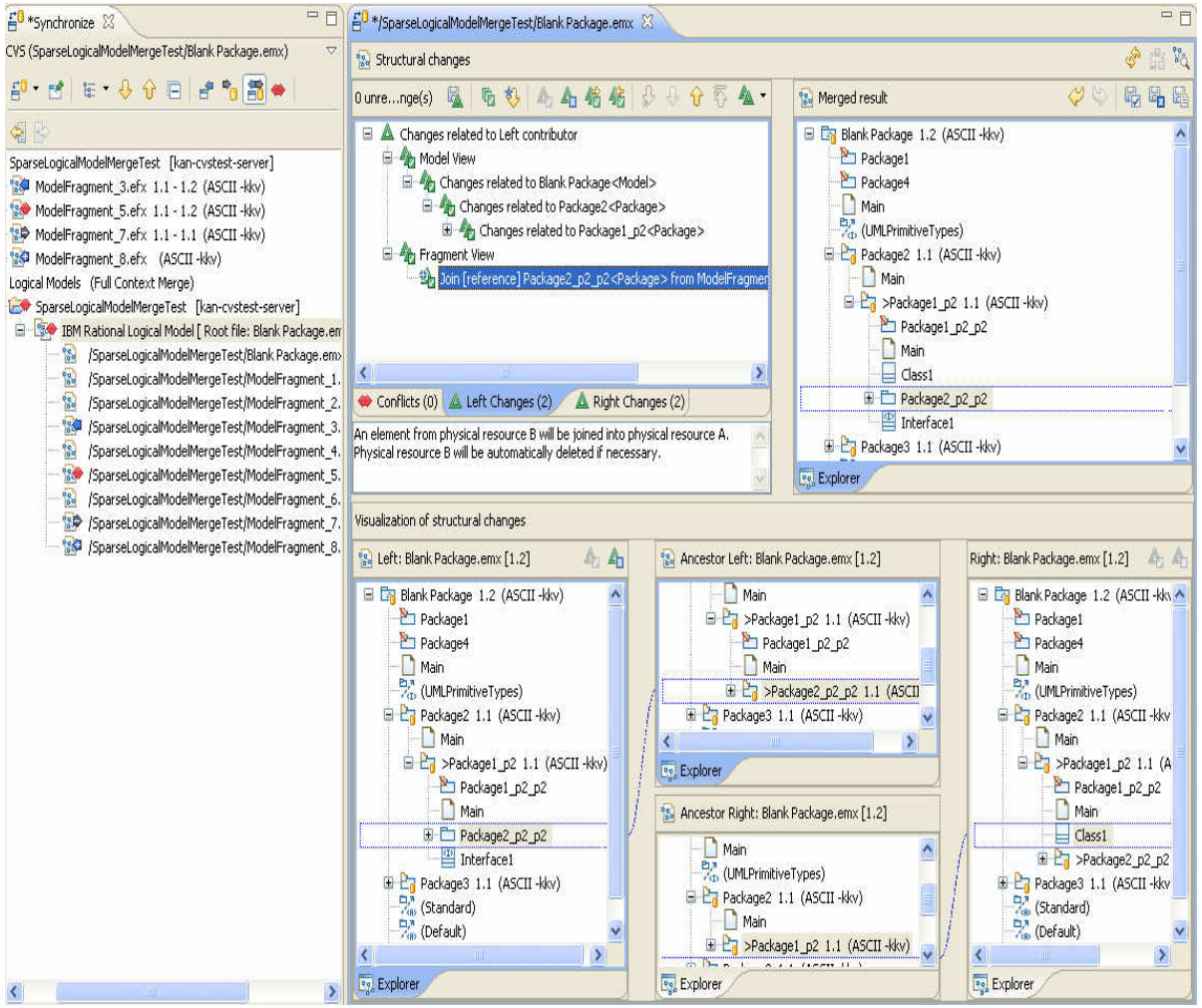
- Compare and Merge improvements
  - New **morph delta** to denote Model to Package, Control Flow to Object Flow and vice versa
  - **Sparse Logical Model Merge** to handle massive fragmented models
  - Text View Mode for multi line text field to improve usability
  - Delta Tree **filters, sorters and groupers** to reduce interface clutter
  - Editable custom delta tree filter to tailor to your environment
  - Reorganize contributors before first compare session for efficiency
- Fusion (Structural Merge) improvements
  - **Merge as text** for text field deltas for Fusion, Compare and Merge
  - Remember pane visibility settings for Fuse

Morph: Compare Model and Package

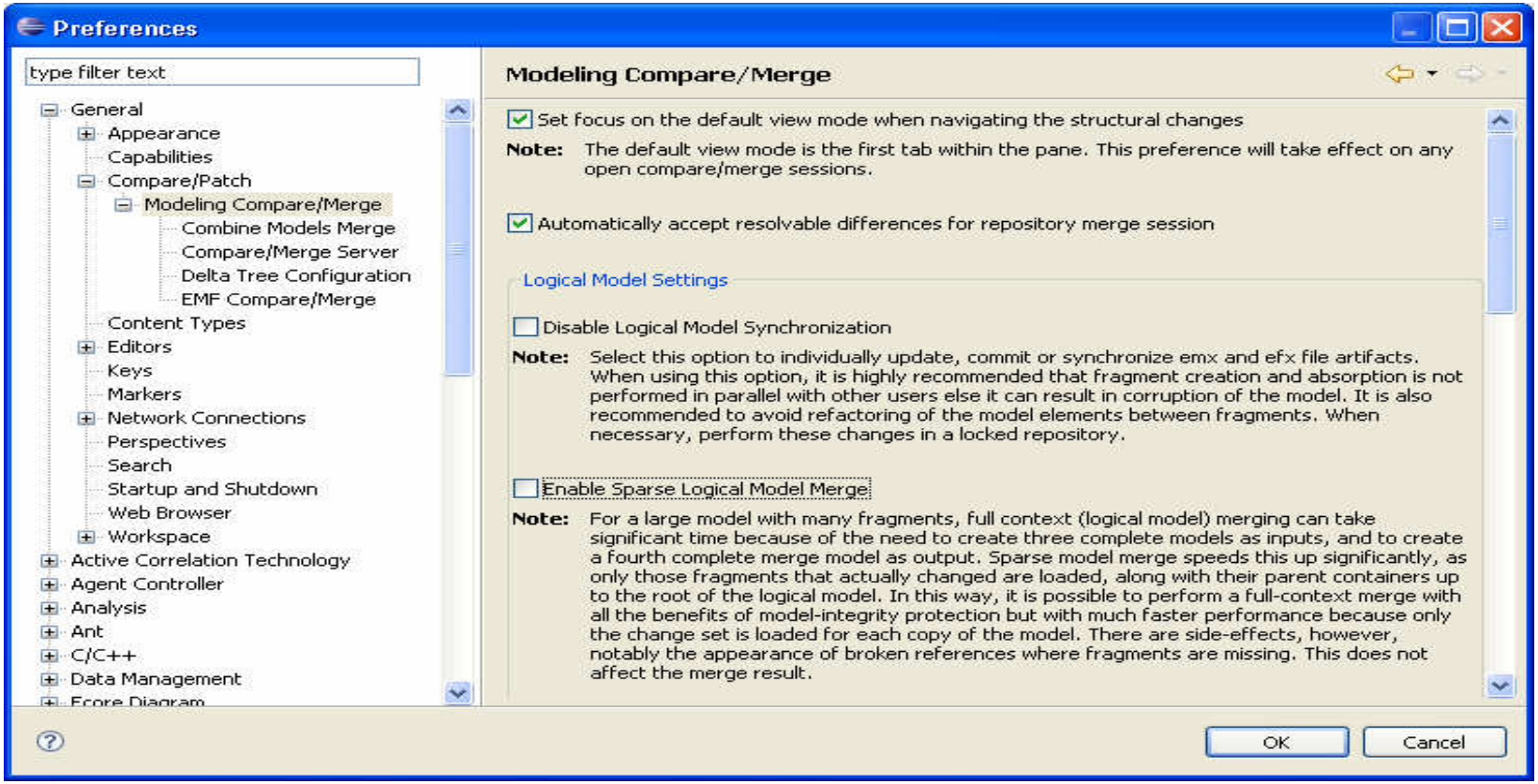


Sparse Logical Model Merge

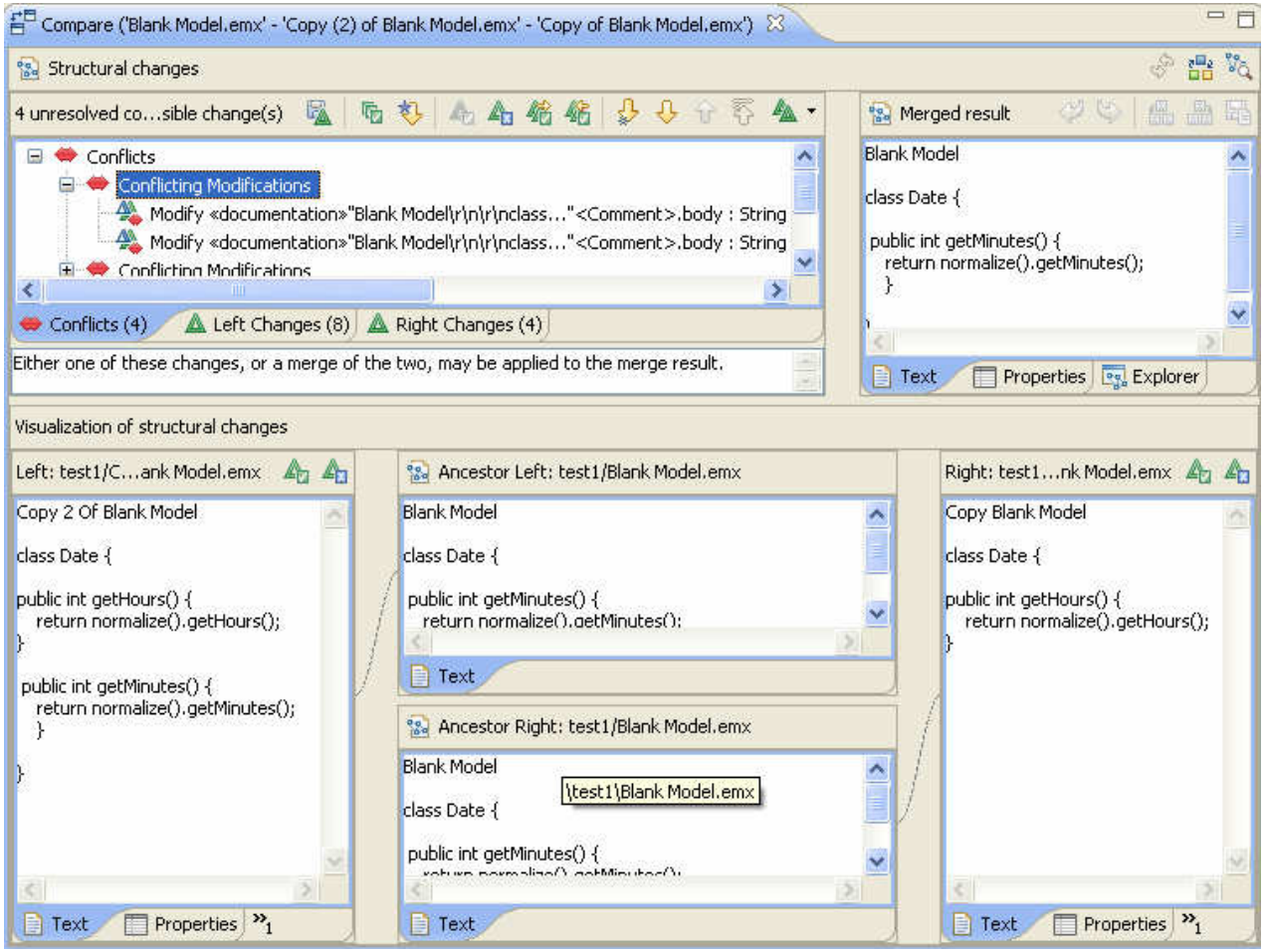
- Load the modified fragments and their ancestors
- Scales based on change set size, **not** model size!



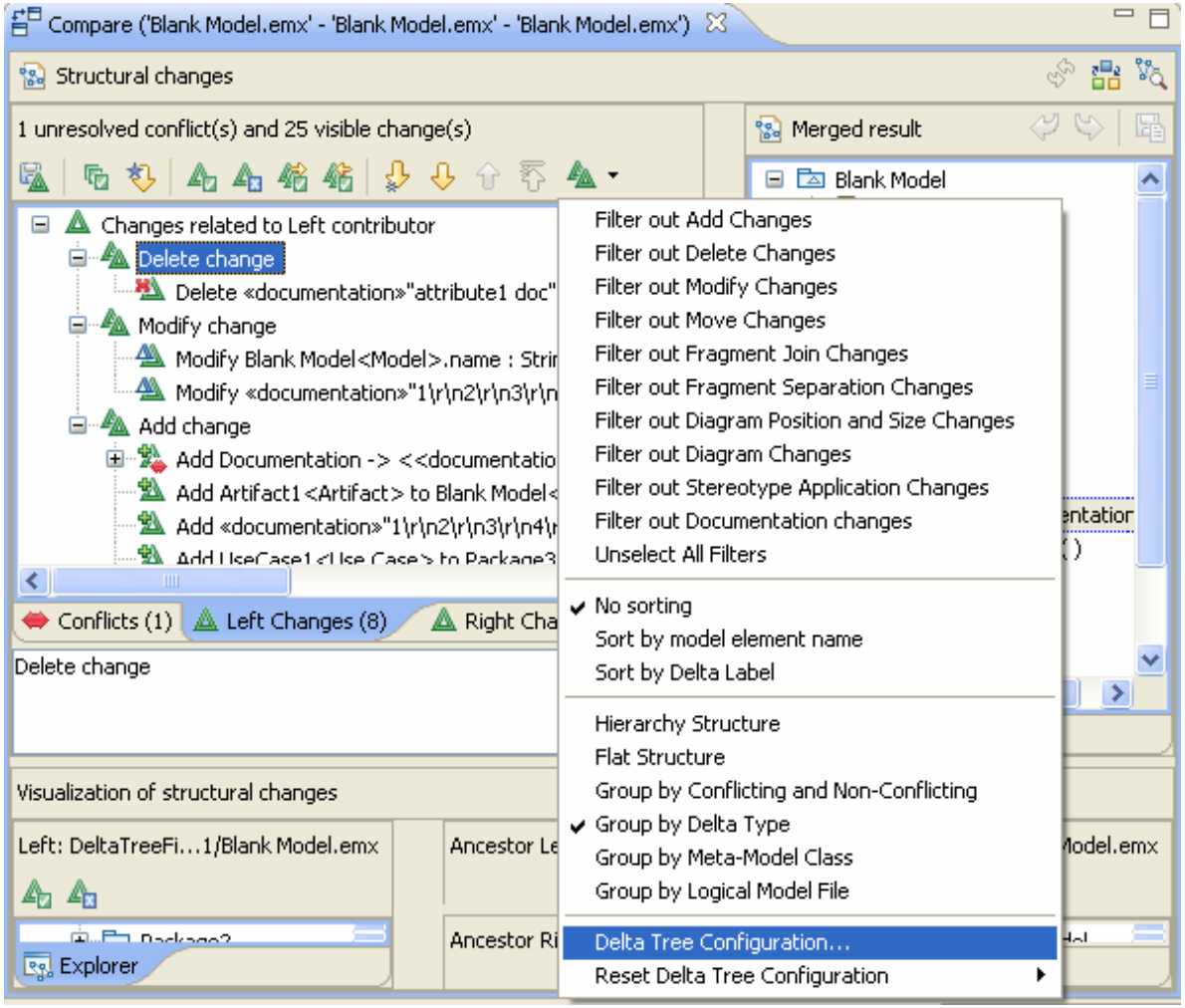
Enable Sparse Logical Model Merge option



Text View Mode for Multi-line Text Field

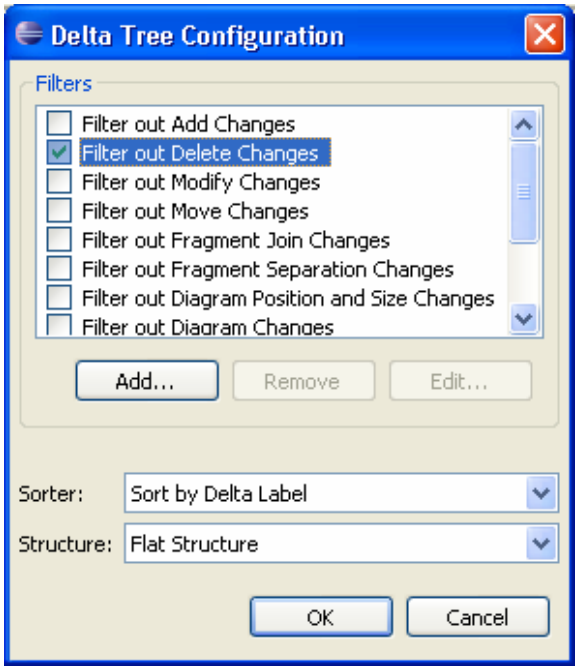


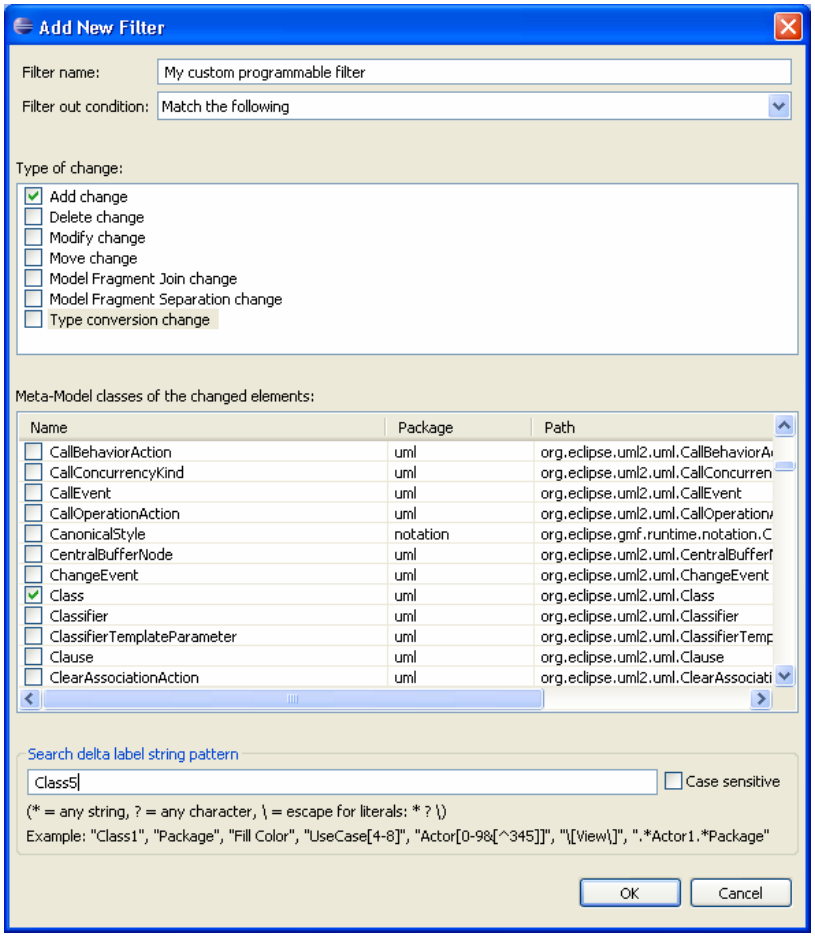
Delta Tree Filters, Sorters and Groupers



Editable Custom Delta Tree Filter

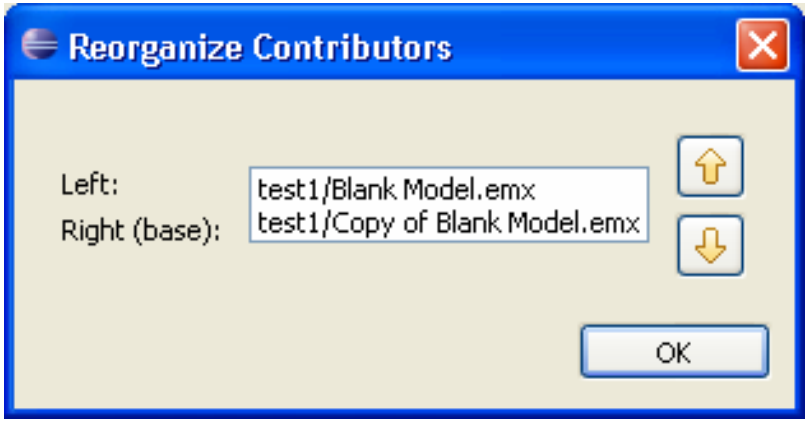
- Can be saved and loaded, thus sharable in team repository

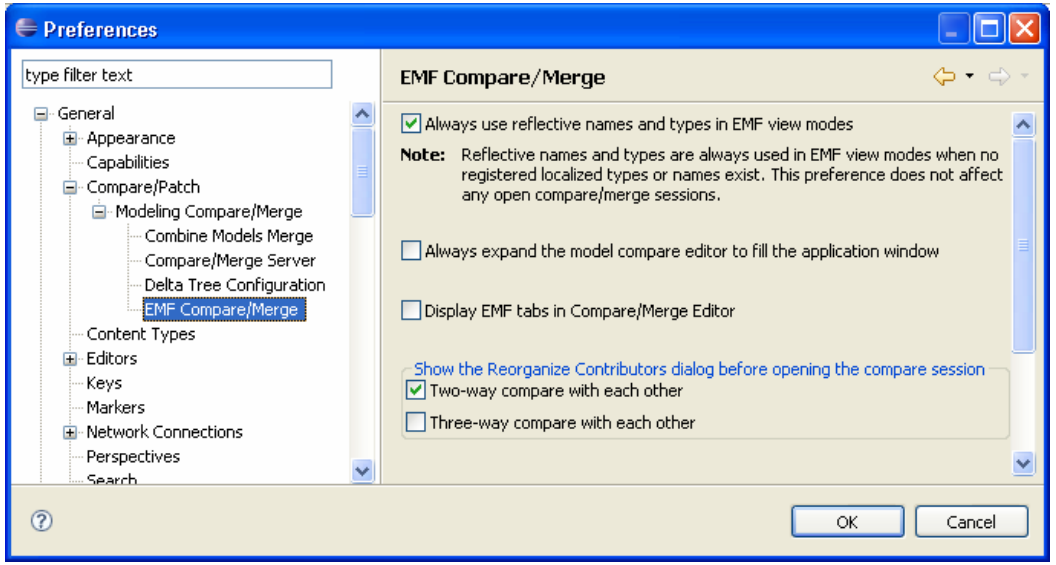




Reorganize Contributors Before Session

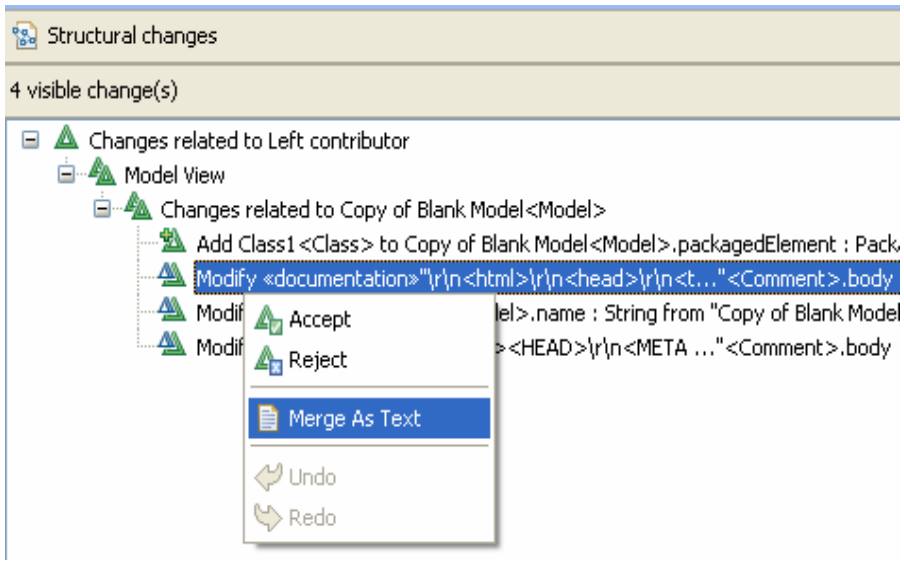
- Offers the opportunity to organize the contributors for Eclipse “compare with” commands before starting session
- Cuts time to get files in the right role in half, makes a big difference for massive models

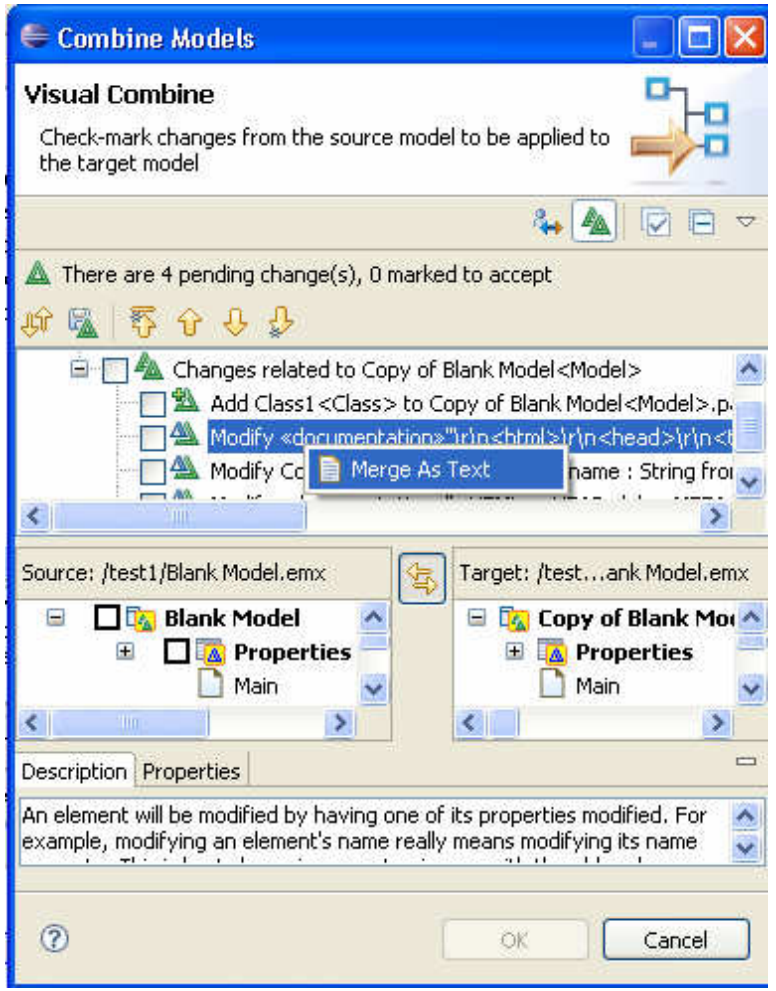




Merge As Text for Compare Merge and Fuse

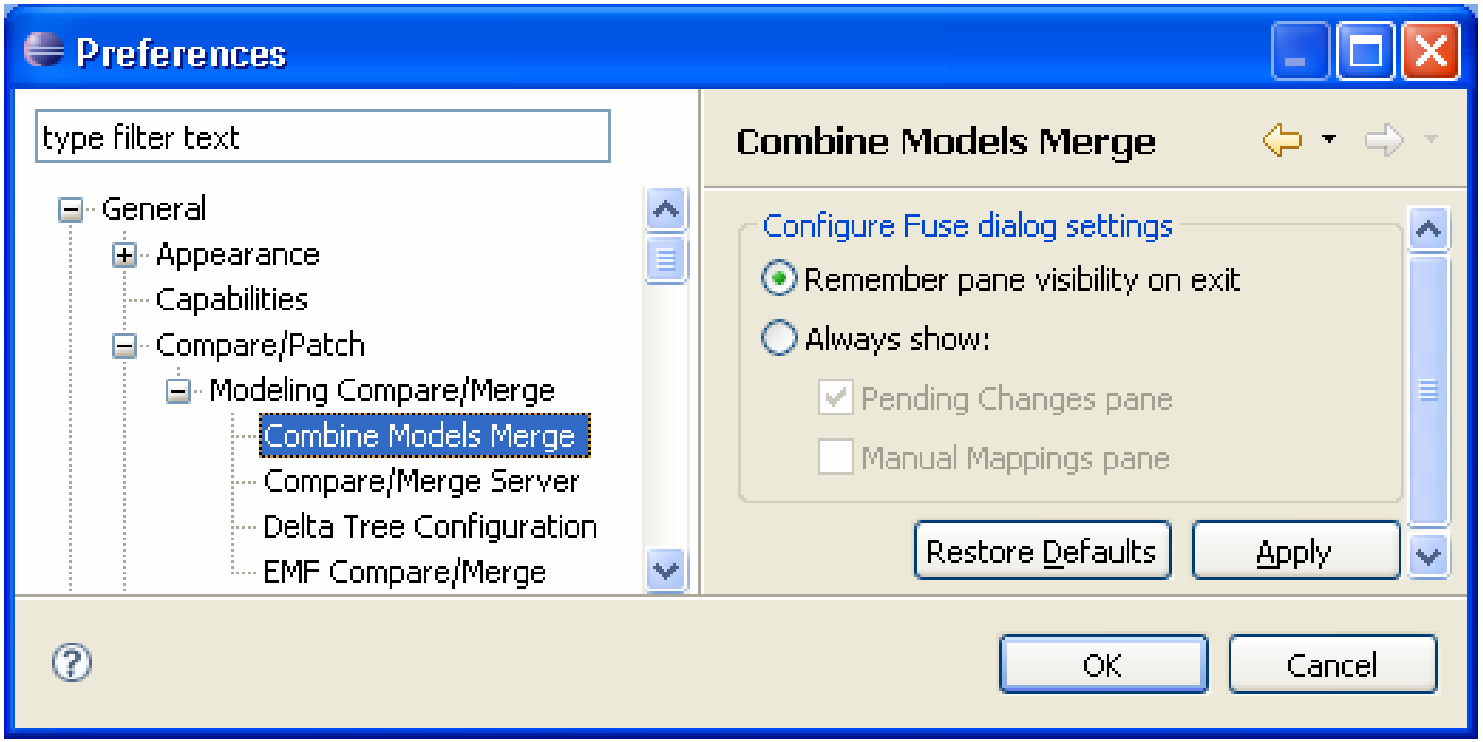
- Allows any text delta or conflict in either team merge or fusion (combine models command or transform) to be compared or merged recursively





Pane visibility settings for Fuse

- Remembers whether pending changes pane and/or manual match pane are visible
- Controlled by preferences



AM03

What‘s New in RSA and RSM 7.5

Model-Driven Development (MDD) Tooling

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

IBM

Rational. software

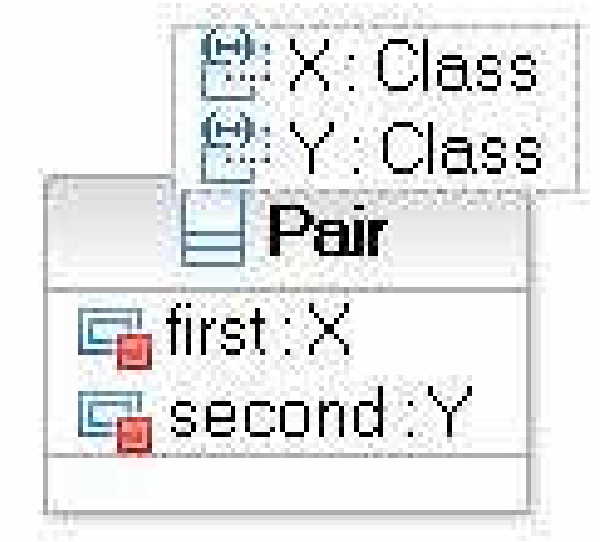
Java Modeling

- Sequence diagrams
  - “Drill-down” selected method calls on a static method sequence diagram
  - Create static method sequence diagrams for multiple java methods using a single action
    - API to do the same programmatically
  - Copy-paste between static method sequence diagrams and UML diagrams
  - Method signature refactoring
- Visualized java element navigation to project explorer
- Extend JPA/EJB 3.0 visualizers with existing java modeling functionality
- Harvest documentation on visualized elements
- Simultaneous package and type search

Java Transformations

- Java generics
- Preservation of fuse filter settings
  - Delta filter settings preserved per transformation configuration
  - Can reset to default settings via menu item
- Finer grained source for Java-to-UML
  - Java project
  - Java source folder
  - Java package
  - Java class/interface/enumeration
- UML-to-Java supports types defined in XSD data type library

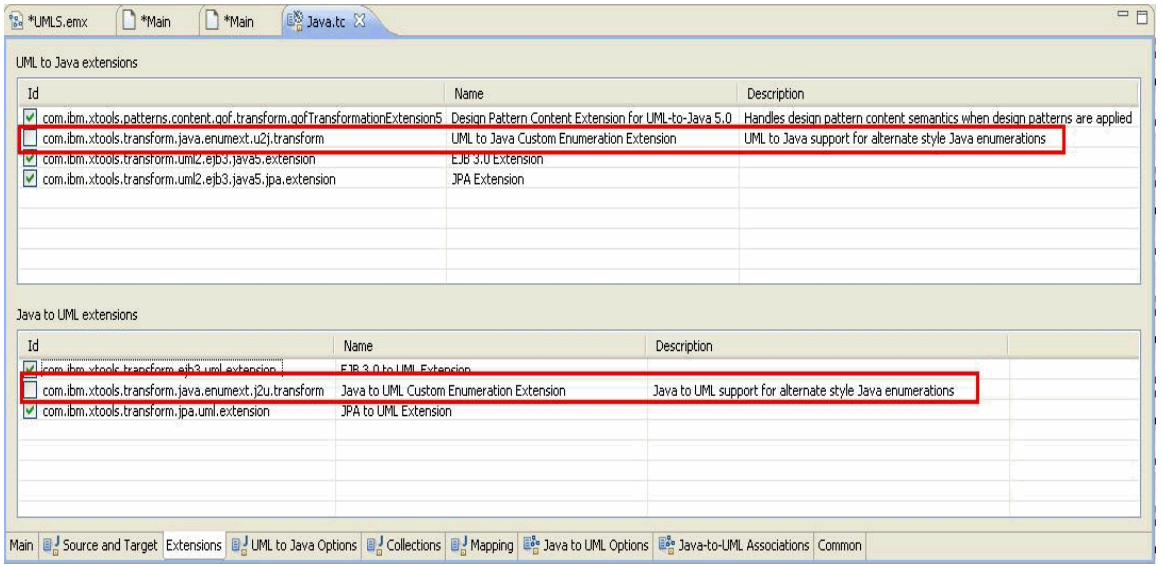
Java Generics Transformation




```
class Pair<X, Y> {
    private X first;
    private Y second;
}
```

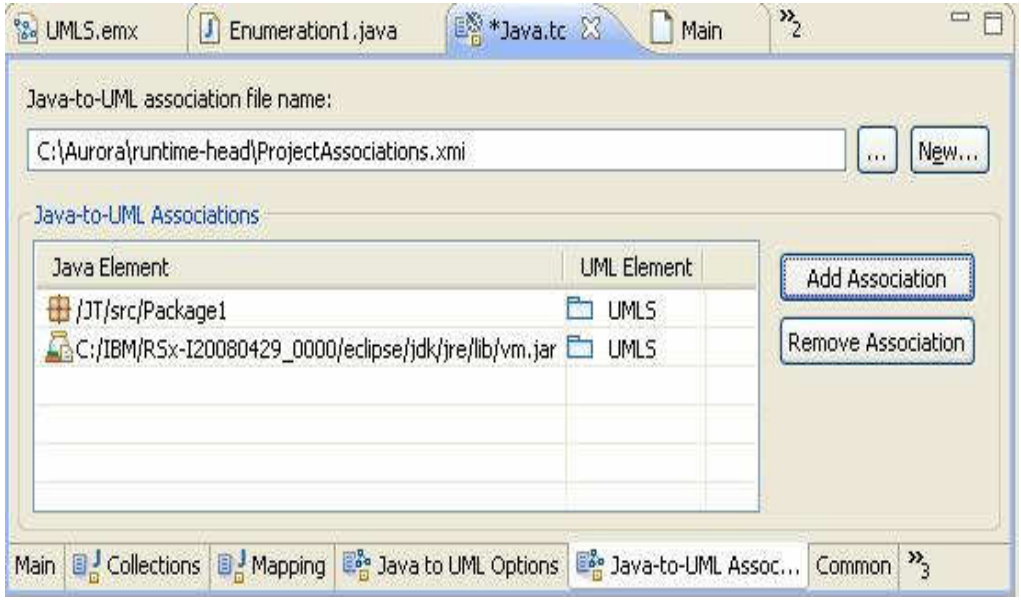
Custom Enumeration Generation

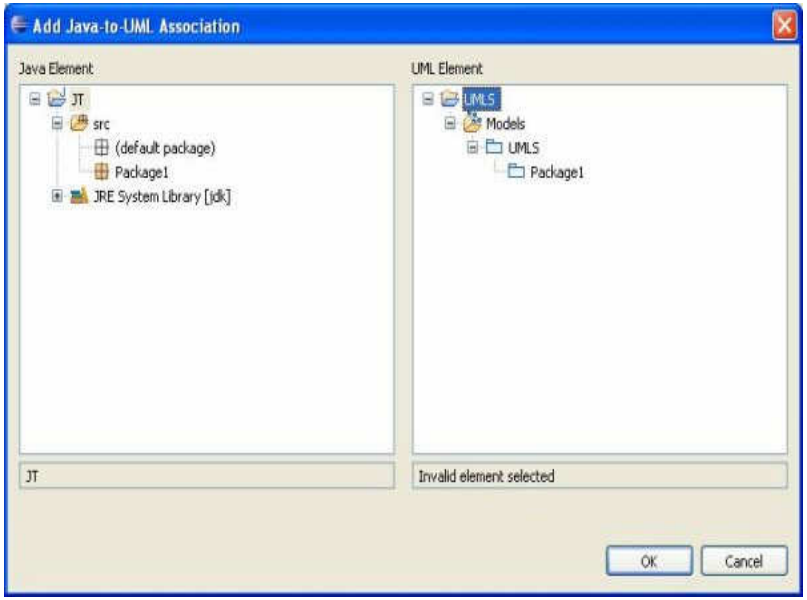
- Optional extension preserves the UML enumeration literal string name and value in a generated java 5.0 style enumeration
- Java 5.0 enumerations do not preserve the UML literal string
- Both are available as options in the transformation



Java To UML Associations

- Link java elements to UML elements
- Enables proper generation of relationships between UML models during Java-to-UML



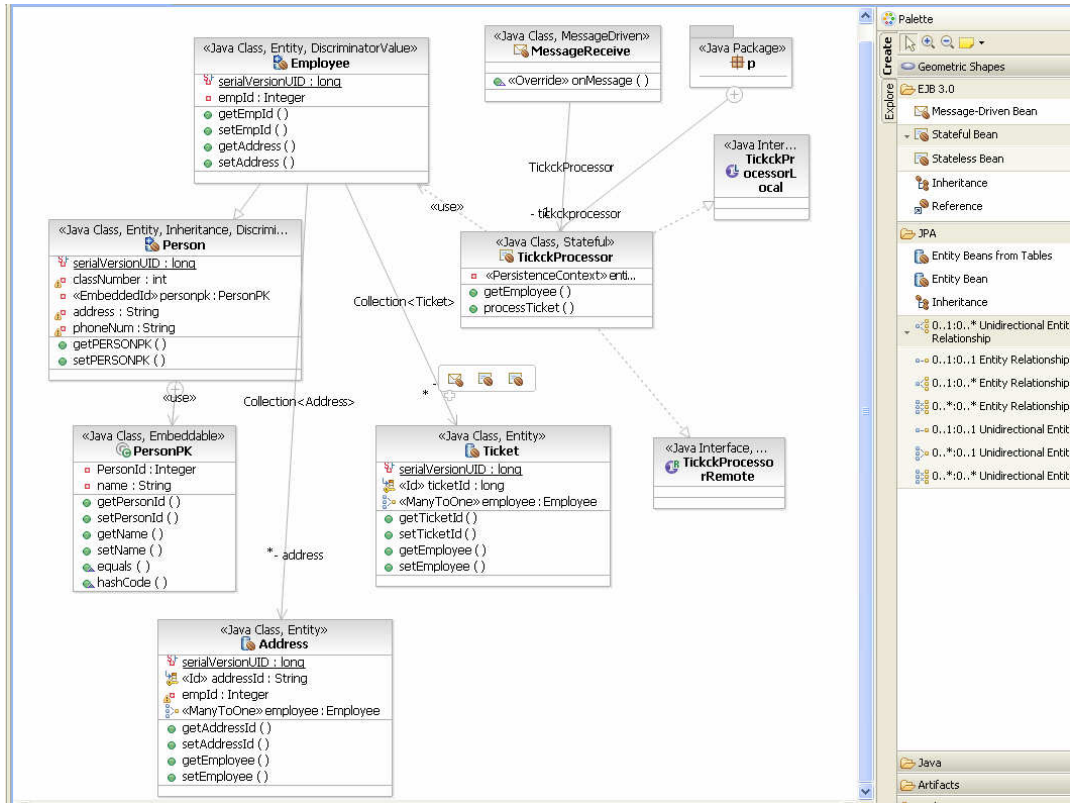


JPA, EJB 3.0 Modeling support

- Java visual editing
  - Create JPA Entity, JPA Entity from table, Session Bean, Message Driven Bean
    - relationships between JPA Entity beans
    - inheritance relationships between JPA Entity beans
  - Create EJB reference relationships
  - Common EJB Annotation Settings for Security, Interceptors, Transaction management type/attribute
  - Diagrammatic rendering of annotations
- Integration with JPA, EJB 3.0 tooling
  - Show in Deployment Descriptor, Persistence editor
  - Creation of Named Query, CRUD methods
  - Annotations view, JPA Details view

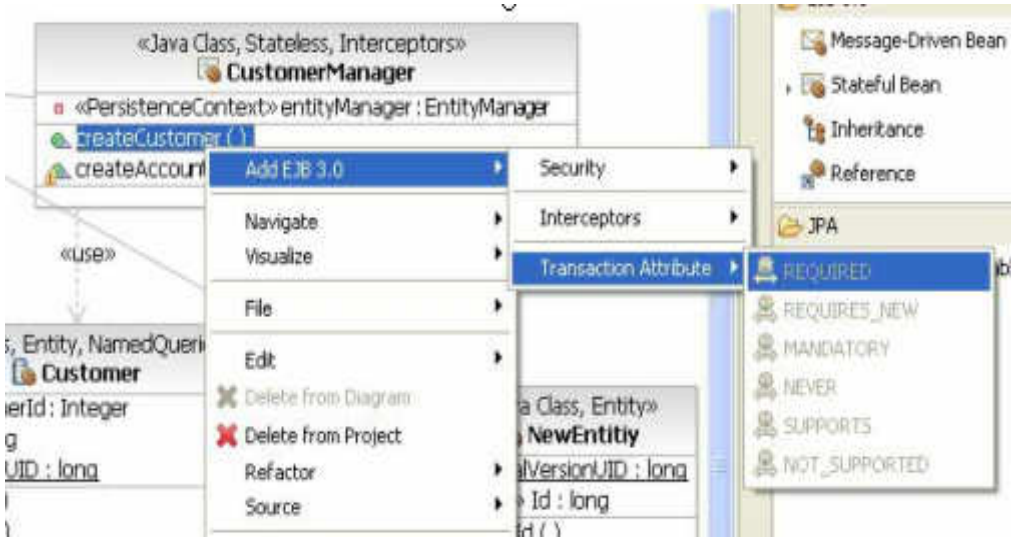
JPA, EJB 3.0 Visual Editing

- Layout annotated JPA and EJB 3.0 classes
- Palette for JPA and EJB 3.0 artifacts
- Create and edit relationships
- Set properties in diagram



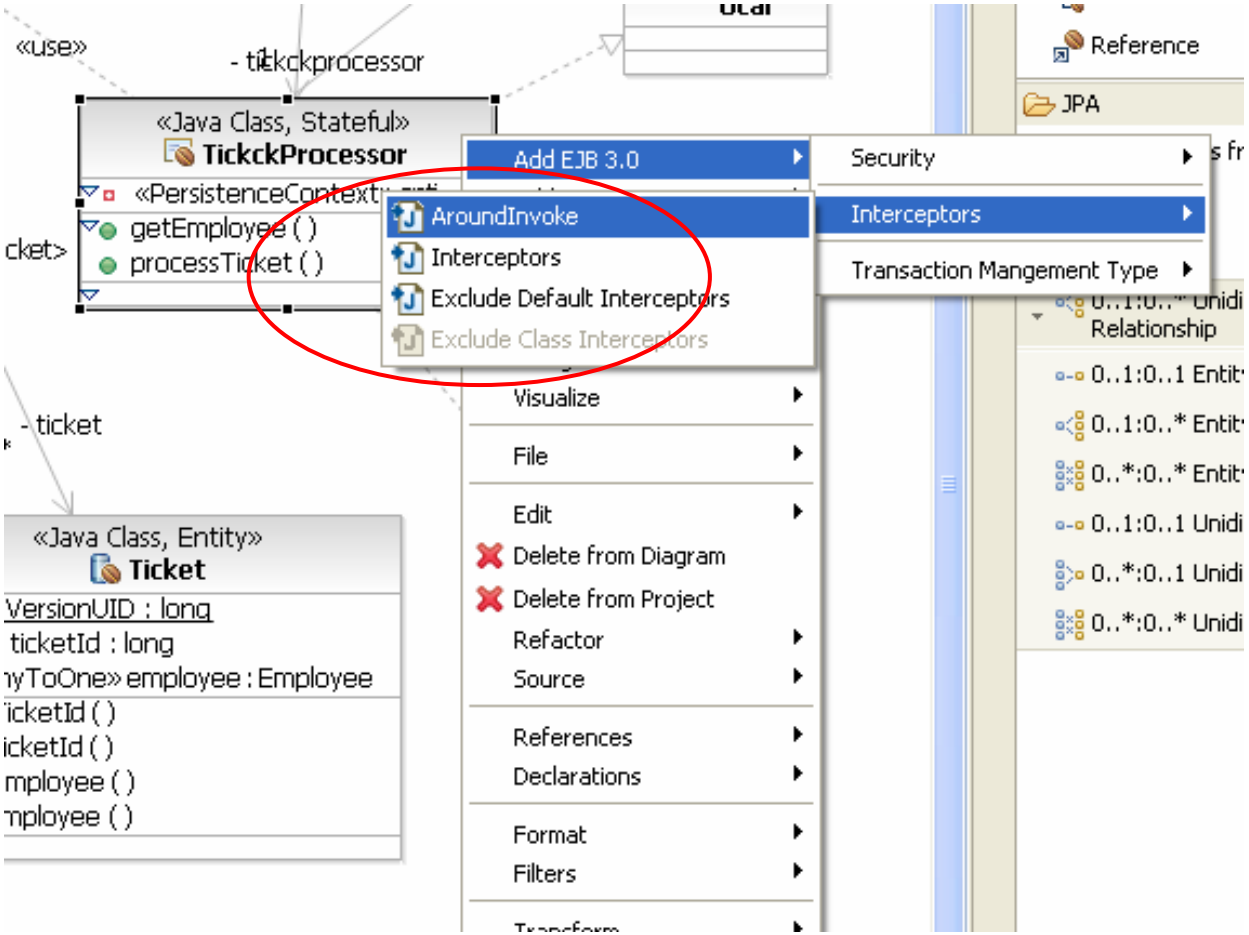
EJB 3.0 Security and Transaction Support

- Easily add security and transaction settings for EJB 3.0 beans
- Hides the annotation notation
- Support for class, field, and method level settings



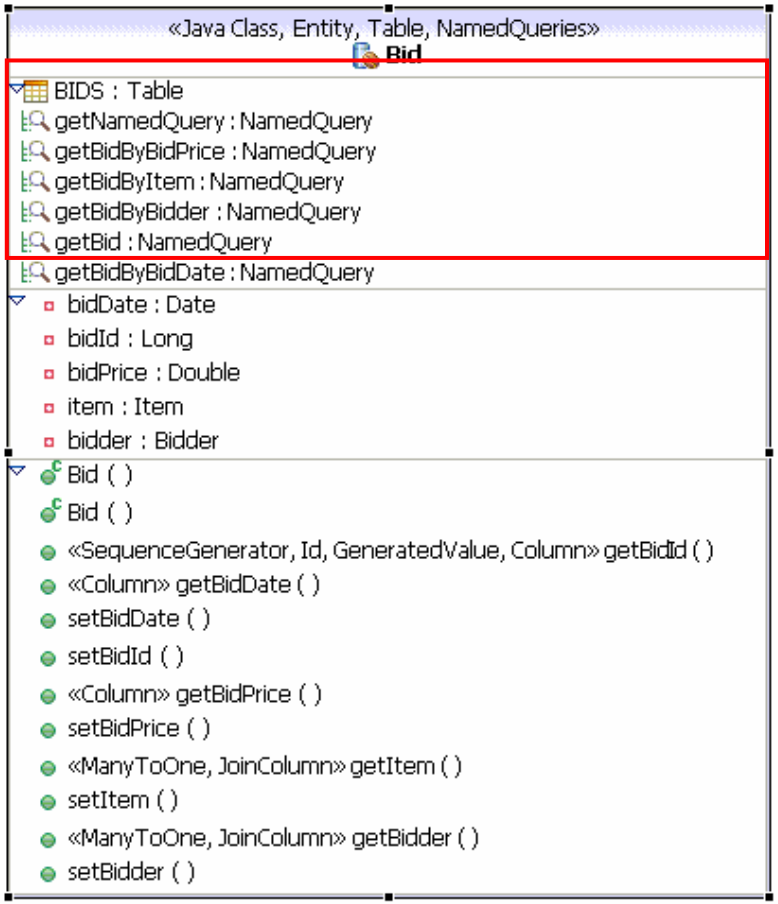
EJB 3.0 Interceptors

- Selectively add life-cycle inceptors to your EJB 3.0 beans
- Annotations are automatically added in the correct locations



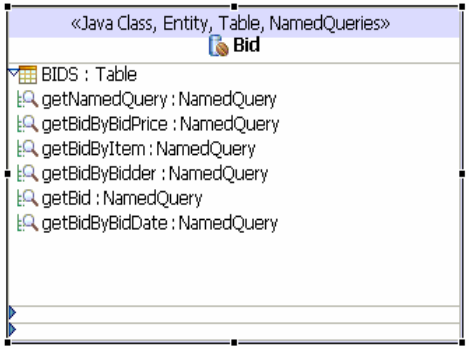
New Annotation Compartment

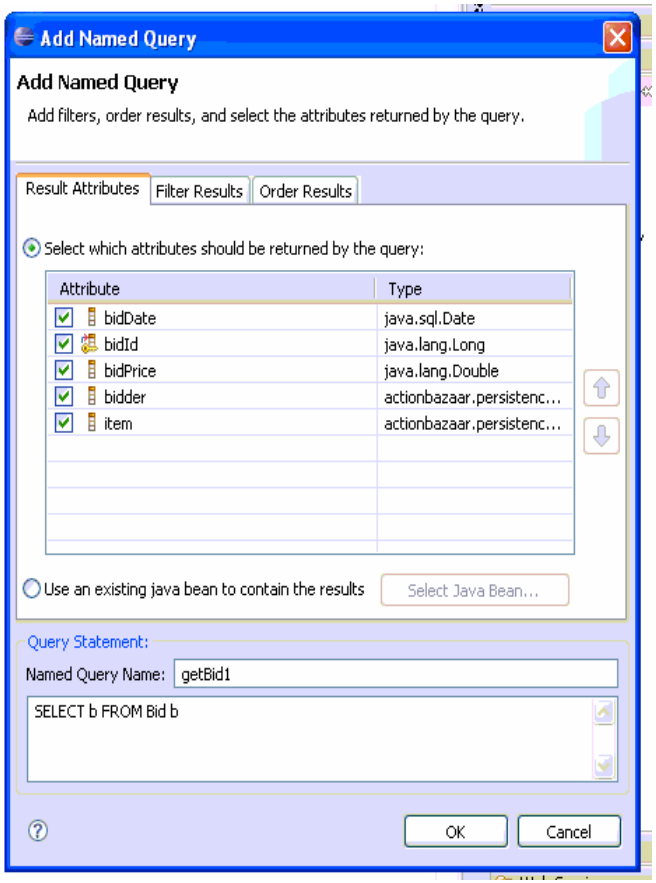
- Provides a location to visualize class-level annotation settings
- Annotations are selectable and will drive properties view and other related views



Named Query

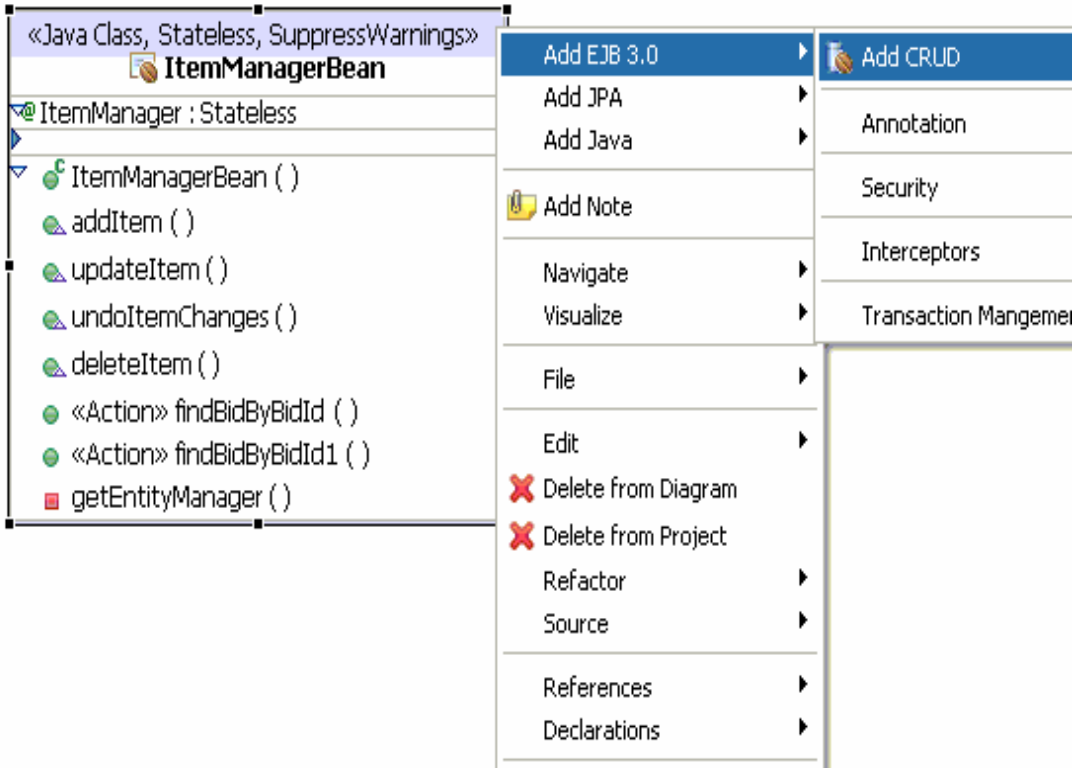
- Easily create Named Queries on JPA Entity beans
- Create and edit queries using a query builder
- Annotations are entered and rendered in the annotation compartment





Basic JPA Entity Management with Session Beans

- Add **C**reate, **R**ead, **U**pdate, **D**elete methods to a session bean to easily manipulate a selected JPA Entity bean
- Enters the necessary EntityManager annotations and invocations to read and persist a selected JPA Entity
- Quickly provides the method generation pattern necessary for interacting with an EntityManager



JPA, EJB 3.0 Transformations

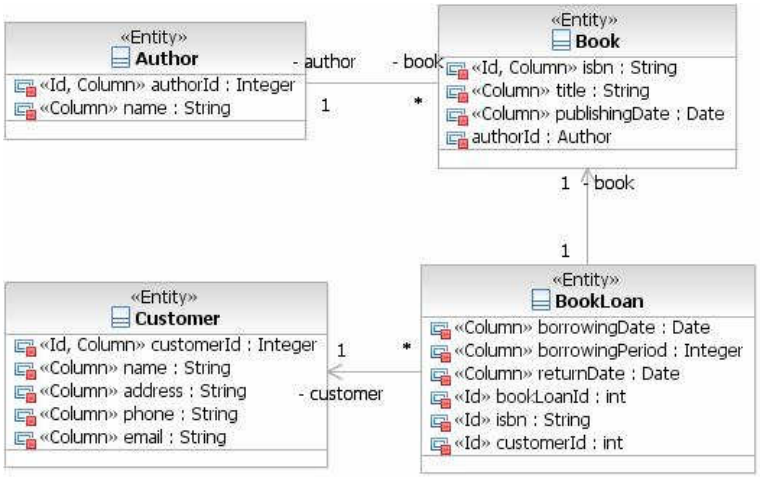
- Four new transformations
  - UML-to-JPA
  - UML-to-EJB 3.0
  - JPA-to-UML
  - EJB 3.0-to-UML
- Implemented as extensions of UML-to-Java and Java-to-UML
  - Contain full UML-to-Java, Java-to-UML feature set
- Profile tooling
  - JPA transformation profile
  - EJB 3.0 transformation profile

JPA Transformations

- UML-to-JPA
  - Generates JPA annotations for Entity classes
  - Supports reapply via JMerge
    - New JMerge rules to handle merging of annotations
  - Supports replacement of UML elements
  - Supports generation of traceability links
- JPA-to-UML supports fusion into target model
- Both transformations extendable
- New JPA transformation profile and supporting profile tools

UML-to-JPA

- JPA profile created for modeling JPA concepts
- Transformation for generating JPA Entities from UML classes
- Palette tools created for working with the JPA profile improves the usability of creating JPA models



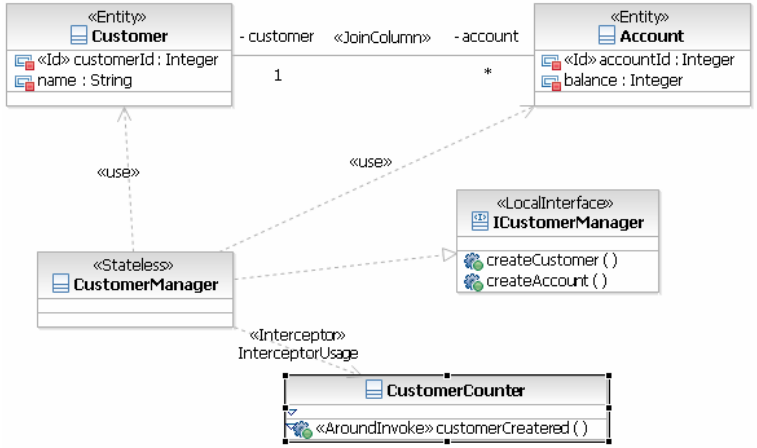

```
 * @author mbadr
 * @generated "UML to EJB 3.0 (com.ibm.xtools.transform.uml2.ejb3.java.internal.UML2EJB3Transform)"
 */
@Entity
@NamedQueries(
        {@NamedQuery(name = "Author.findByAuthorId", query = "select obj from Author where obj.authorId = :authorId")
        ,@NamedQuery(name = "Author.findByName", query = "select obj from Author where obj.name = :name")})
public class Author implements Serializable {
    /**
     * <!-- begin-UML-doc -->
     * <!-- end-UML-doc -->
     * @generated "UML to EJB 3.0 (com.ibm.xtools.transform.uml2.ejb3.java.internal.UML2EJB3Transform)"
     */
    @Id
    @Column(name="AuthodID")
    Integer authorId;
```

EJB 3.0 Transformations

- UML-to-EJB 3.0
  - Generates EJB 3.0 artifacts
    - Session Beans, JMS and non-JMS Message-Driven Beans
    - External elements like Datasource, Topics, Queues
    - Transforms sequence diagrams to initial implementation of operations
  - Supports reapply via JMerge
    - New JMerge rules to handle merging of annotations
  - Supports replacement of UML elements
  - Supports generation of traceability links
- EJB 3.0-to-UML supports fusion into target model
- Both transformations extendable
- New EJB 3.0 transformation profile and supporting profile tools

UML-to-EJB 3.0

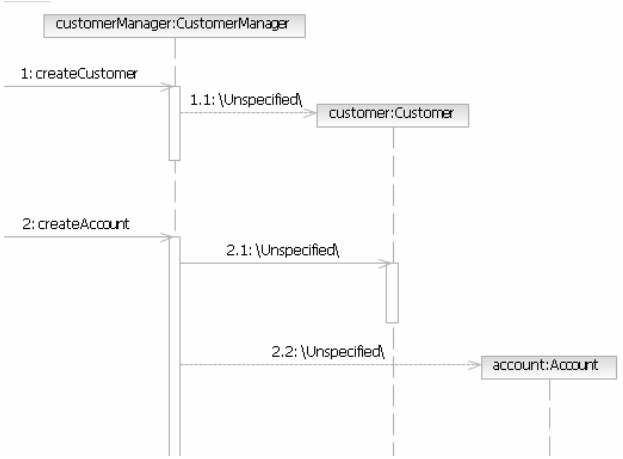
- EJB 3.0 profile created for modeling EJB 3.0 concepts
- Transformation for generating EJB 3.0 beans from UML classes
- Palette tools created for working with the EJB 3.0 profile improves the usability of creating EJB 3.0 models




```
 * @author mbadr
 * @generated "UML to EJB 3.0 (com.ibm.xtools.transform.uml2.ejb3.java.internal.UML2EJB3Transform)"
 */
@Stateless
@Interceptors(piggybank.CustomerCounter.class)
public class CustomerManager implements ICustomerManager {
    /**
     * <!-- begin-UML-doc -->
     * <!-- end-UML-doc -->
     * @generated "UML to EJB 3.0 (com.ibm.xtools.transform.uml2.ejb3.java.internal.UML2EJB3Transform)"
     */
    @PersistenceContext
    private EntityManager entityManager;
```

UML-to-EJB 3.0 Interaction Support

- UML to EJB 3.0 transformation supports interactions described in a sequence diagram
- Generates JPA entity interactions from a Session bean
- Inserts the necessary EntityManager invocations




```
public Boolean createCustomer(Integer customerID, String name
    {
        Customer entity = new Customer();
        //TODO set the entity properties
        entityManager.persist(entity);
    }
    {
        // begin-user-code
        // TODO Auto-generated method stub
        return null;
        // end-user-code
    }
}
public Boolean createAccount(Integer customerID, Integer acc
    {
        Integer id = null; //TODO initialize and set the valu
        Customer entity = entityManager.find(Customer.class,
    }
    {
        Account entity = new Account();
        //TODO set the entity properties
        entityManager.persist(entity);
    }
    {
        // begin-user-code
        // TODO Auto-generated method stub
        return null;
        // end-user-code
    }
```

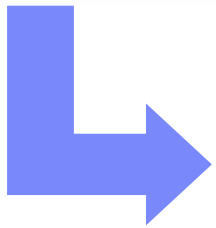
SOA Transformations

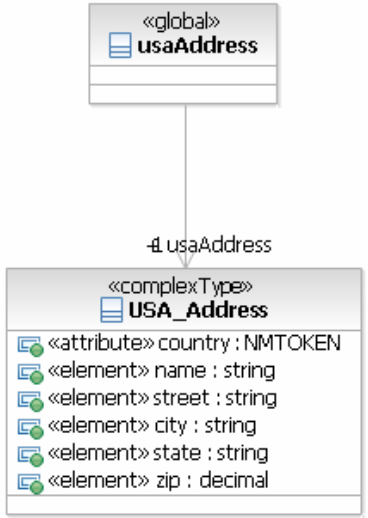
- XSD visualization of individual XSD files
- XSD-to-UML
- Service modeling transformations
  - Java-to-Service Model
  - SessionBean-to-Service Model
- Software Services Profile tooling
- Merge enhancements
- UML-to-XSD, UML-to-WSDL configuration enhancements
- UML-to-BPEL support for Human Task generation
- SOA transformation extension sample

XSD-to-UML Transformation

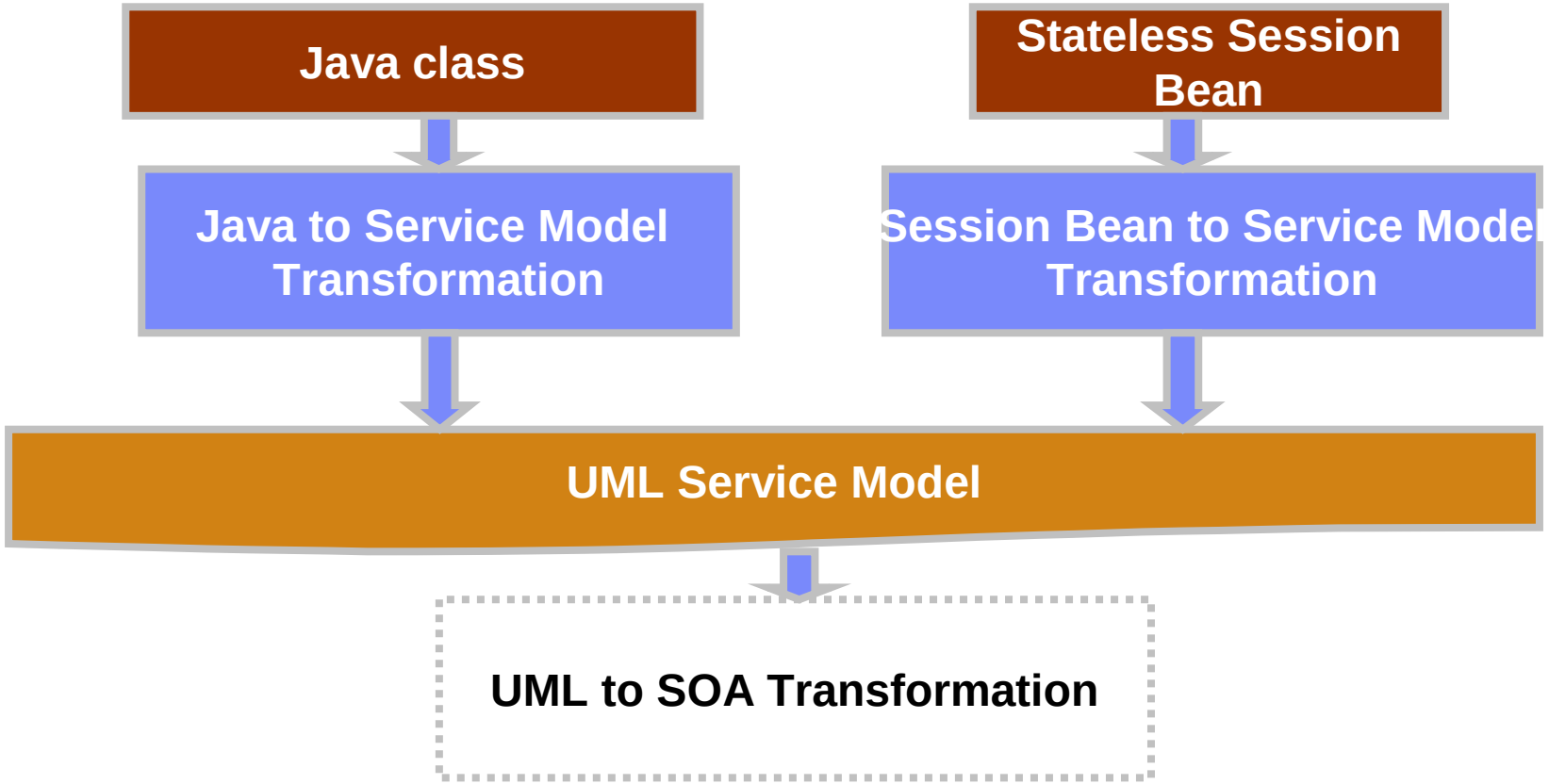
- Authored with RSA Model-to-Model transformation authoring tools
- Supports >98% of XSD element combinations
- Output consumable by the UML-to-XSD transformation
- Fusion of generated UML models with existing ones (Automatic, Silent, Overwrite and Visual)
- Map multiple source files to multiple target models

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:test="http://www.bryan.com/test"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    targetNamespace="http://www.bryan.com/test">
    <xsd:complexType name="USA_Address">
        <xsd:sequence>
            <xsd:element name="name" type="xsd:string" />
            <xsd:element name="street" type="xsd:string" />
            <xsd:element name="city" type="xsd:string" />
            <xsd:element name="state" type="xsd:string" />
            <xsd:element name="zip" type="xsd:decimal" />
        </xsd:sequence>
        <xsd:attribute name="country" fixed="US" type="xsd:NMTOKEN" />
    </xsd:complexType>
    <xsd:element name="usaAddress" type="test:USA_Address" />
</xsd:schema>
```





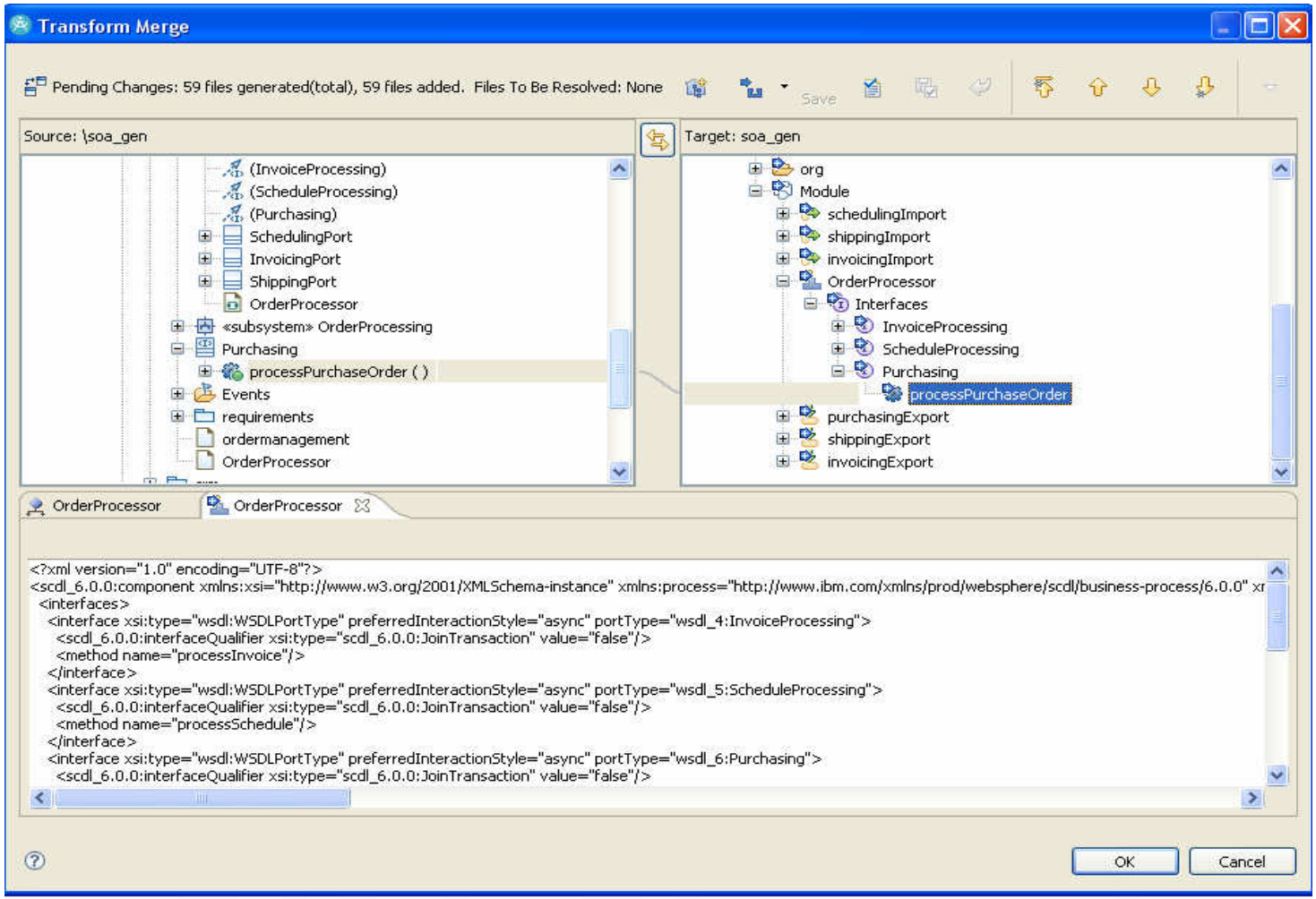
Service Model Transformations



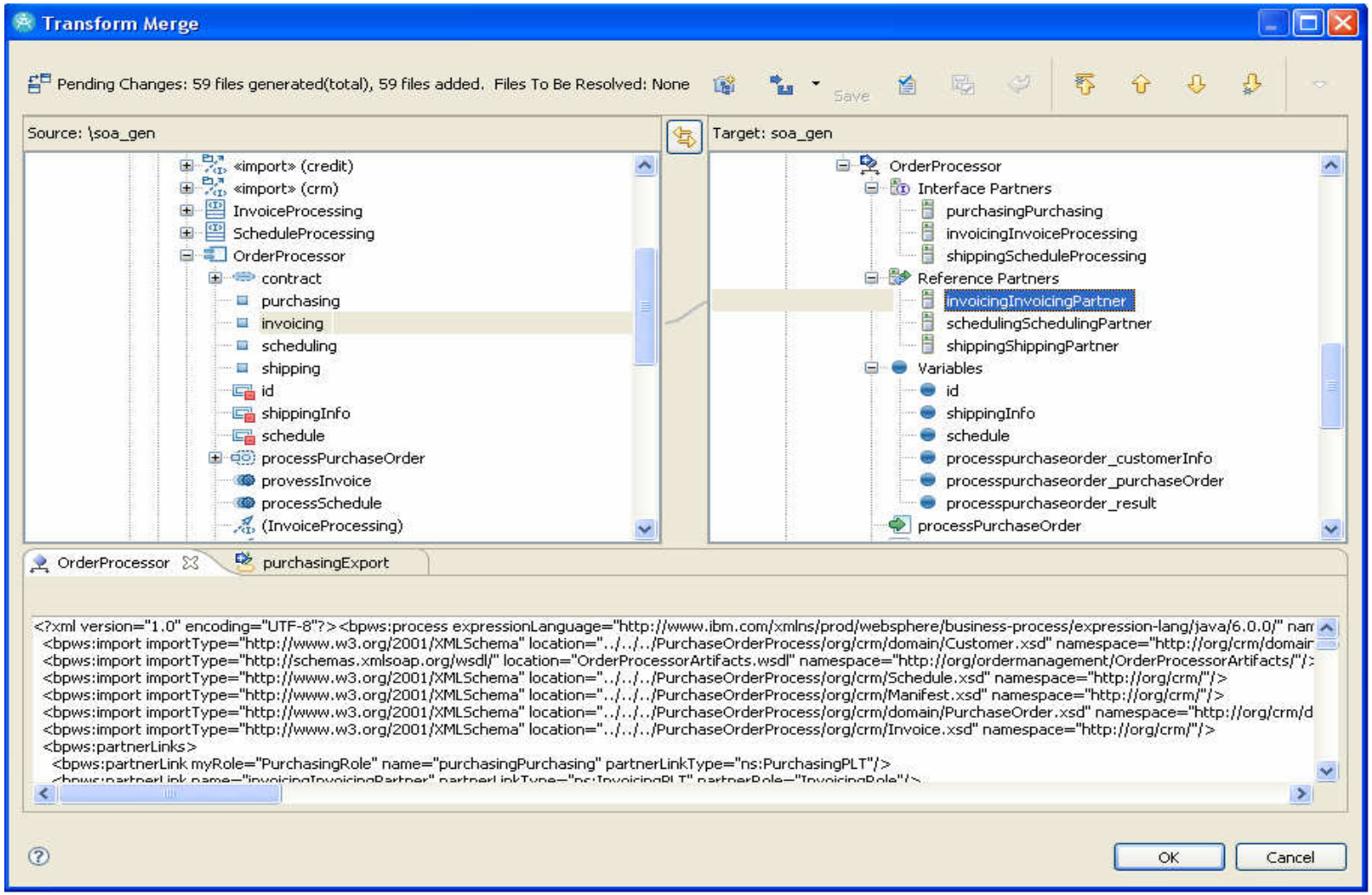
Merge Enhancements

- New “cross-domain” merge view
- Provides visual mapping of UML source with generated domain elements (XSD, WSDL, SCDL, BPEL)
- Deltas in target elements associated with UML source

Merge Enhancements

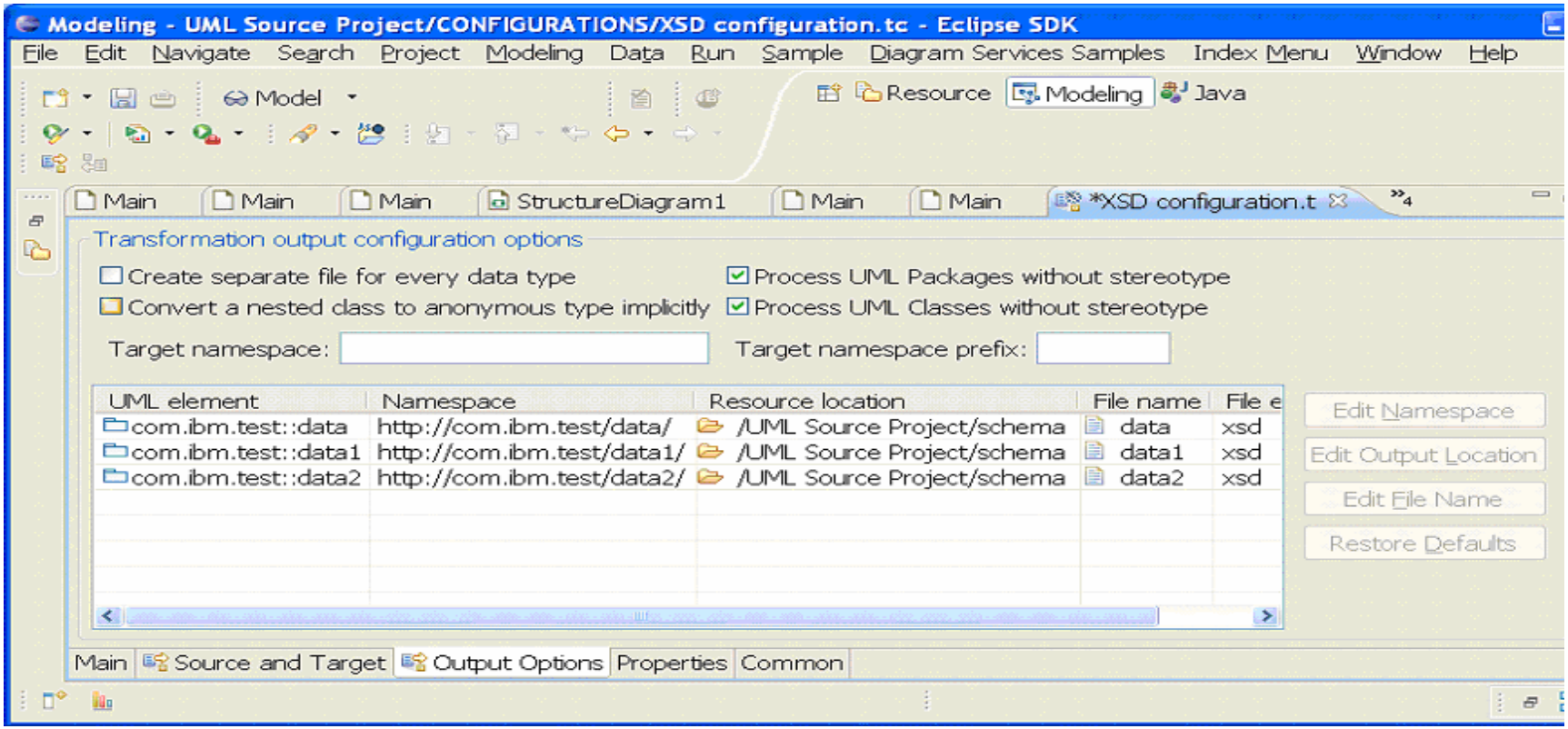


Merge Enhancements



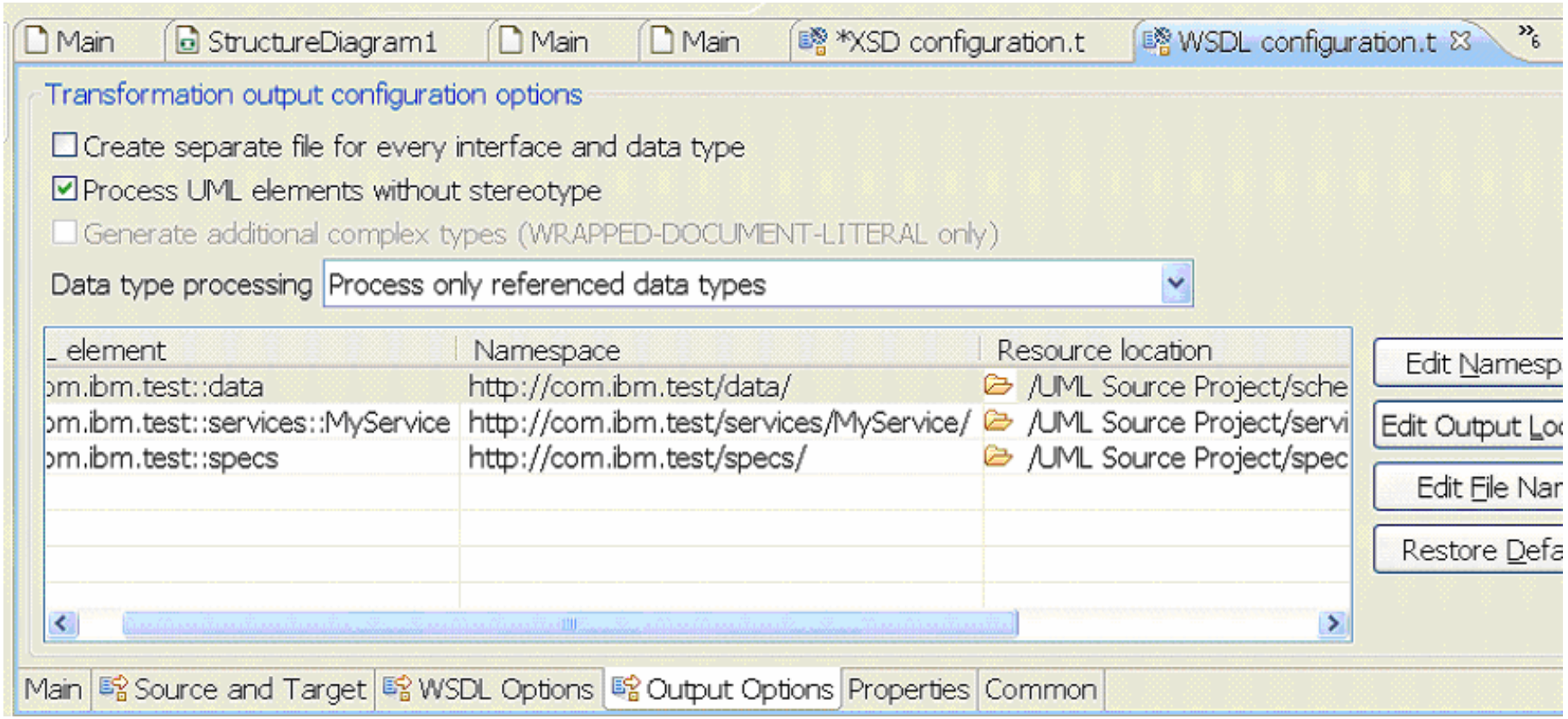
UML-to-XSD Configuration Enhancements

- New Output Options tab



UML-to-WSDL Configuration Enhancements

- New Output Options tab



AM03

What‘s New in RSA and RSM 7.5

Model-Driven Development (MDD)

Authoring Tools and Solutions

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

IBM

Rational. software

AM03

What‘s New in RSA and RSM 7.5

Domain-Specific Modeling

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

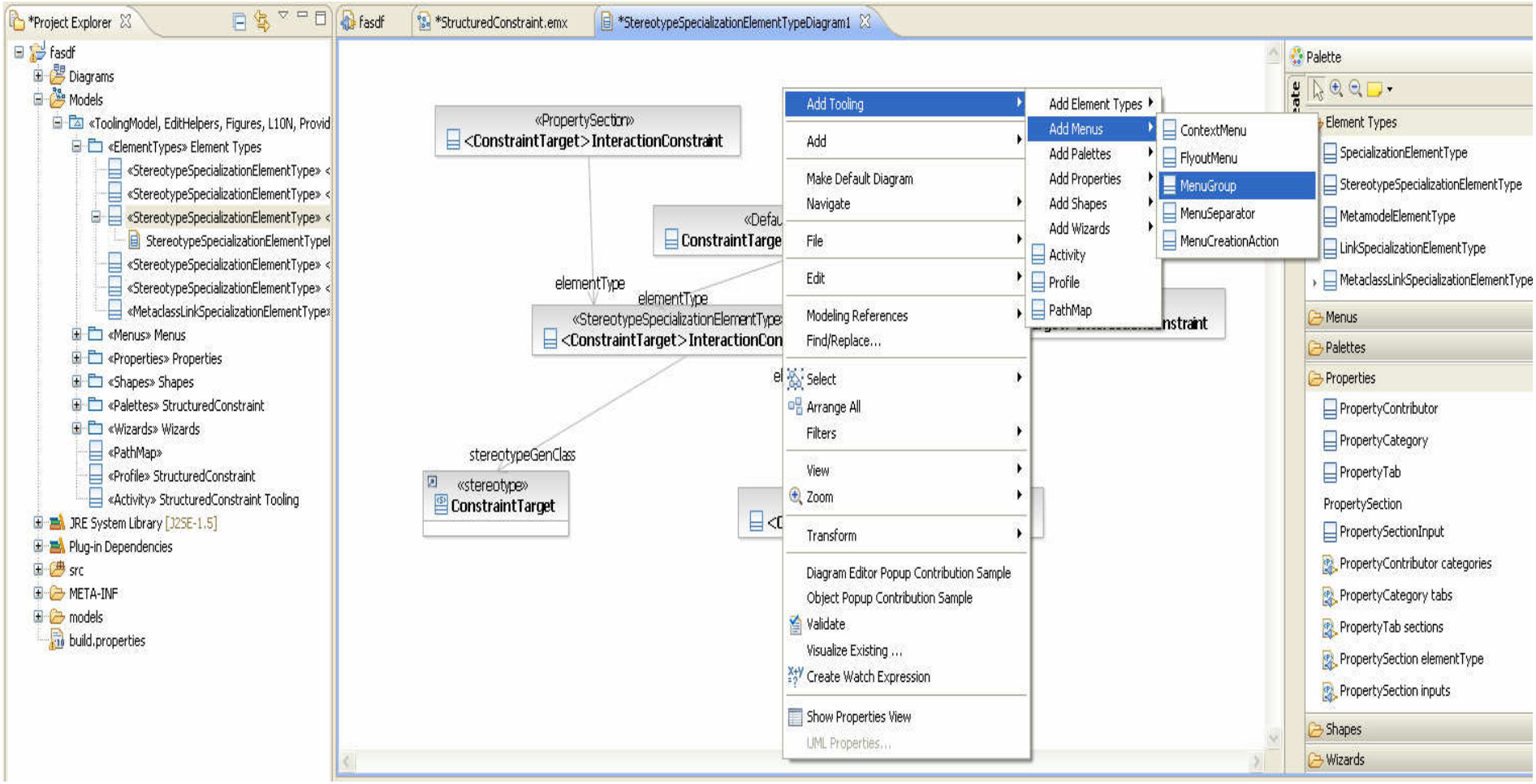
IBM

Rational. software

Profile Tooling Enhancements

- Generate deployed profiles or profiles outside of the workspace
- New visual editing approach
  - Tooling architects customize their tooling using diagrams, palettes, menus etc
- Query-based profile tooling diagrams give tooling architects an easy way to visualize their tooling
- Generated code supports palettes and menus per diagram
- New custom property section support for generated tooling code

Profile Tooling Enhancements



AM03

What‘s New in RSA and RSM 7.5

Transformation Core Technologies

IBM Rational Software Development Conference 2008

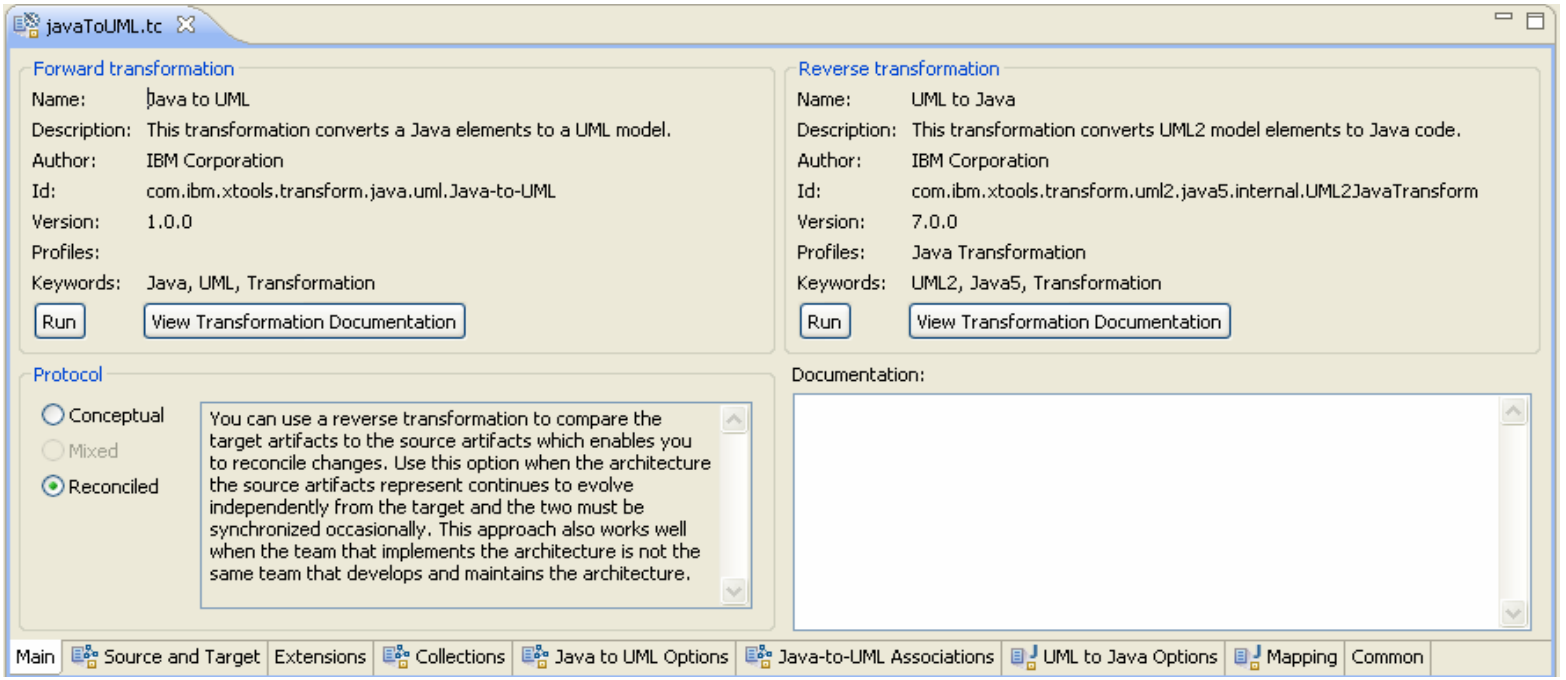
WHERE TEAMS ARE R-HEROES

IBM

Rational. software

Transform Wizard Simplification

- Drive a Transform Configuration by Intent
  - Specifies the “type” of modeling (Design Contract Modeling Protocol) intended – conceptual or sketch modeling, mixed modeling or reconciled modeling
  - DCMP selection drives other transformation configuration properties.



Transform Automation at Command Line

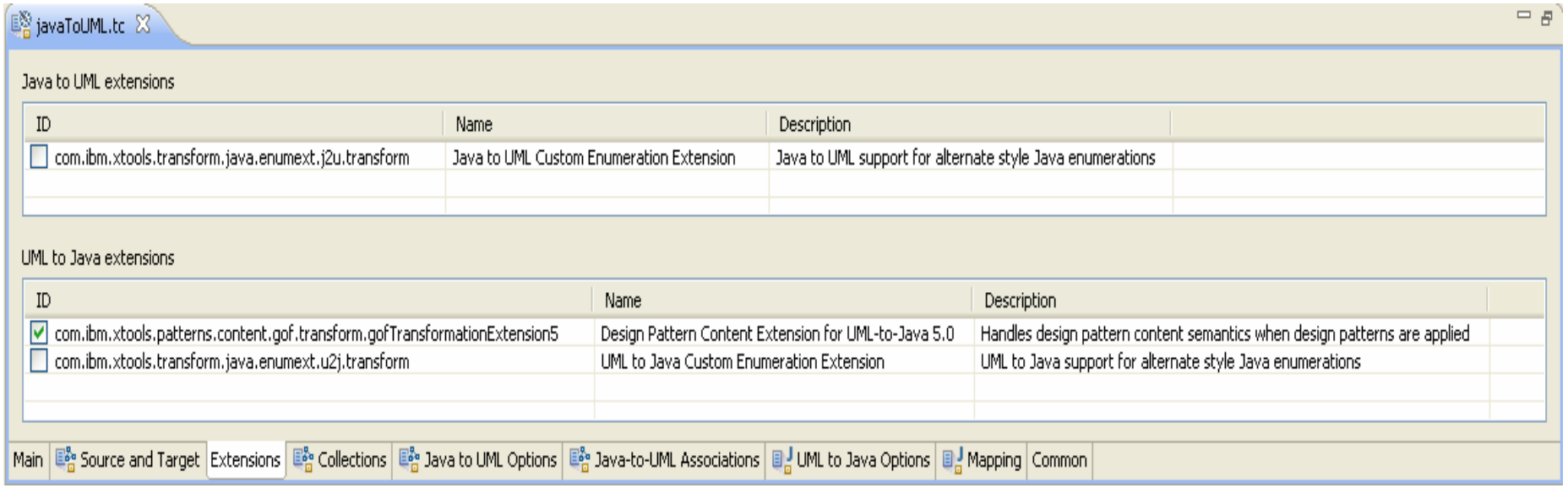
- Can run a transformation from command line using ANT
- Very useful for builds and other automatic tasks

build.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project name="hw" default="generate" basedir=".">
    <target name="generate">
        <com.ibm.xtools.transform.core.runTransformation
            transformConfig="models/java.tc" reverse ="false"/>
    </target>
```

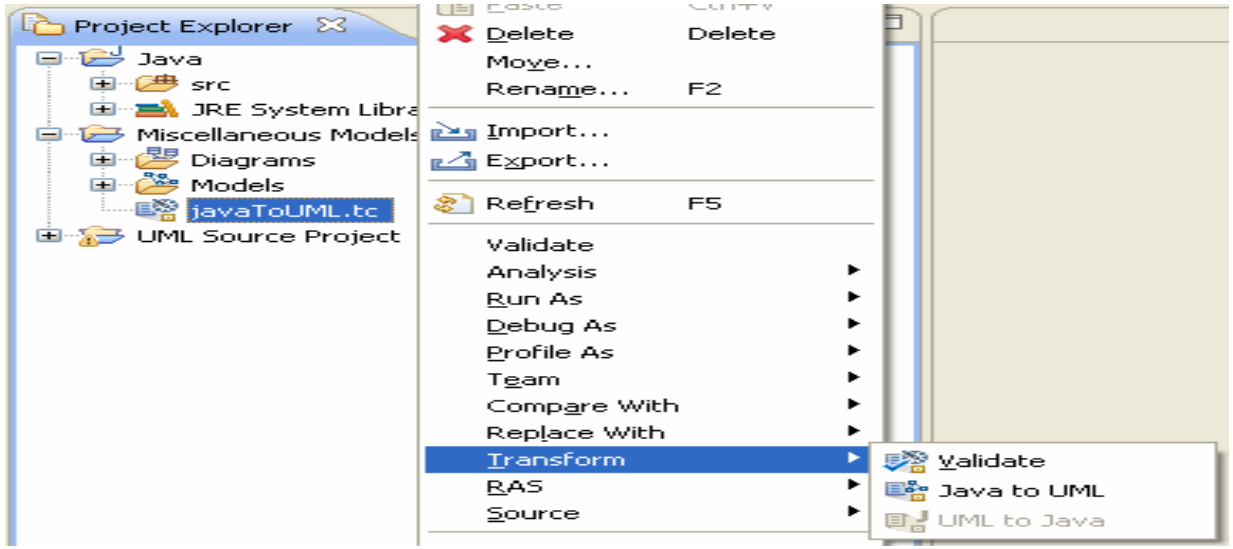
Transform Instances

- Transformation extensions configurable in a transform configuration
  - Introduces the concept of a “transformation instance” usable separately
  - Contribute tabs to the user interface
  - Participate in the transformation context validation
  - Enable or disable extensions for an instance at any time



Transform Core

- New documentation property for transformation configurations
- Transformation configurations can be programmatically loaded from any URL
- New context menu to validate a transformation configuration
  - Editor loads much faster



Other Transform Core Improvements

- Optionally displays log at transformation completion
- Mapping utility now uses workspace paths to locate mapping models instead of file system paths
- Collections and Maps can be transformation properties
- Introduced headless application for transformations. Previous application had a workbench dependency, preventing its use in some contexts.
- Lazy initialization of transformation configuration properties improves responsiveness
  - Property values are proxies until resolved by a fetch
  - No longer resolves entire configuration to retrieve a property value
  - Long operations such as opening models delayed until required

AM03

What‘s New in RSA and RSM 7.5

Model to Model Transformations and Authoring

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

IBM

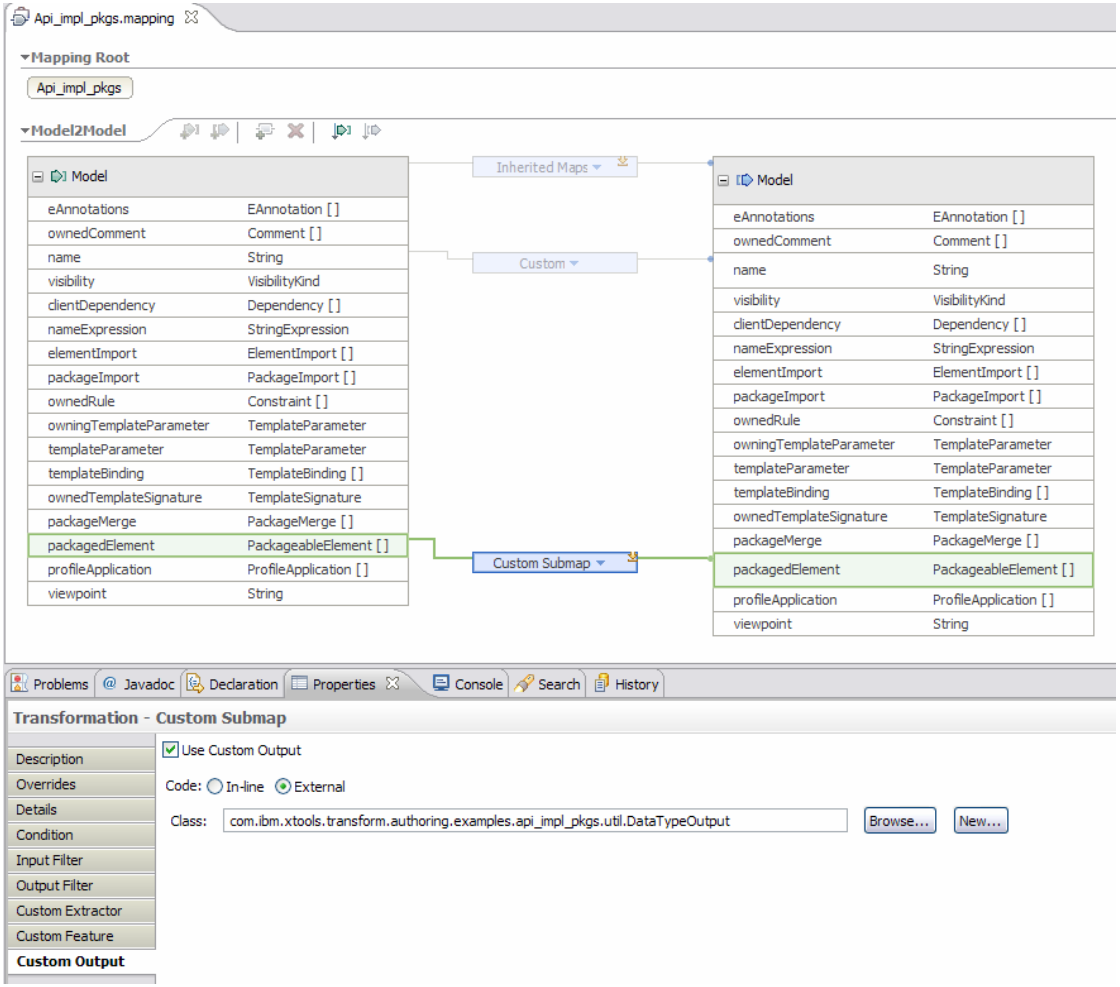
Rational. software

M2M Transform Authoring Improvements

- Additional mapping types and predefined rules
- More extensive UML profile support
- Many usability improvements
- Flexible transformation source code generation

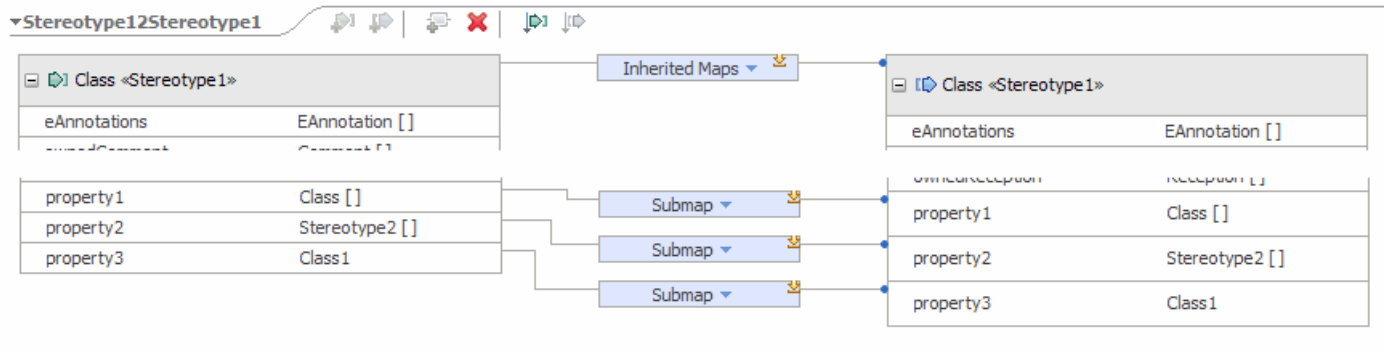
M2M: Additional mapping types and predefined rules

- Custom sub-maps simplify the task of generating objects in the target model that don't have explicit corollaries in the source model
- ChainRule supports connecting one transformation to another

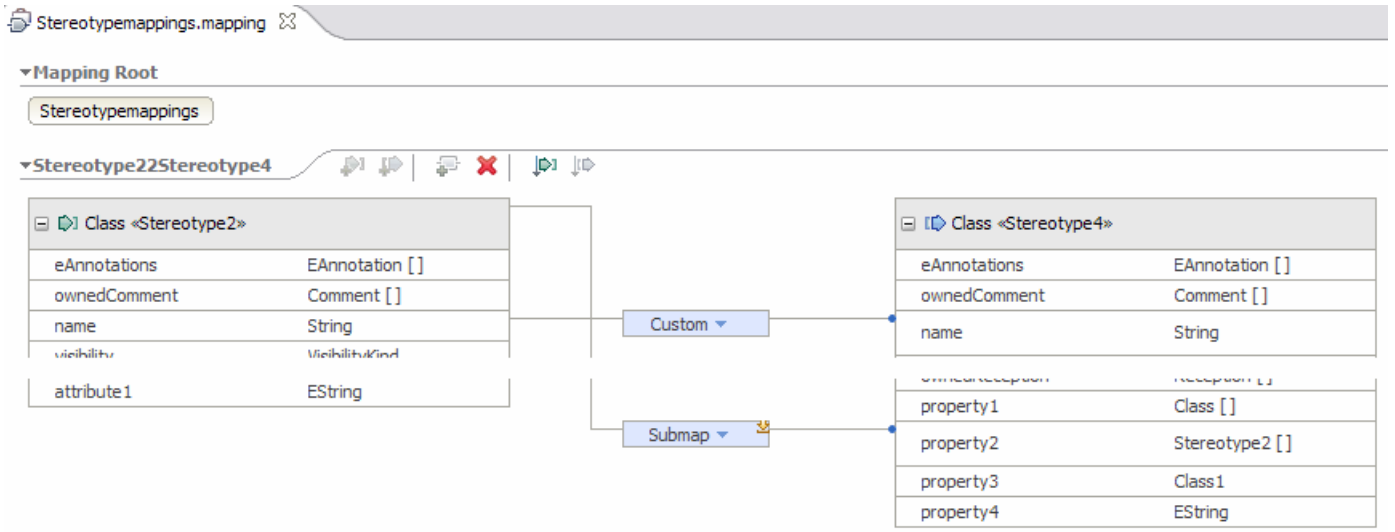


M2M Extensive UML profile support

- Mappings with stereotype properties whose type is a stereotype or profile defined class

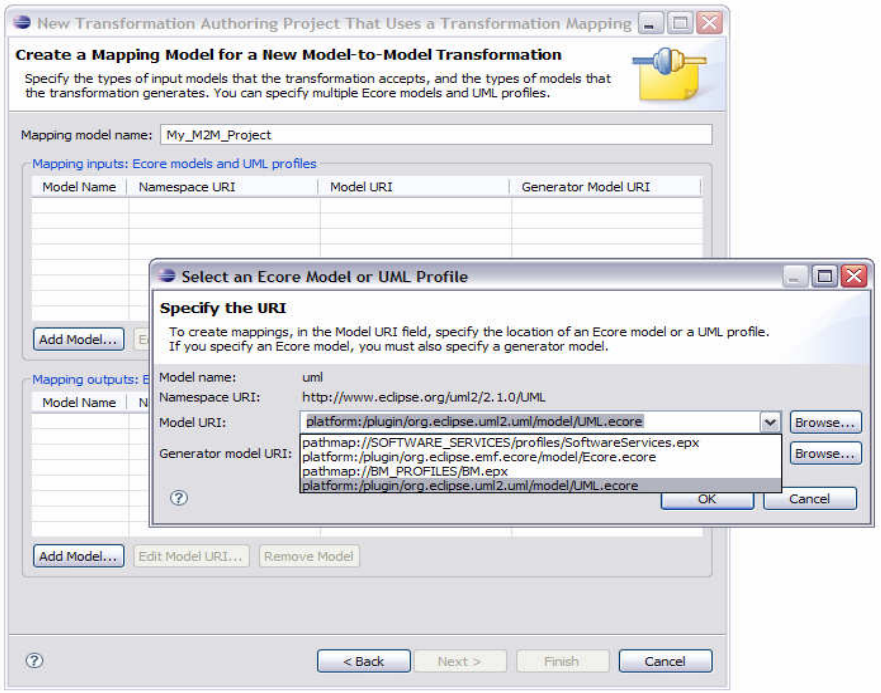


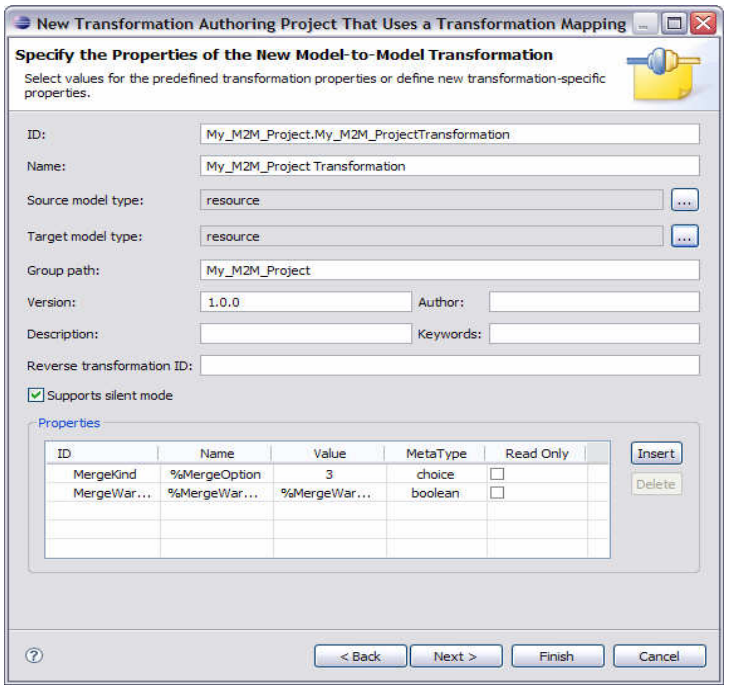
- Mappings with stereotype elements



M2M Usability Improvements

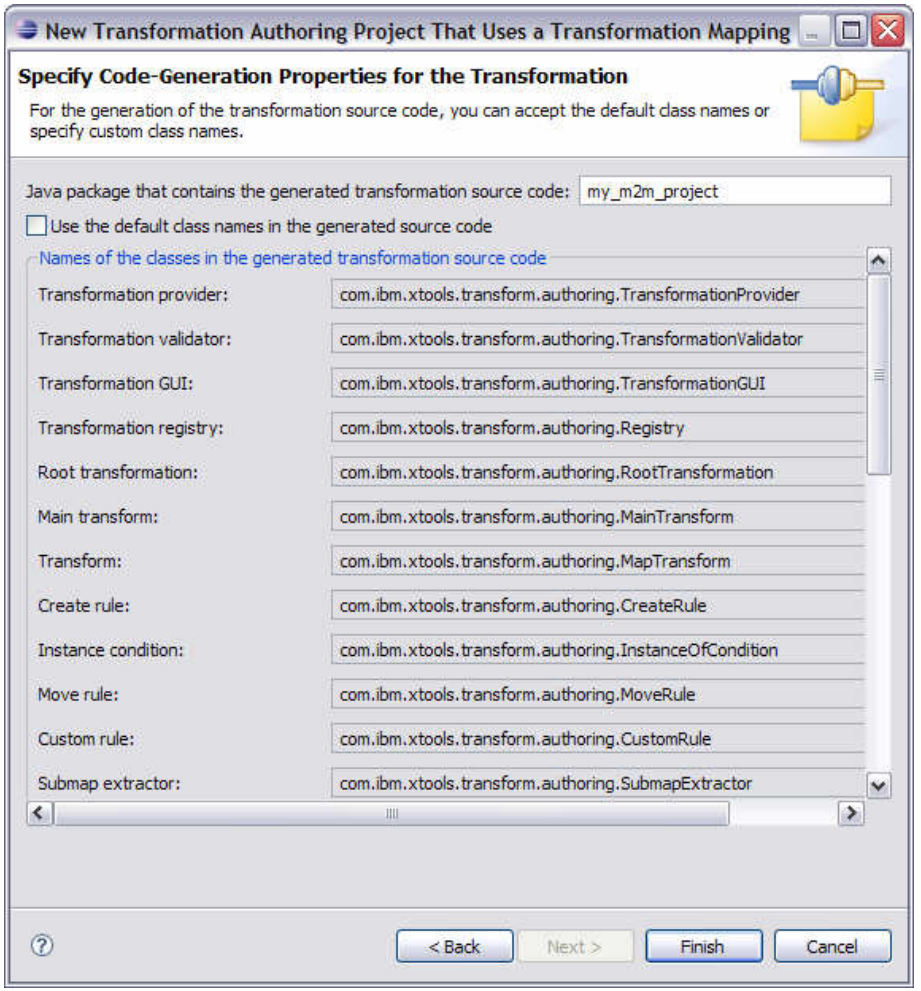
- Paths to mapped models are now editable post-project creation
- New version of the New Transformation Authoring Project Wizard





M2M Flexible Transform Source Code Generation

- Target user customized transformation framework
- Set workspace defaults as well as project specific code generation properties



AM03

What‘s New in RSA and RSM 7.5

Model to Text Transformations and Authoring

IBM Rational Software Development Conference 2008

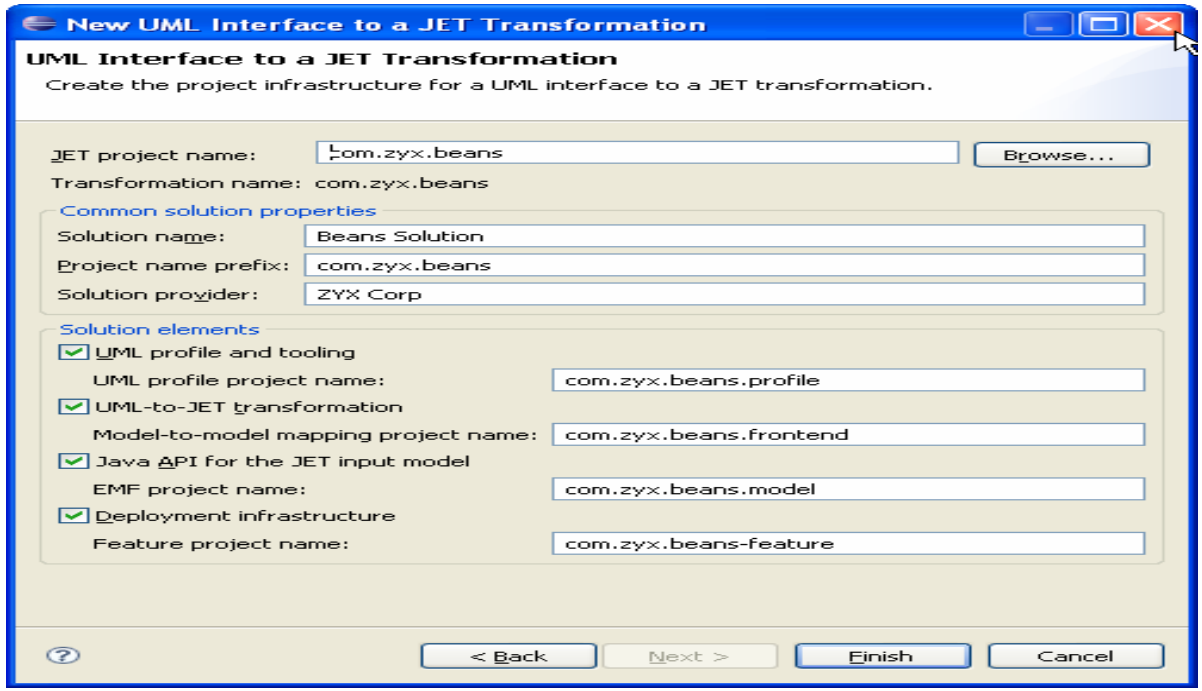
WHERE TEAMS ARE R-HEROES

IBM

Rational. software

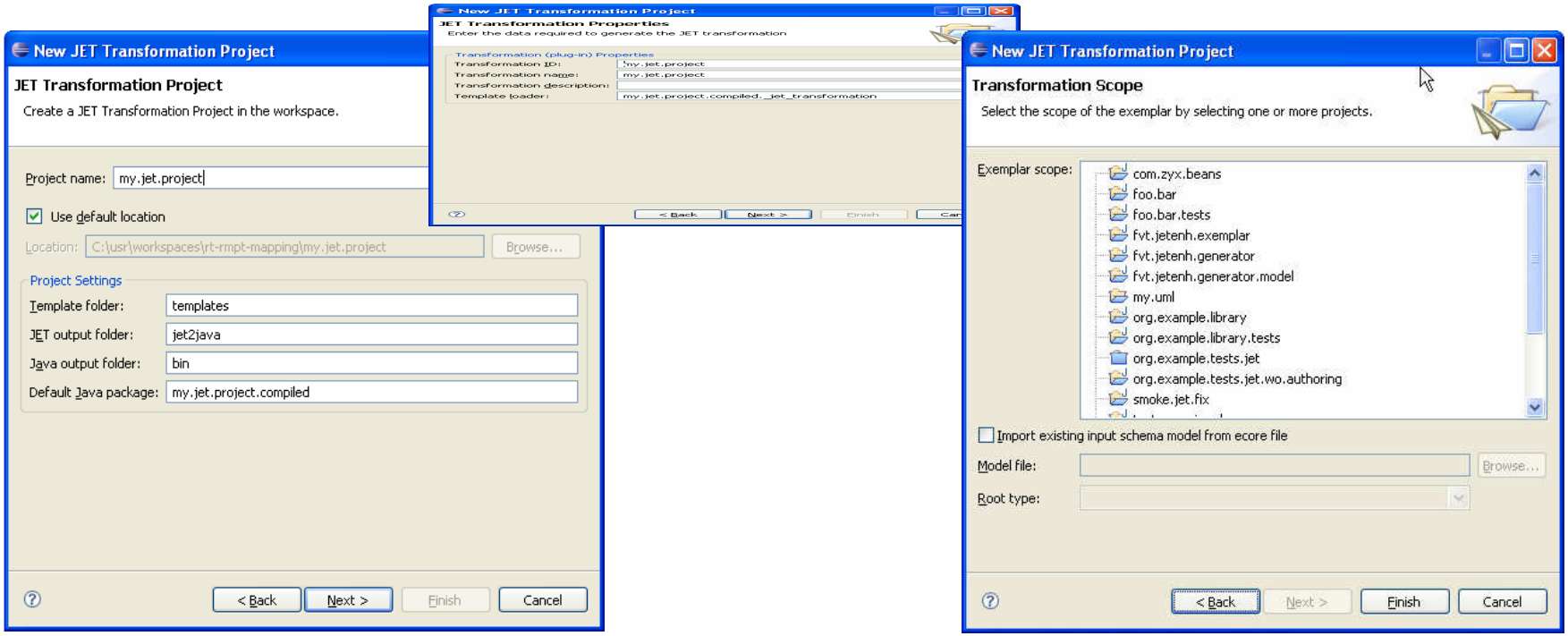
JET Authoring Next Steps Wizard

- Avoids ~20 minutes of manual steps in developing a UML interface to a JET transformation.
- Equivalent to running four other wizards



JET Authoring Common Tasks Streamlining

- ‘Actions’ may now be conditional
- New JET project wizard now includes prompts for more information



JET Engine (Open Source)

- Improved performance
  - No file writes without change
    - Improved compatibility with pessimistic locking repositories like ClearCase
  - Many hot spots removed
- User-driven improvements
  - Sort function
  - Support more XPath axes
  - Improved testability of JET functions producing random values
  - Output indentation tags

AM03

What‘s New in RSA and RSM 7.5

Model-Operative Patterns

IBM Rational Software Development Conference 2008

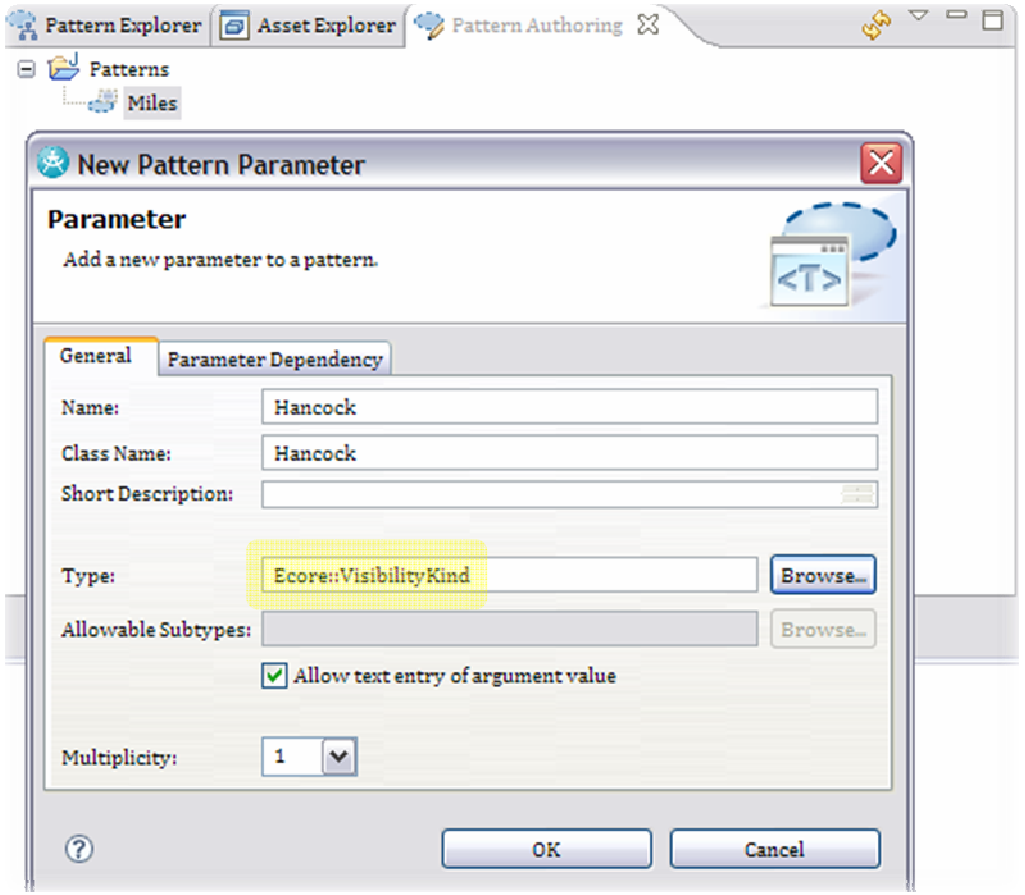
WHERE TEAMS ARE R-HEROES

IBM

Rational. software

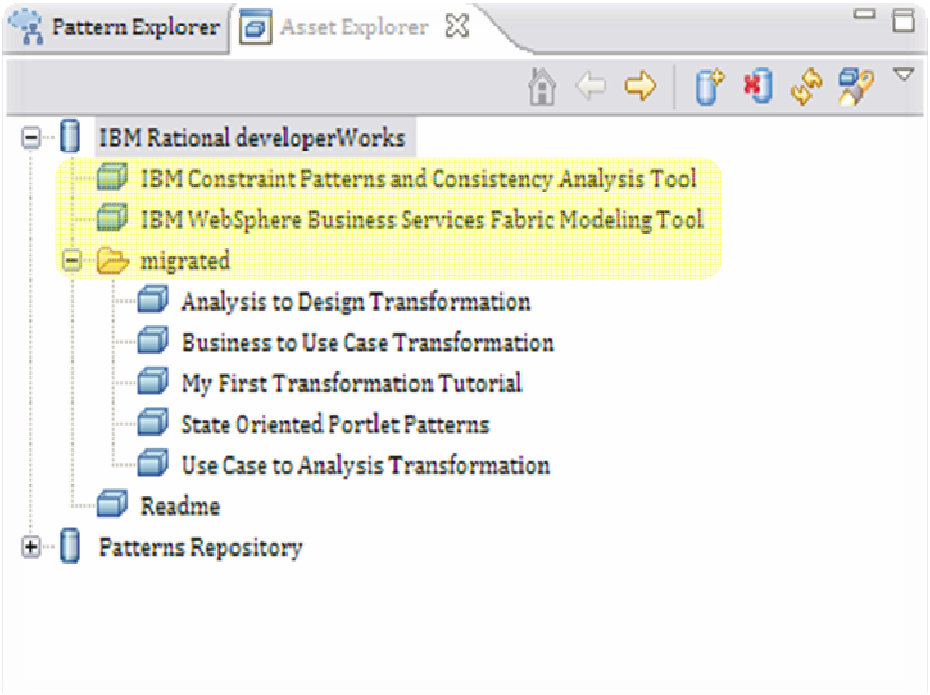
Patterns Enhancements I

- Design pattern content now uses the specialized pattern framework which enables better handling of read-only pattern argument model elements during pattern application
  - Pattern instances now own the trace relationships to involved elements
- UML profile support at authoring and application time now allows profile enumerations to be used as pattern parameters



Patterns Enhancements II

- RSx developerWorks RAS repository migration completed for the relevant and most popular assets
  - Two newly published assets are also available related to constraint patterns and business fabric modeling
- Minor incremental pattern authoring user interface enhancements



AM03

What‘s New in RSA and RSM 7.5

RSM and RSD 7.0.5 Features Migrated to RSA in 7.5

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

IBM

Rational. software

7.0.5 Feature Highlights

- Diagramming improvements
- Rich text support for documentation, notes and comments
- Configurable modeling UI, viewpoints
- Modeling reminders
- Model management improvements
  - Support for diagram fragments
- UML 2.1 support improvements
  - Support information flow elements (UPDM requirement)
  - Deployment diagram and sequence diagram enhancements
  - Query-based diagram population

7.0.5 Feature Highlights

- BIRT-based reporting for UML/EMF models
- Model analysis and metrics
- UPDM support (beta 1 spec)
- Improvements to reapply/reconcile scenarios in transformations

7.0.5 Extensibility Feature Highlights

- Ability to generate tooling from a UML profile
  - Palettes, menus, diagram assistants, capabilities, …
  - For building UML-based DSLs
- Reminders framework
- Queries framework
- Viewpoints
- Configurable modeling UI
- Model analysis/metrics

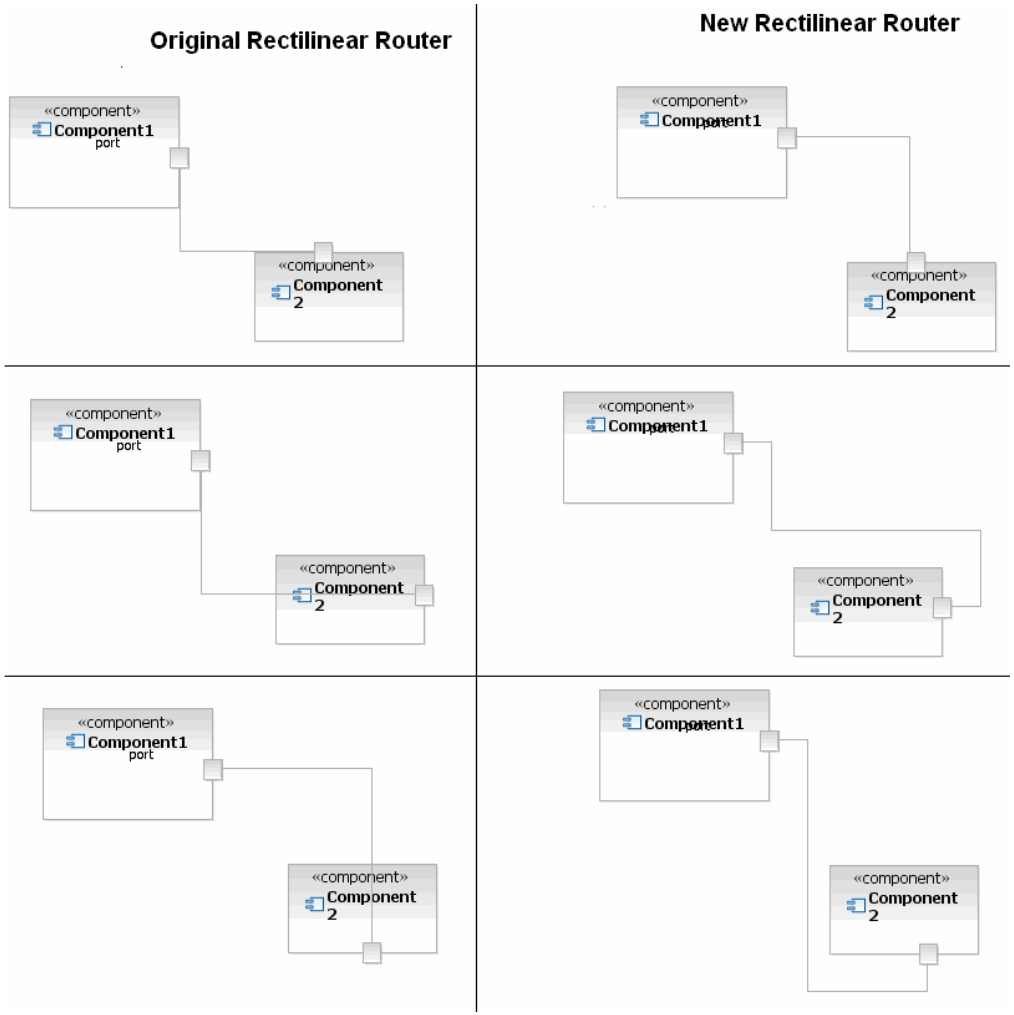
7.0.5 Extensibility Feature Highlights

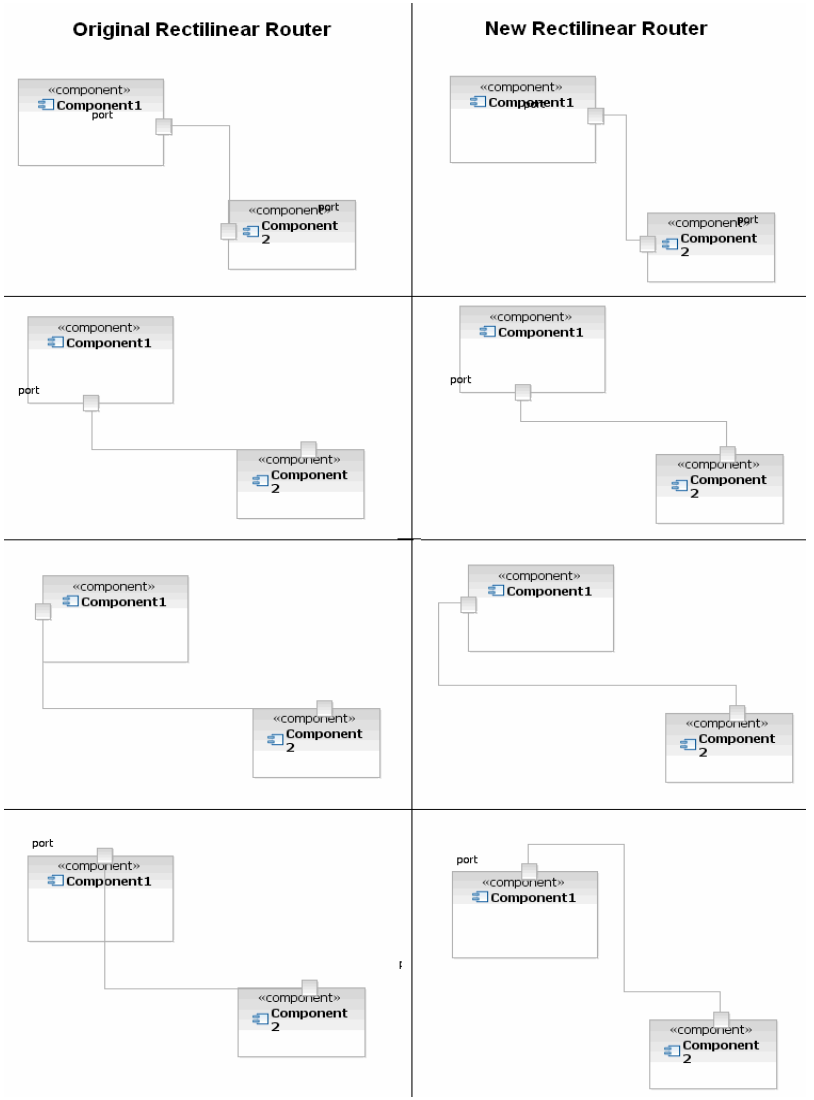
- Model2model transformation authoring enhancements
- Model2text transformation authoring enhancements
  - JET2 syntax/content aware editor
- Model operative patterns framework enhancements
- Significant improvements in open source components (GMF, EMF, UML2 …)

Rectilinear Routing Enhancements

- Improved rectilinear routing
  - Connectors no longer overlaps shape edges
  - Connectors no longer overlap other connectors
  - Connector between ports/pins no longer cross shapes owning the ports/pins
  - Smarter connector endpoint placement on ports/pins

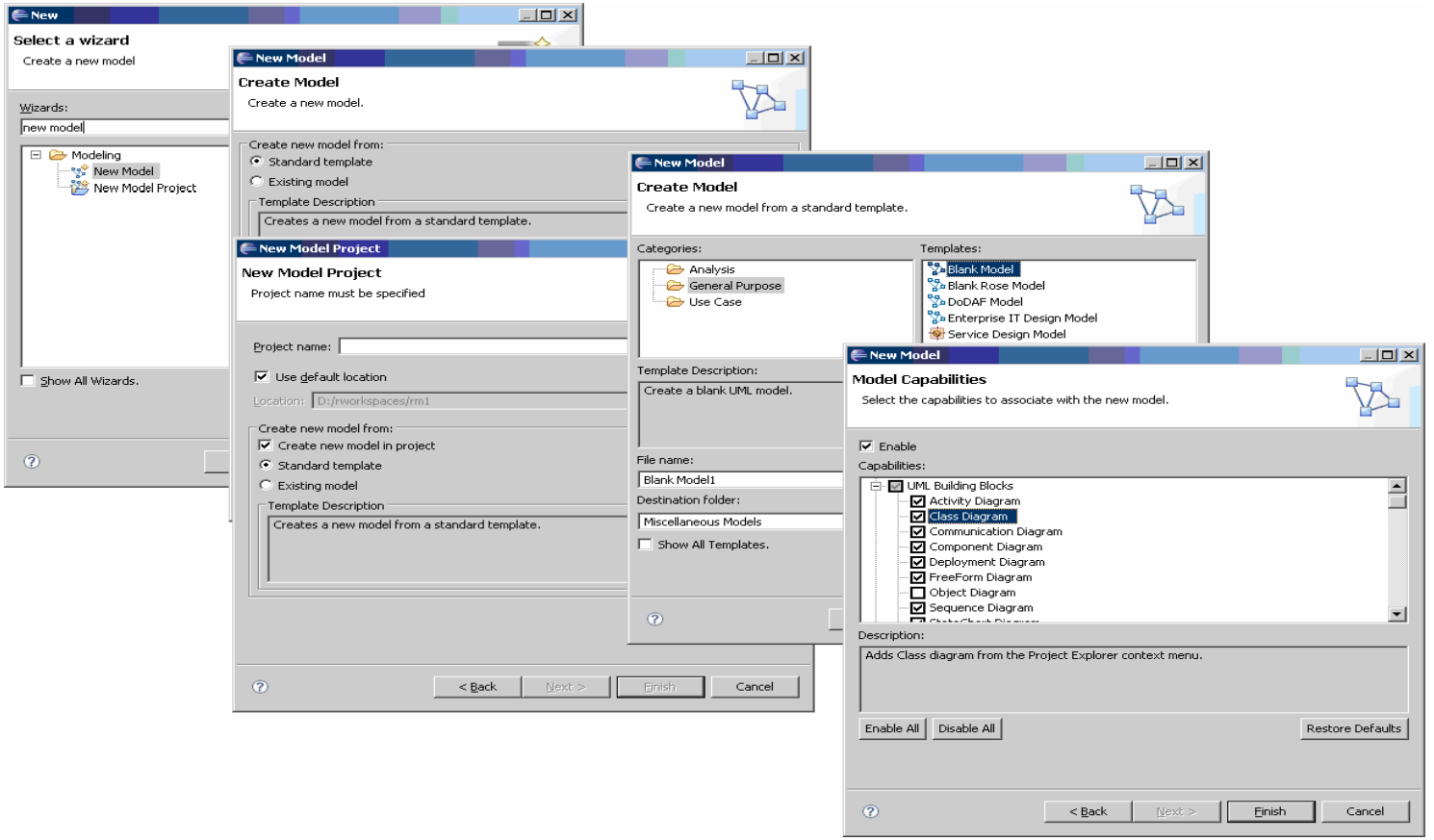
Rectilinear Routing Improvements





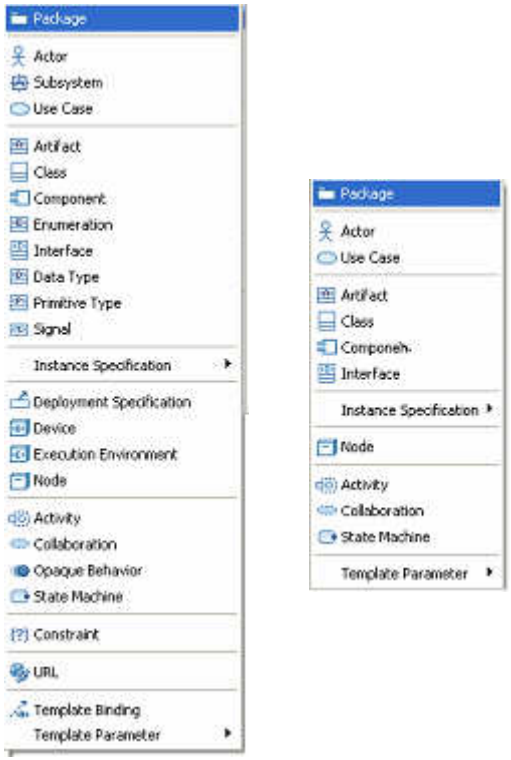
New Model and Model Project Wizard

- New Model Wizard with a new page add modeling capability context to the new model

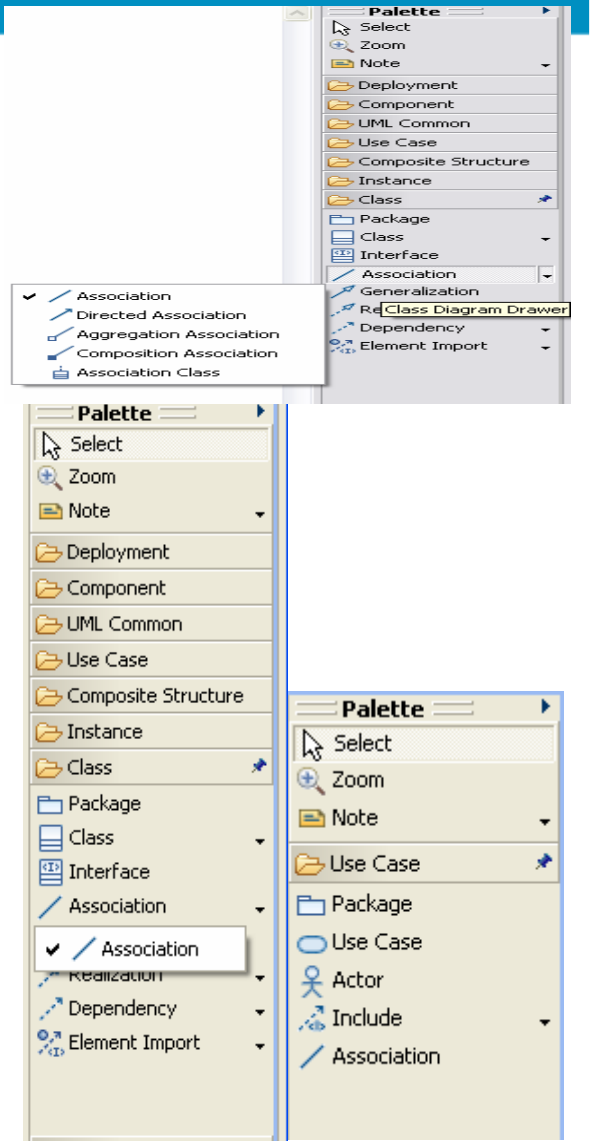


Configurable Modeling UI

- New Model wizard contains various model templates, each with preset modeling capabilities
  - UI components configured: Palette, content menus and action bar
  - Use can configure via modeler capabilities page
- Existing templates that can be shared among team members



Context menu:
Blank Model vs. Blank Rose Model

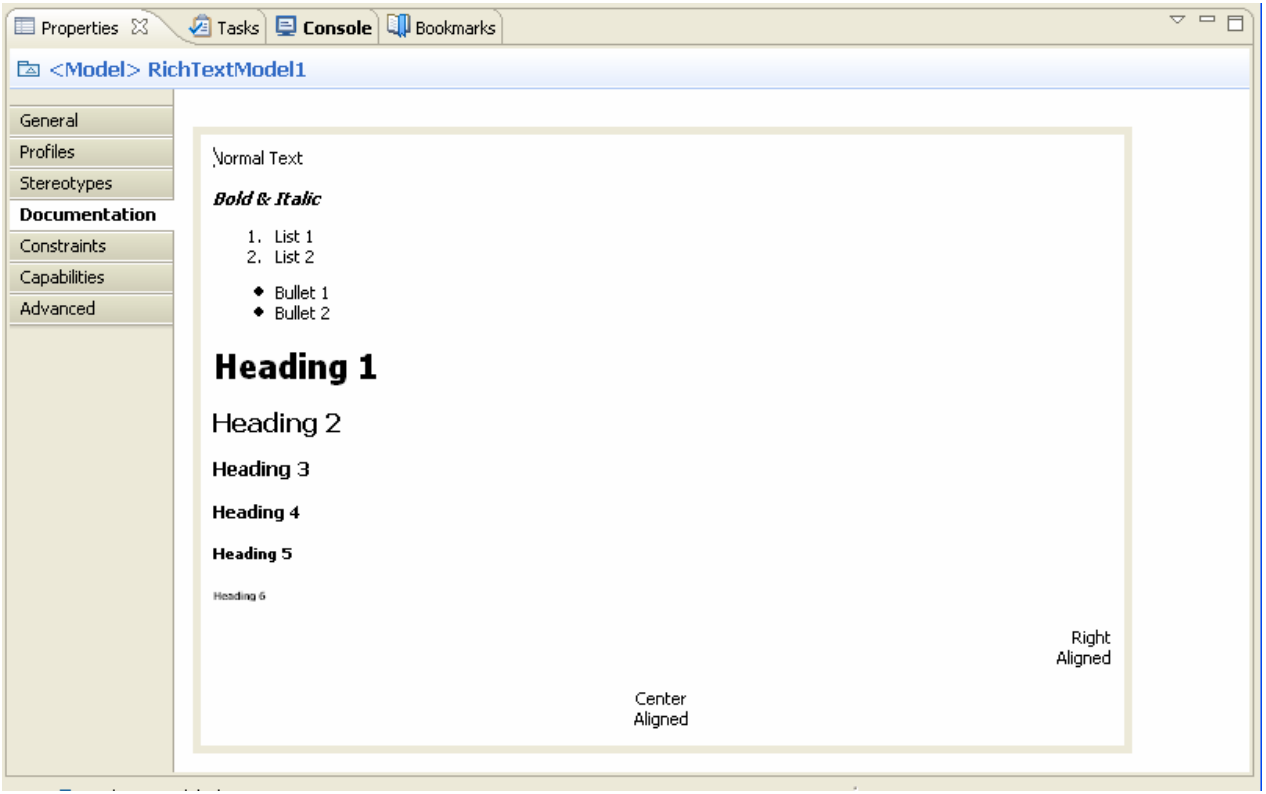


Palette Blank Model vs. Blank Rose Model vs. Use case Model

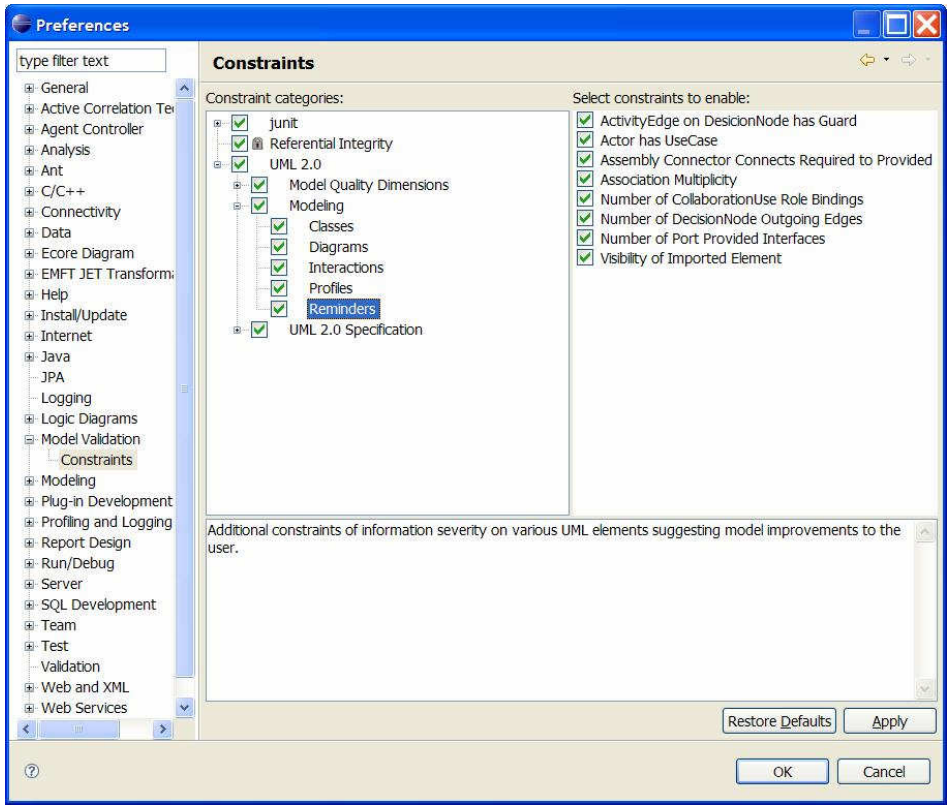
Viewpoints Properties Page

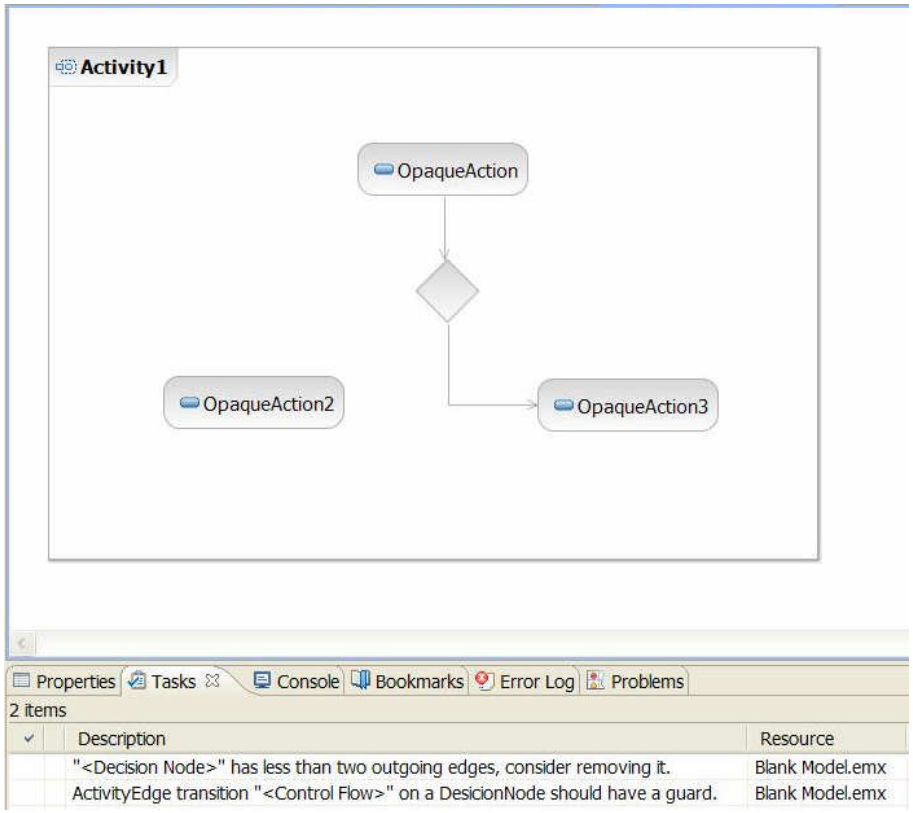
Rich Text Support

- Rich Text Support added to
  - Documentation tab
  - Note & Comment shape



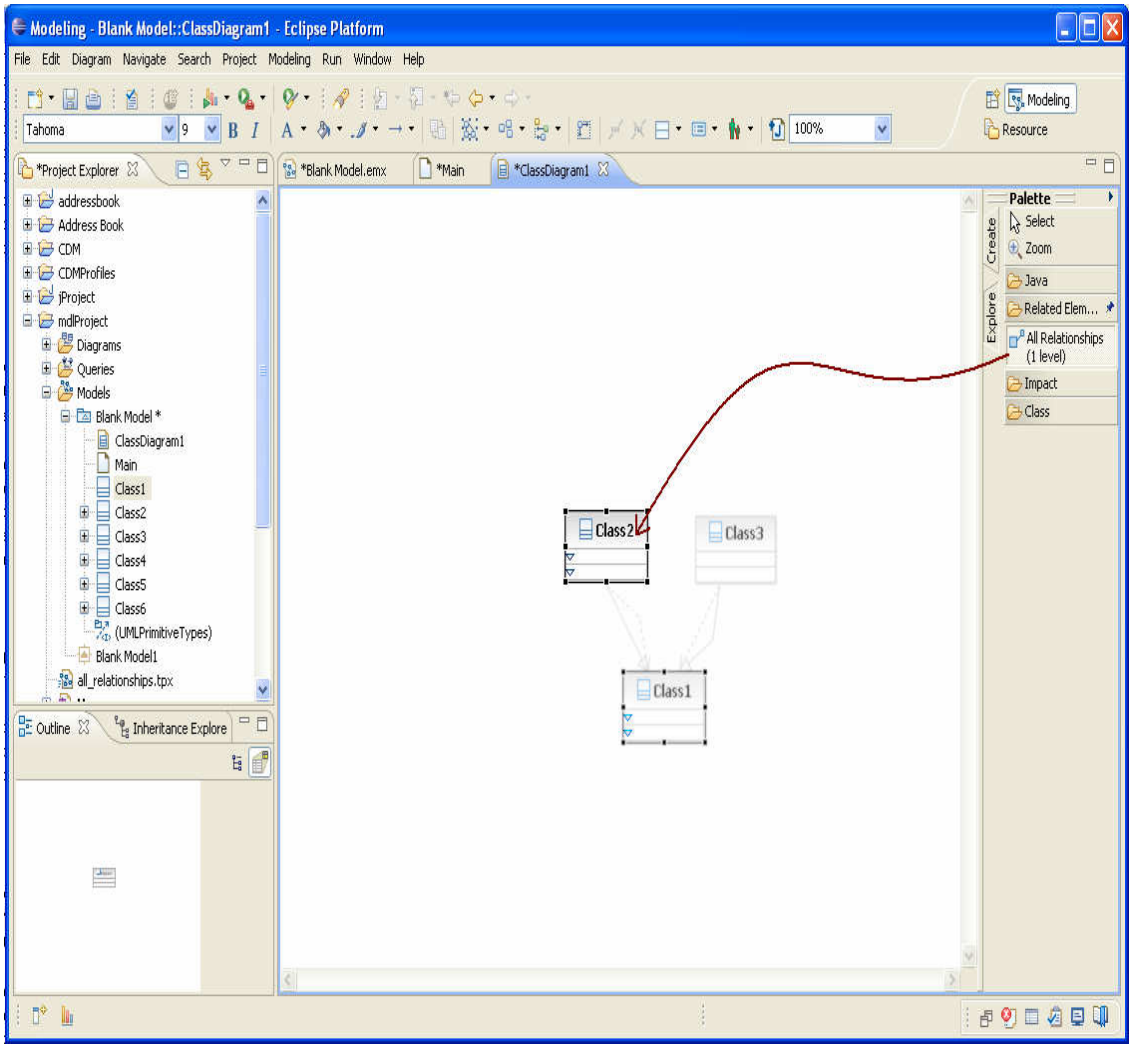
Modeling Reminders





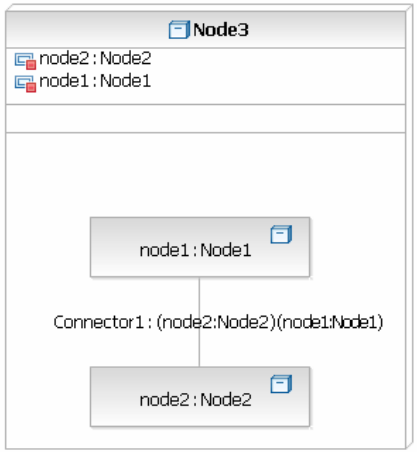
Build Diagram using Queries

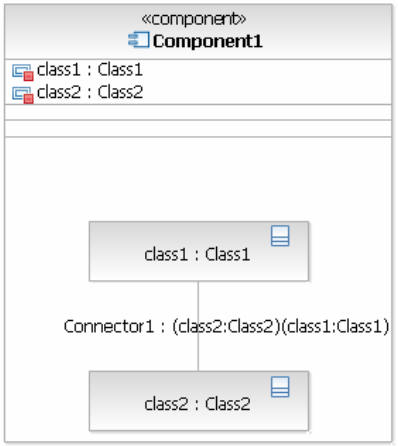
- Tabbed Palettes
- Creation and Explore Palette
- Explore Palette with Query Tools
- Trace, Java and Domain Modelers planning to use the explore palette to contribute existing topic presets.

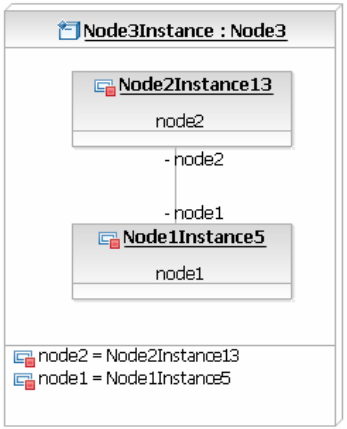


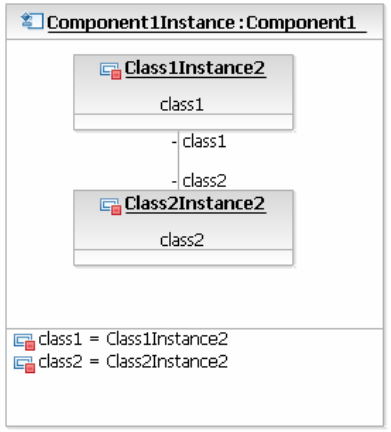
Deployment Diagram Enhancements

- Support for Slot Structure Compartments in Node Instances and Component Instances
- Support for Slot Values for attributes with multiplicity more than 1.
  - Specifying values for attributes whose multiplicity is > 1 is now supported
  - Specifying values for attributes that aren't typed by primitive types (e.g. integer, string) is now supported.
- Enhancement to Link Creation and Slot Value Settings
  - Link source and targets are now set to the source and target object instances.
  - Link ends now correspond to the typing association's role ends
- Improved Modeling Assistant Capabilities
  - Added Dependency to the list of valid choices in the diagram assistant for instance modeling.
  - Diagram assistant support for <<deploy>> is now enhanced.
  - Improved Show/Hide relationships on <<deploy>> and <<dependency>> for deployment/instance diagrams.
- Improved Usability Enhancements
  - Added Dependency tool to the Deployment palette
  - Link is auto-populated when possible; otherwise, default select element behavior is used.
  - Support creating nodes as parts of nodes (disallow creation of parts typed by other classifiers).



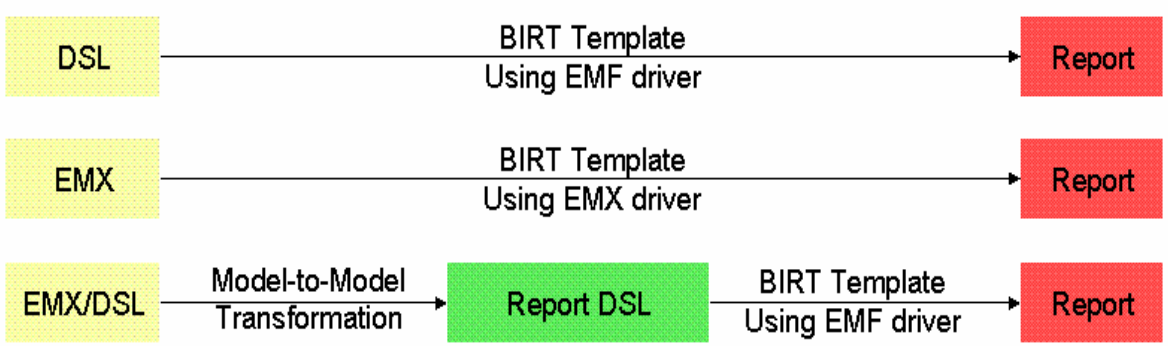




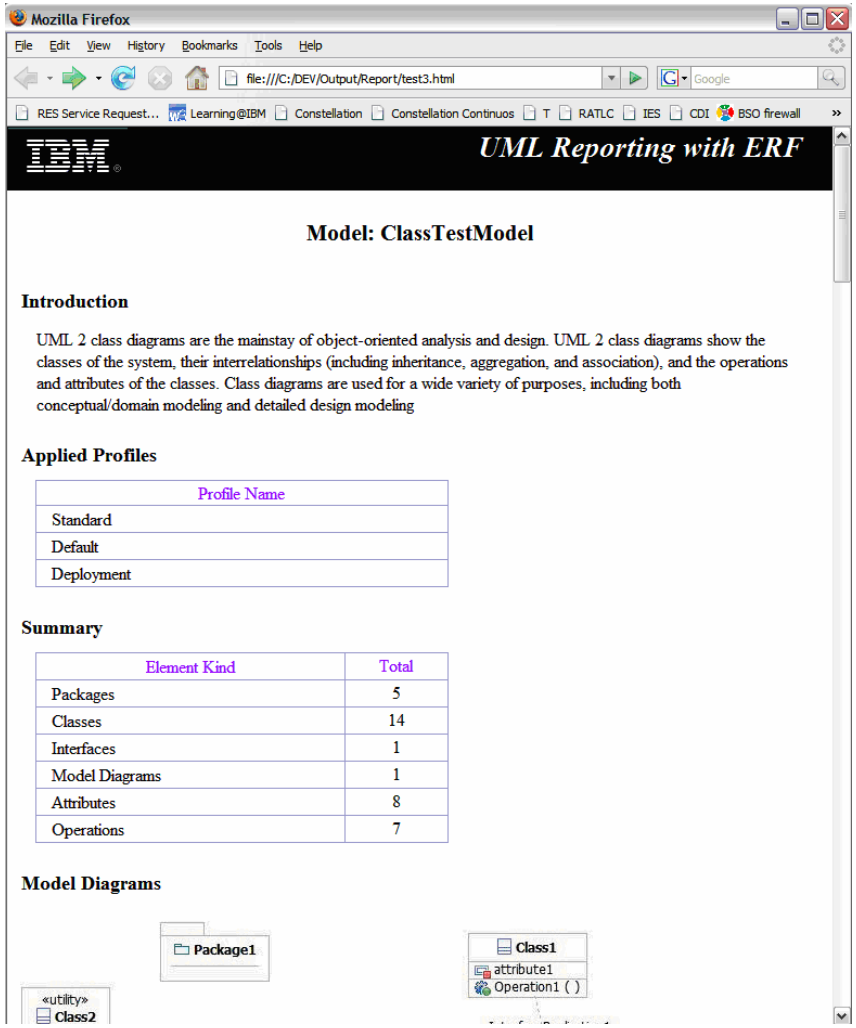


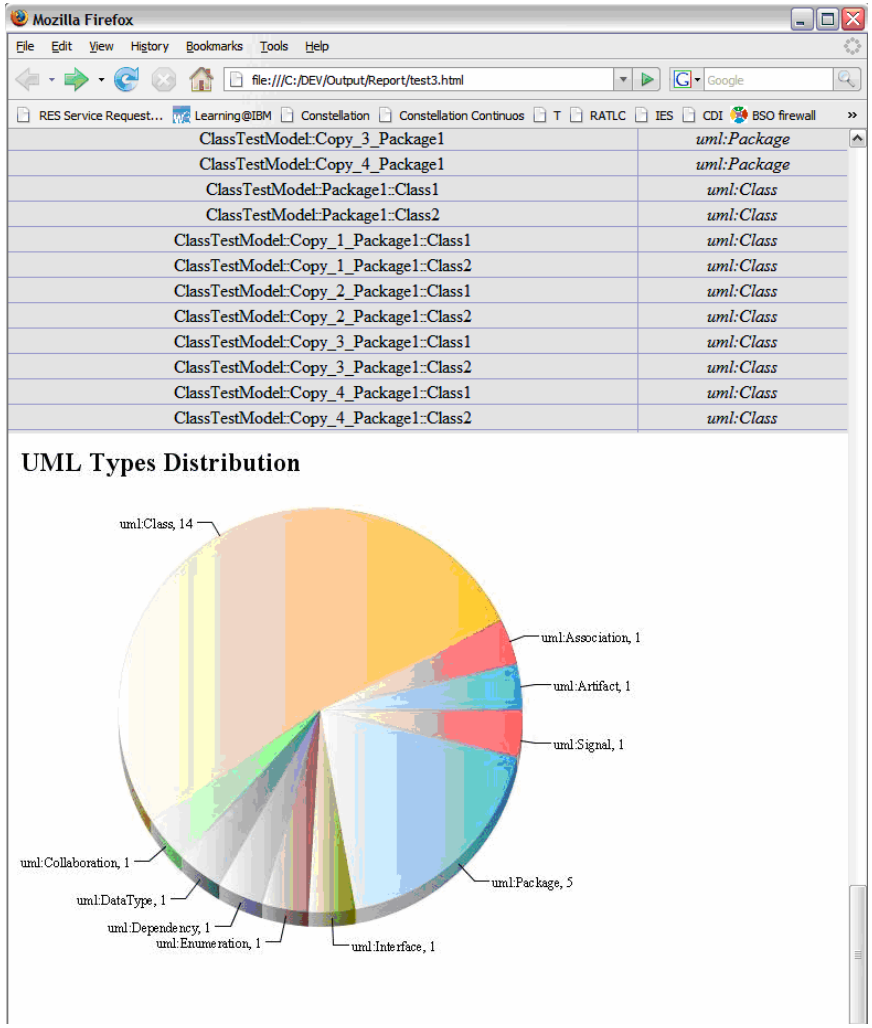
BIRT Model Reporting

- ODA driver for EMF-friendly access to data
  - OCL and XPath as query languages
- ODA driver for EMX access to UML schema + profiles, notation, requirements...etc
- Handles complex projections or computations with BIRT scripting and M2M transform as first phase

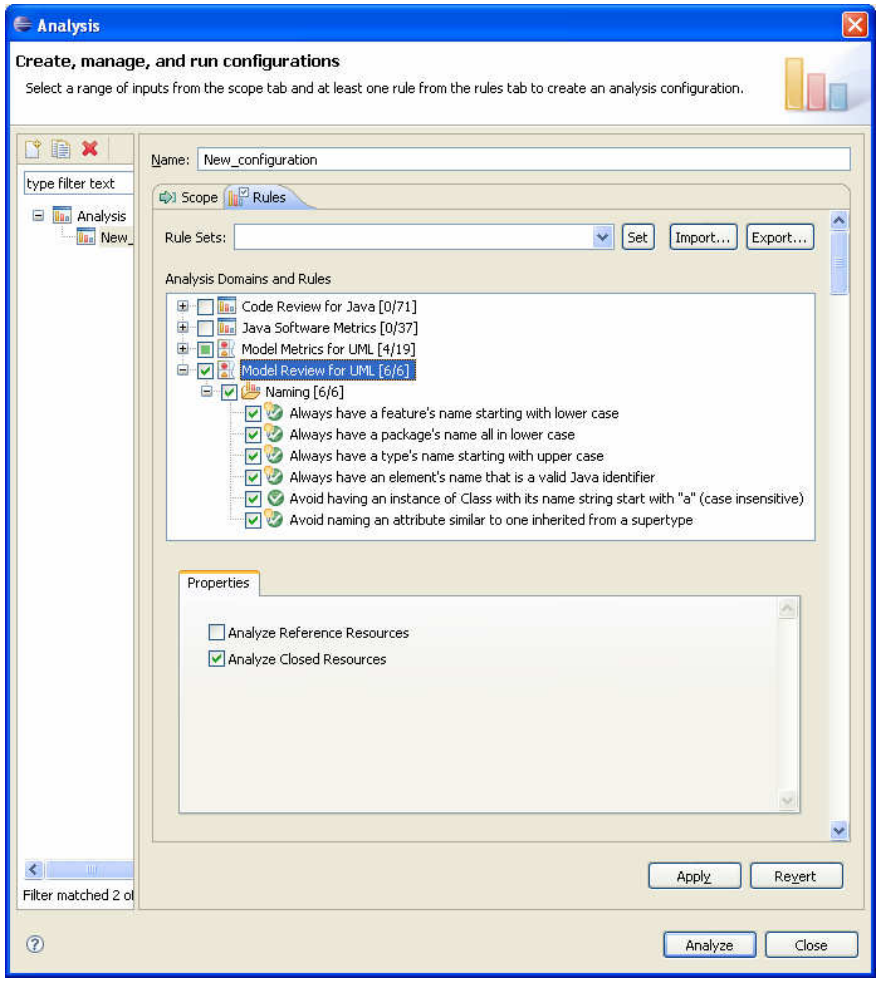


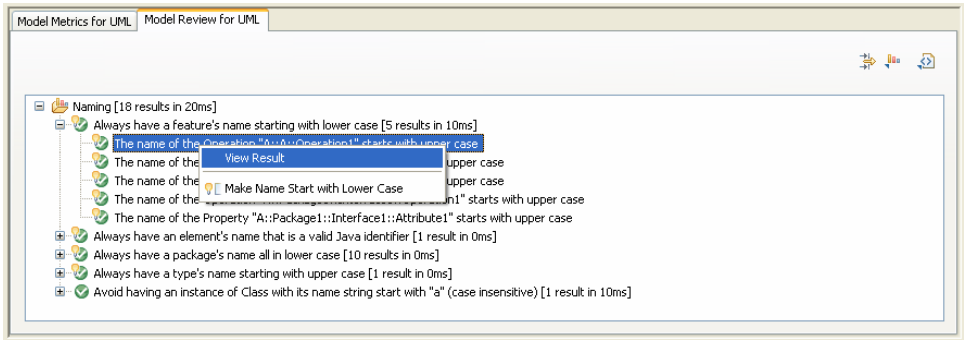
Sample UML report

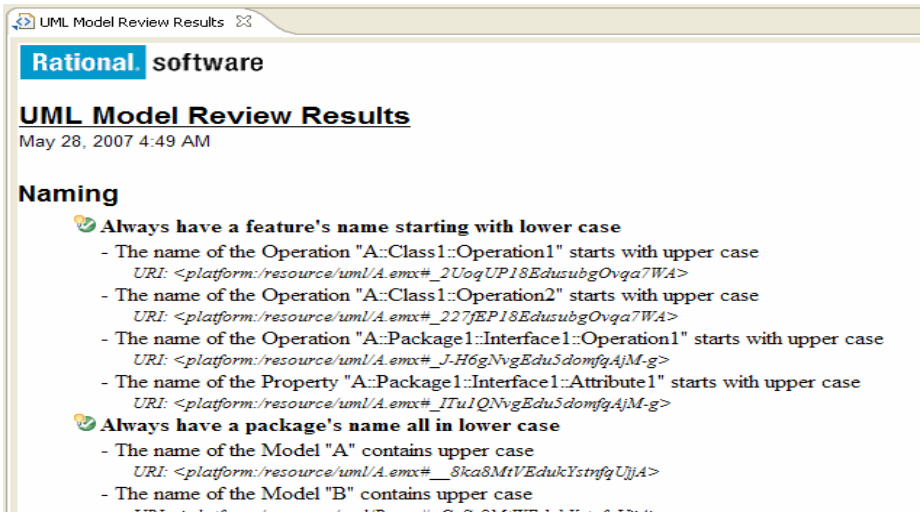




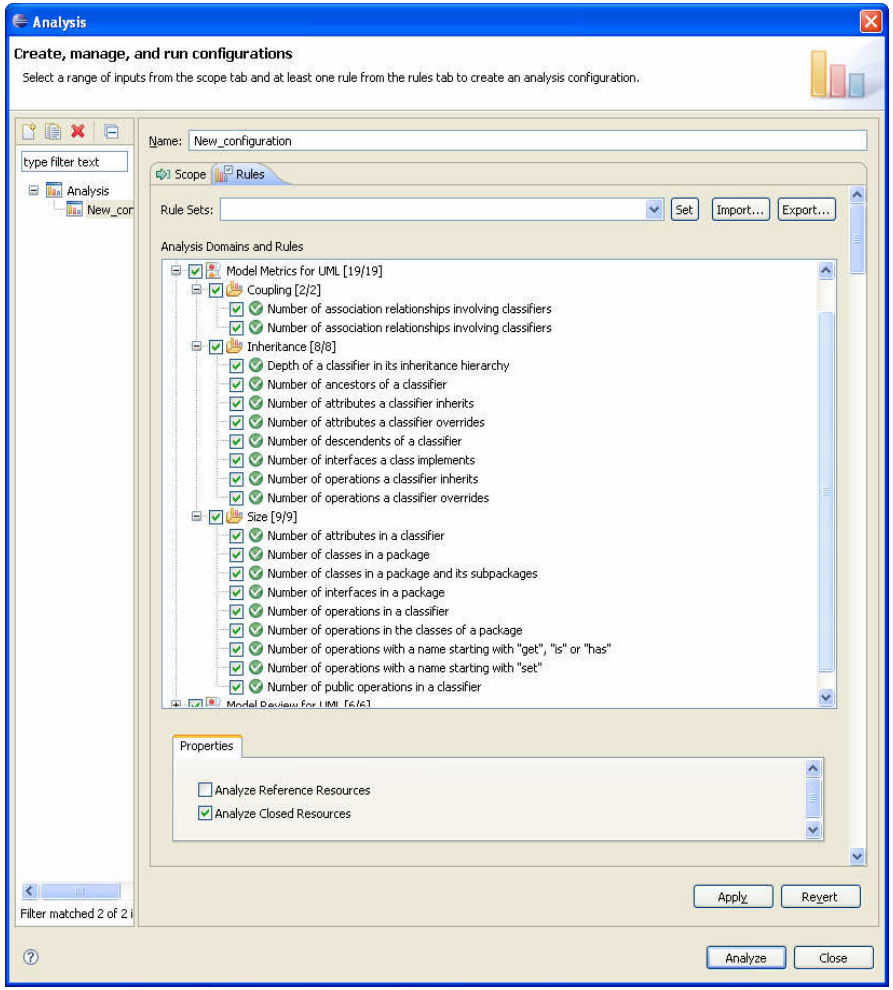
Model Analysis

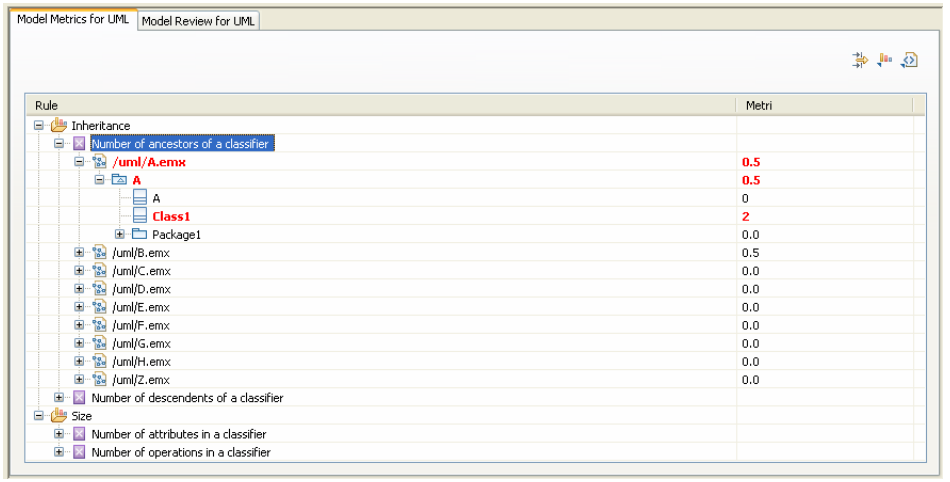


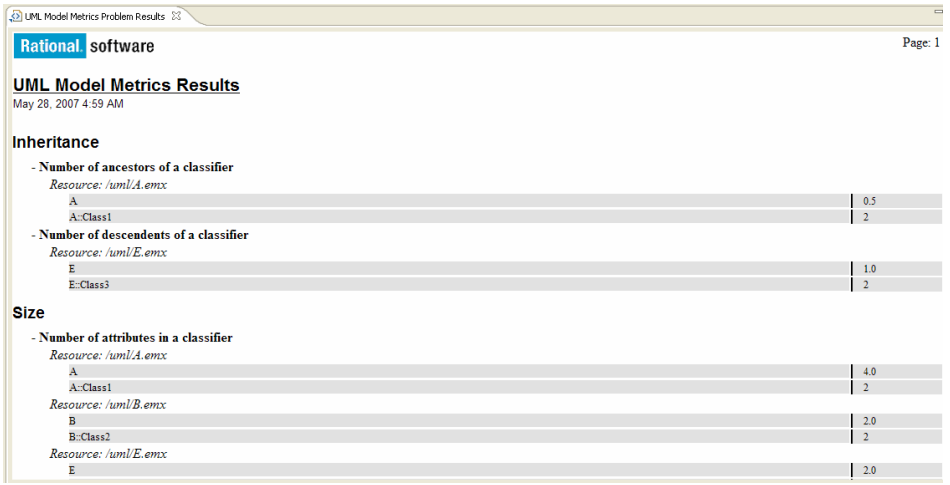




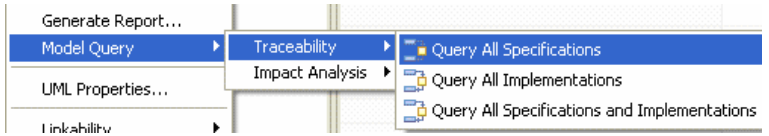
Model Metrics



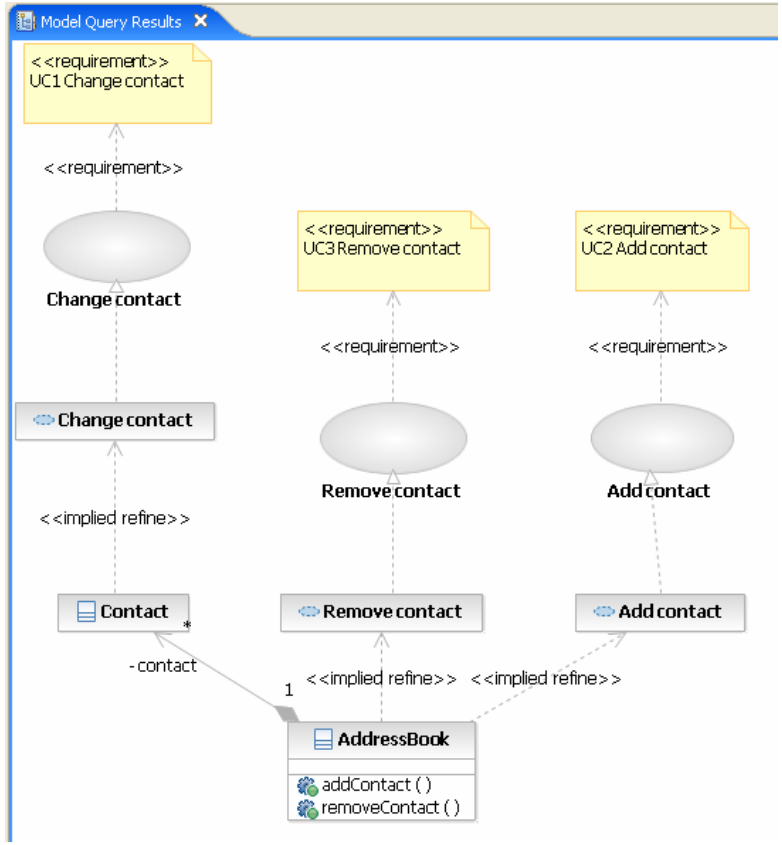




Traceability



- Traceability queries
  - All Specifications, Implementations or both
- Impact Analysis queries
  - All Causes of Change, All Effects, or both
- Results displayed in editable topic diagram
- Queries search closed resources and referenced resources
- Specifications and Implementations supported by Web Publishing



Specifications

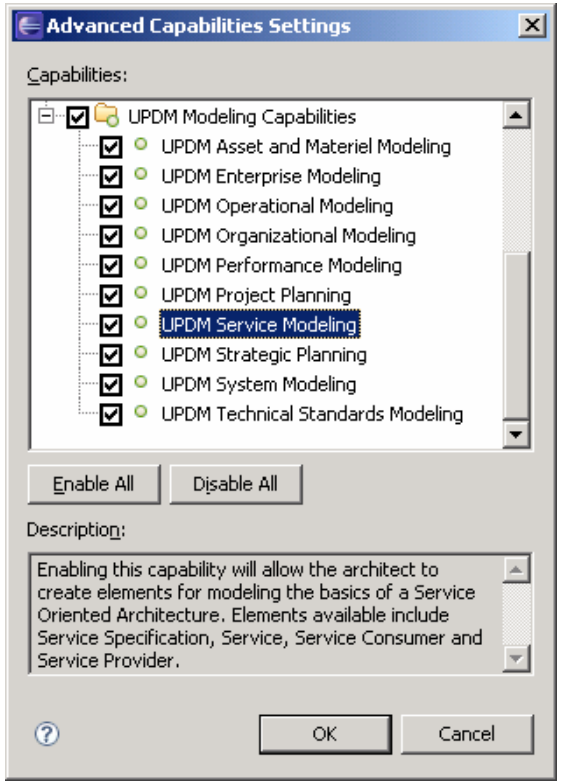
Specification	Trace Relationship
Add contact	(Add contact)

UPDM – UML Profile For DoDAF And MODAF

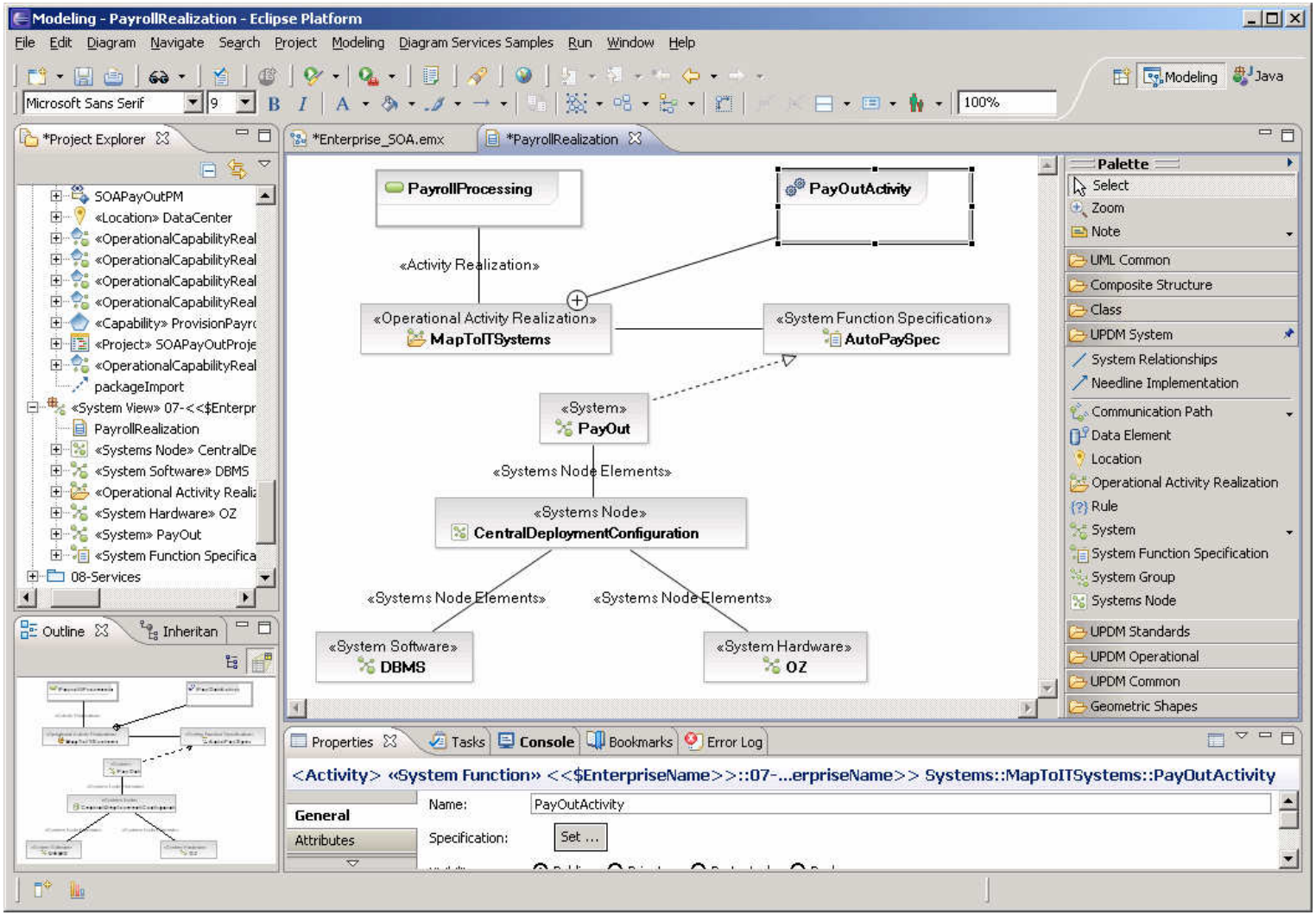
- UPDM 1.0 supports general Enterprise Architecture designs
  - UPDM is model (semantic) centric and not view centric like DoDAF
  - UPDM does not limit the views to DoDAF or MODAF
  - UPDM has hooks for Service Oriented Architecture designs
- UPDM profile is complex but has built in validation rules
  - 165 stereotypes (extending both classifier and relationship meta-classes)
  - 8 data types and classes (defined in a UPDM class library)
  - over 100 attributes (many of which define relationships with other elements)
  - 62 specific relationship based stereotypes (associations, dependencies, realizations)
  - over 260 OCL constraints

UPDM (continued)

- RSx supports UPDM basic tools for creating elements and relationships
  - no support yet for reporting or creating specialized views
- UPDM element types logically divided into 10 architectural modeling phases (capabilities)
- UI contributions organized by capabilities
  - separate diagram palette drawers
  - separate context menus
  - integrated into new RSx UI reduction framework
  - smart “relationship” creation tools that determine possible UPDM relationships which can be created between two classifiers
  - power tools perform multiple modeling steps automatically

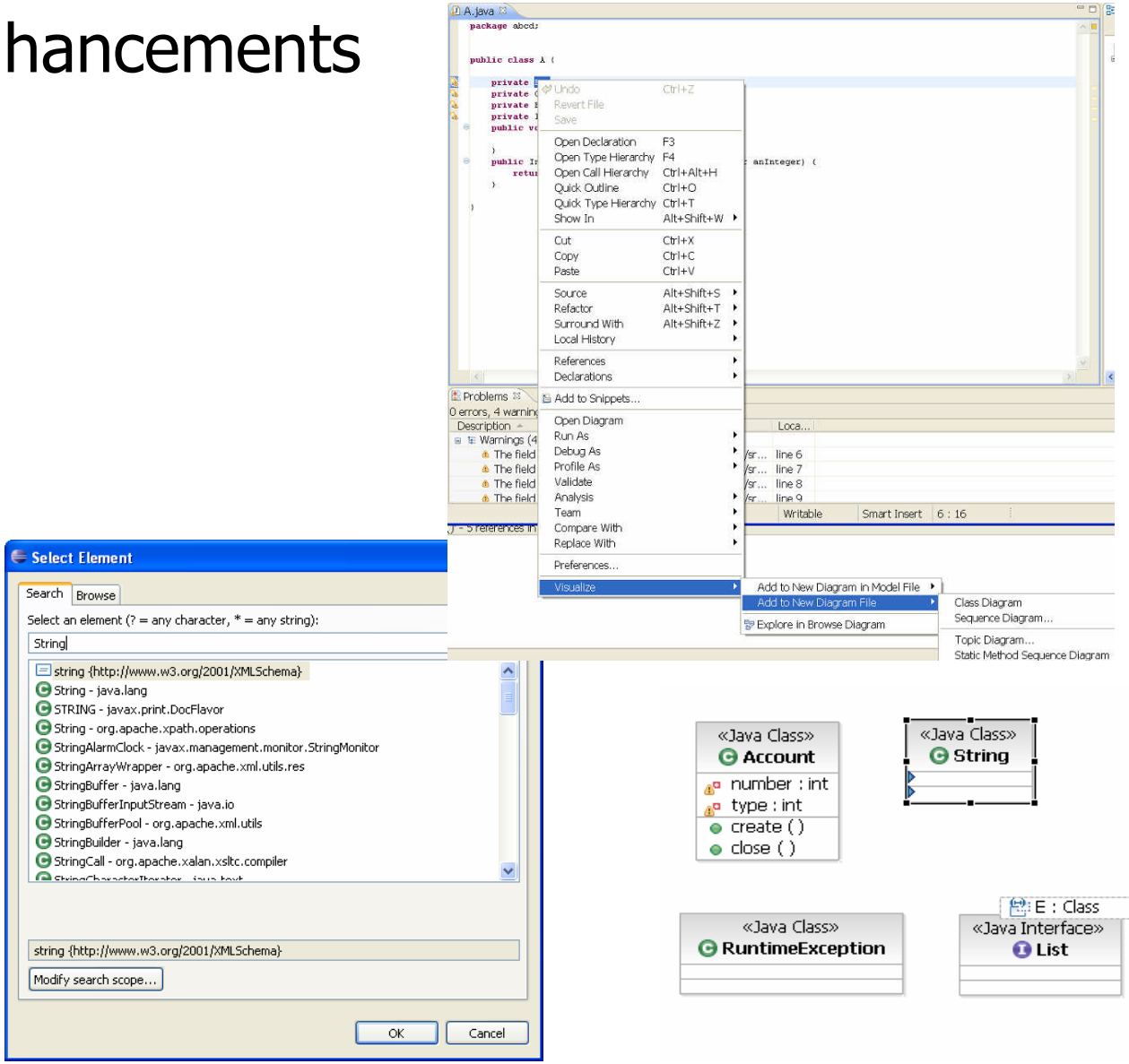


UPDM (continued)



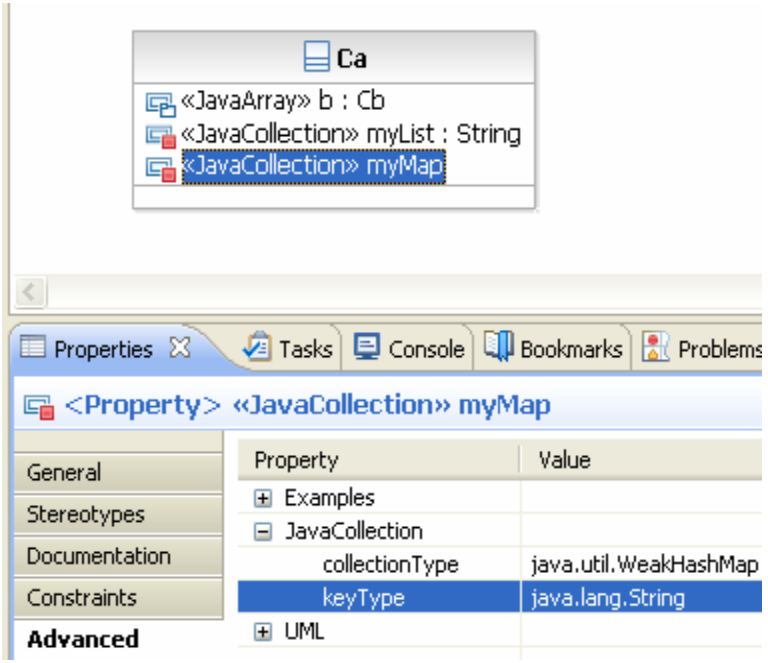
Java Modeling enhancements

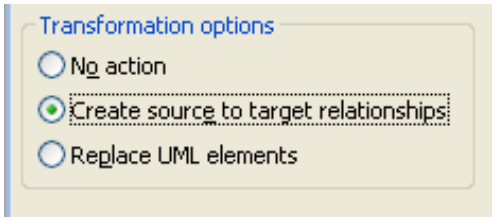
- Visualize action availability on Java source editor.
- Attribute and Operation compartments on Java Binary types collapsed by default.
- Faster update of Java search results on Select Existing element dialog.



Java Transformation enhancements

- <JavaArray> & <JavaCollection> stereotype support in Java transform
- GUID support in both forward and reverse Java transform.
- Raised exception, documentation, limited keyword support added to Java transforms.





C++ Modeling

- Support for CDT 4.0
- Ability to delete C++ Elements from diagram
- Properties view for C++ Elements
- C++ Source code documentation in Properties view

UML2C++ Transformation Enhancements

- Support STL types in the C++ transform
- GUI option for the "Exclude path for Includes" in the transformation configuration.
- CDT 4.0 migration
- Ability to specify separate output folders for generated source & headers

UML2C# Transformation

- Support for GUID based Reapply
  - Uses "source to target" relationship to map model elements to code and vice versa
  - Supports more intelligent refactoring
  - Rename or move model or code elements without losing user code
- Basic use case
  - Class1 has Operation1
  - Run UML-to-C# transform to generate code, "source-to-target" relationship **on**
  - Add code to Class1::Operation1
  - Rename Class1 to Class2 in the model
  - Reapply UML-to-C# transform
  - *Class2 will contain Operation1 with user added code preserved*

AM03

What‘s New in RSA and RSM 7.5

New Extensibility Features

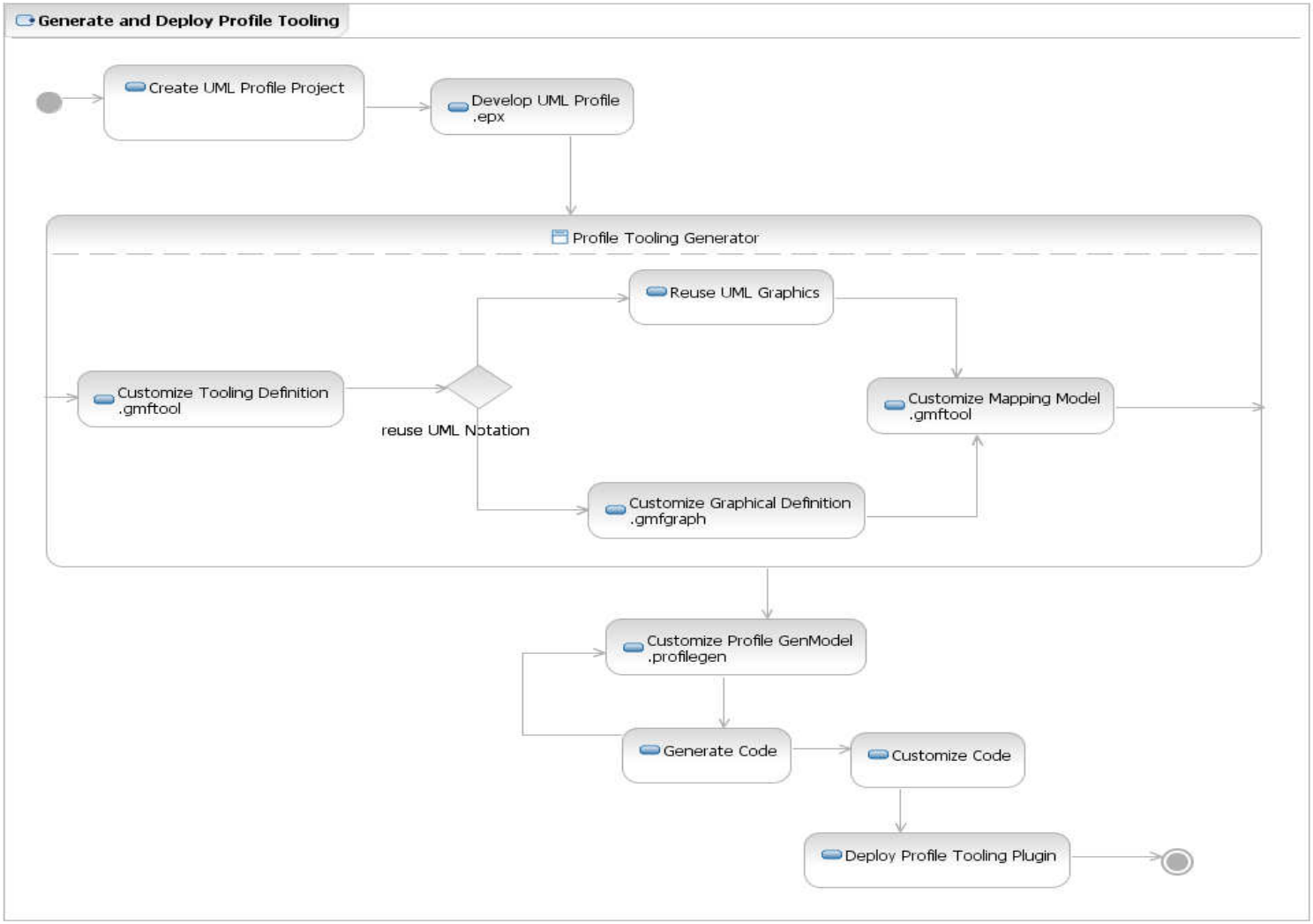
IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

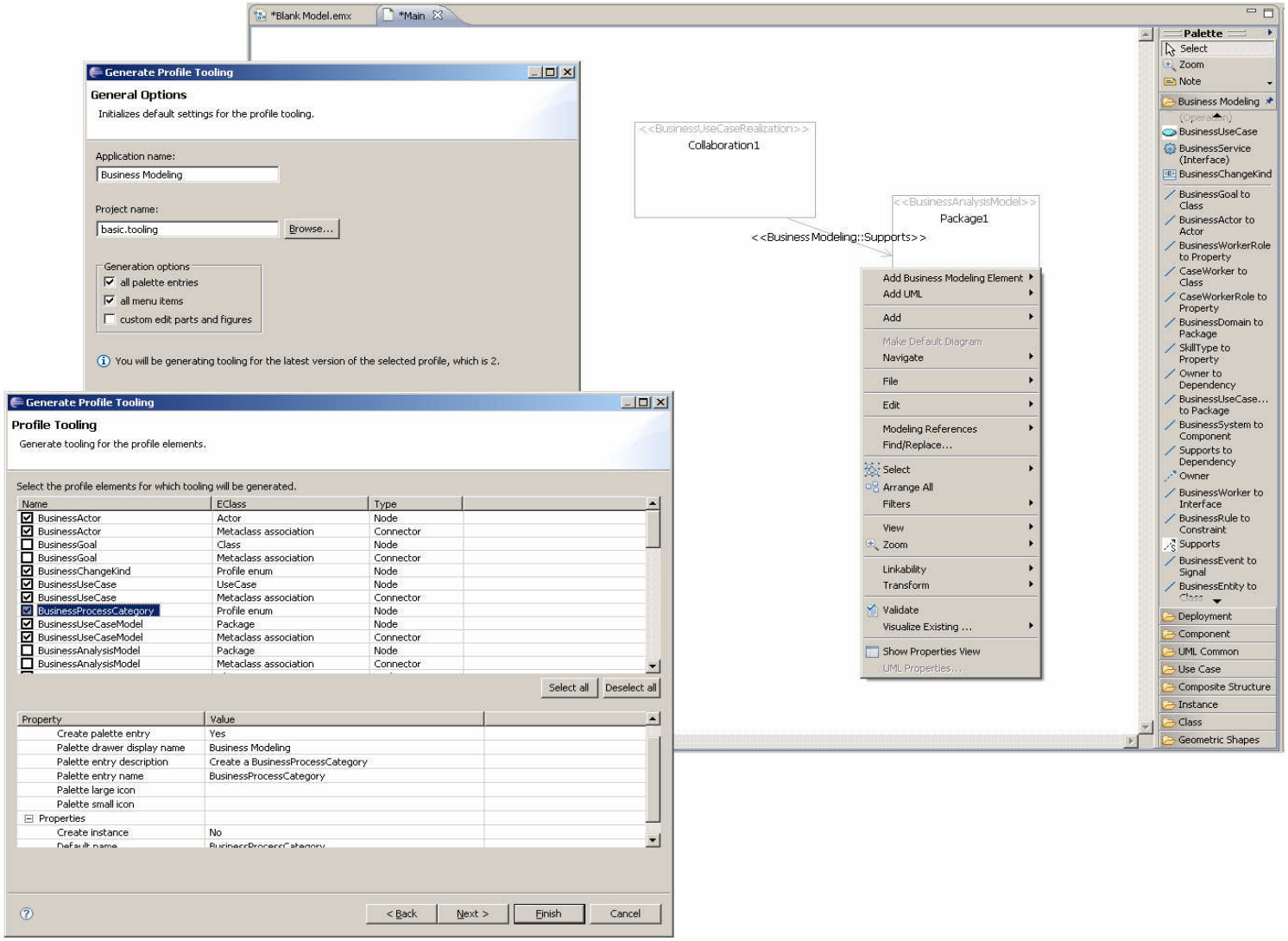
IBM

Rational. software

Generate Profile Tooling

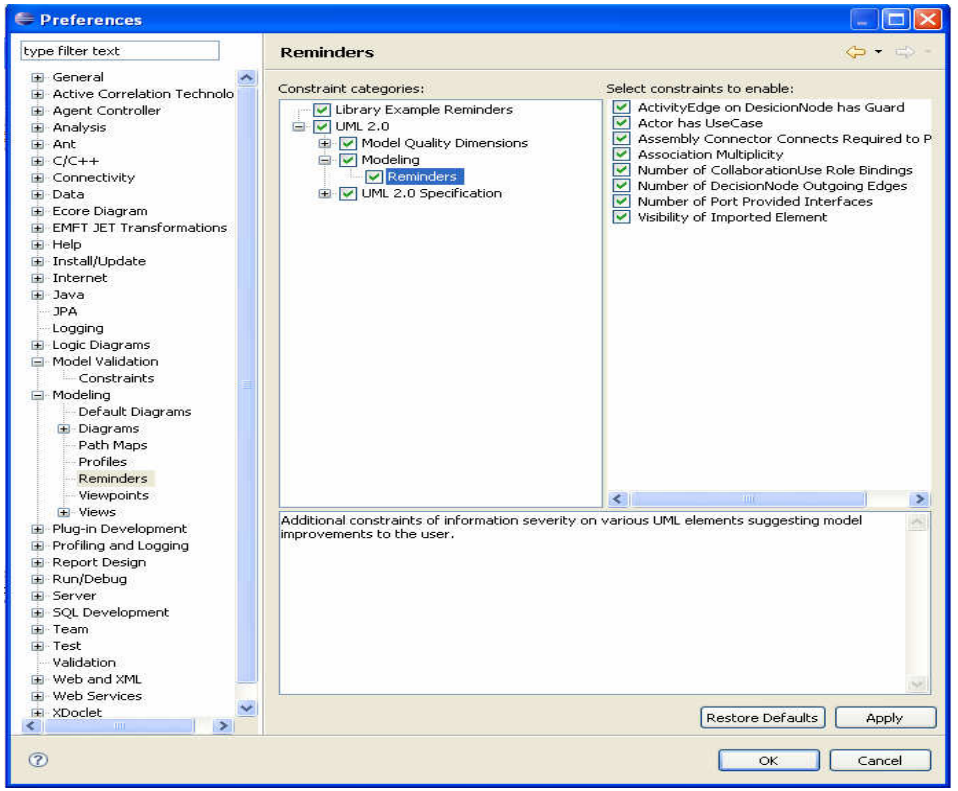


Business Modeling Profile Tooling Example



Reminders Framework

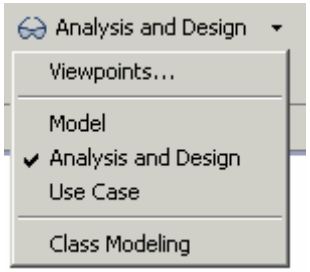
- Built on EMFT Validation Framework, and Model Indexing.
- Reminders integrate into Tasks View
- Reminders are always live, keep the user informed about the missing items (best practices, incomplete model) in the model.
- Unlike Live Validation Rules where the client has to implement lots of notification handling code, this framework provides an easy xml based registration of target based validation rules.
- Providers Filtered Reminders Customization Page.



Queries Framework

- Contribute to Explore Palette using System Queries
- Creation Executors for custom query types
- Create or reuse result presenters for custom query types

Configurable Modeling UI



- Use Eclipse capabilities for context based Reduction of Modeling tools
- Facilitates creation of Domain Specific Editors by reusing RSA Editors.
- Helps user reduce the clutter of tools on a per model basis (model specific UI configuration)
- User defined Viewpoints enables user to reduce modeling tools further. (User Specific Reduction)
- Contribute and customize Viewpoints (collection of capabilities) using view points preference page,
- Viewpoints toolbar for switching viewpoints

Model Analysis and Metrics

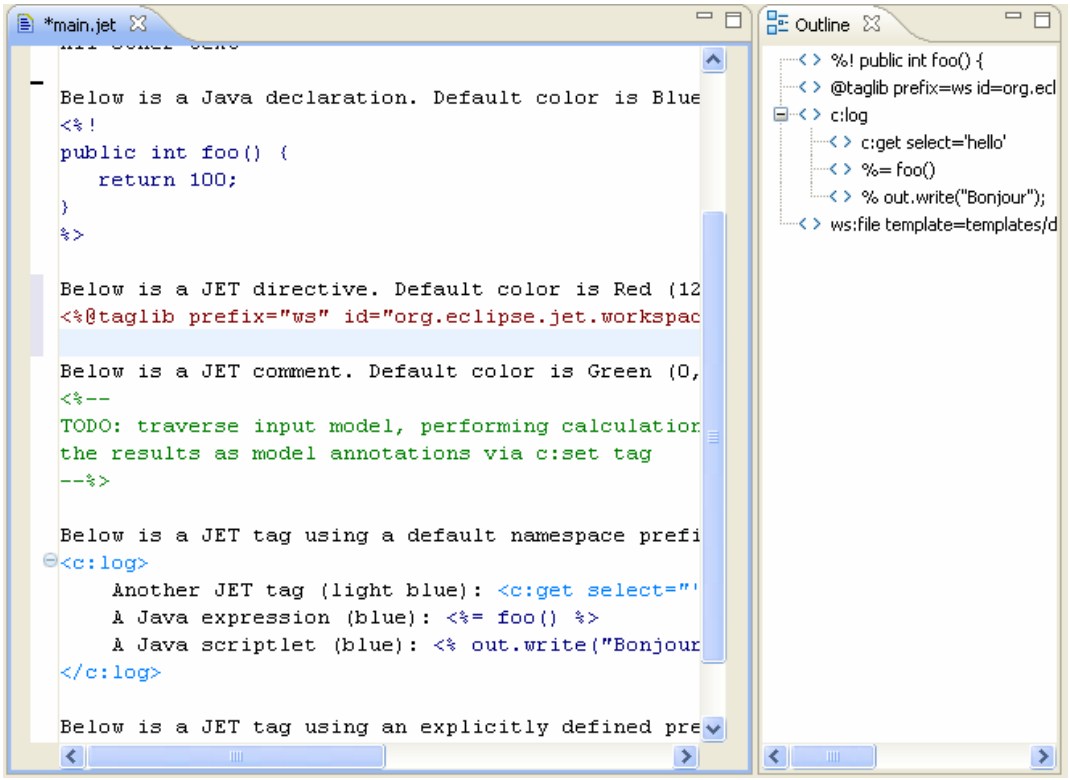
- Framework is implemented at the EMF level
- Two kind of providers are supported: rule based and metric based
- Provides abstract implementations for provider, rules, quick fixes, templates, exporters, viewers..etc
- Provide useful utilities to be used by concrete implementations
- Providers can analyze closed resources and referenced resources
- Rules and metrics have configurable parameters
- Provides quick fixes
- Supports BIRT reporting

Transformation Authoring enhancements

- Mapping Model Extensions
  - Map Inheritance – simplifies mappings and generated transform source code
  - Many-to-1 and 1-to-Many Maps – support now “built-in” (custom code not required)
  - Map Abstract Targets – generated transform code handles references to abstract types
- Mapping Editor Usability Improvements
  - Java Source Code Integration – parameter discovery and class creation wizards
  - Error Recovery – more resilient to changes in mapped metamodels
  - Map Validation and Problem Reporting – more context specific and detailed messages
- Transformation Authoring Runtime
  - Finer Grained Extensibility – less effort to customize generated transform source code
  - Simplification of Inter/Intra-Model Reference Resolution – “built-in” multi-map resolution
  - Transformation Chaining – simple to invoke another transformation

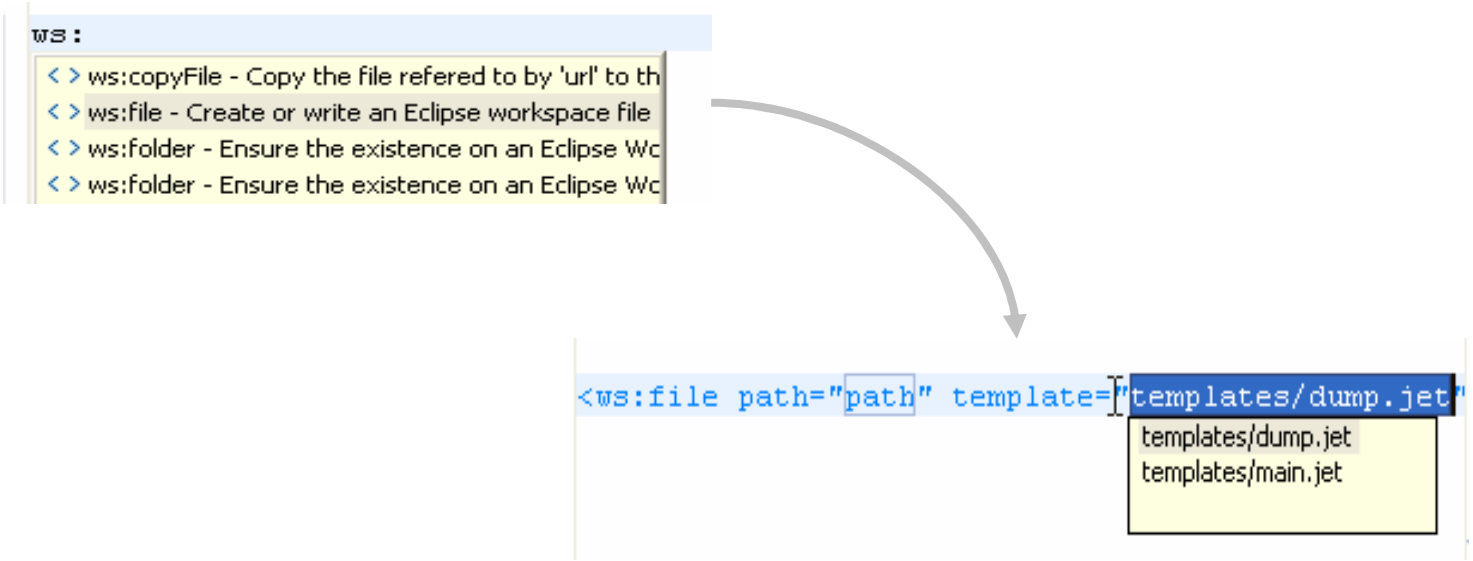
JET Template Editing

- JET now has a real editor
  - Syntax highlighting, content assist, outline view and folding



JET Template Editing

- Content Assist
  - Editor templates for all tags
  - Content Proposals for XPath and Template references

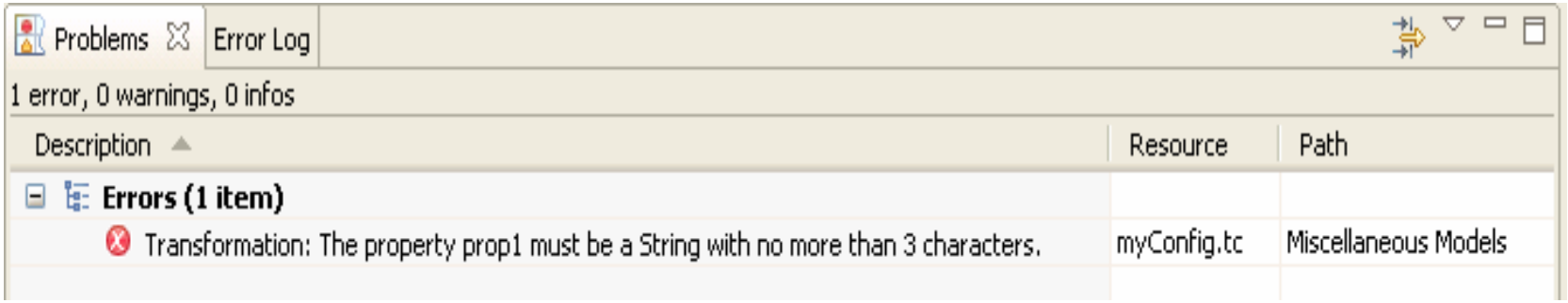


Patterns Framework Enhancements

- Patterns framework modularity is now offered providing a means for pattern authors to easily publish their own framework extensions that are seamlessly plugged in to the patterns authoring experience. Since the governing patterns framework is made on a pattern library basis each pattern author can choose the patterns framework that best suits the needs of their pattern library. It is possible to for the pattern author to create a new patterns framework that provides or changes the standard framework behavior by providing an Eclipse extension implementation to the "com.ibm.xtools.patterns.core.framework.implementation" extension point and choosing which connection points to override. The code generation process recognizes these patterns framework implementation extensions as well as their connection points to the core framework notions. A patterns preference is provided to select the default patterns framework implementation to use for new pattern libraries. This default can be overridden at pattern library creation time via the provided new pattern library PDE wizard user interface option.
- Given the ability for multiple patterns frameworks to coexist, a new specialized patterns framework is available for use by pattern authors. The historically available patterns framework is referred to as the "Generalized UML Patterns Framework" and the new framework is named the "Specialized UML Patterns Framework". The specialized patterns framework extends from the generalized patterns framework and provides two key features: pattern participant role markings (via model element keywords) and pattern participant role traceability (via UML <<trace>> abstraction relationships).
- Patterns authoring and application now recognizes and supports UML profile concepts and types. A pattern library can be associated with multiple UML profiles with the pattern author using stereotypes to further constrain the types of elements that can be bound to a parameter at pattern application time.
- The patterns explorer and patterns authoring views are enhanced to optionally display the parameter details alongside the parameter nodes in the tree view. These additional parameter details are comprised of the parameter type and parameter multiplicity. A new pattern preference is available to toggle the enablement of this new feature and by default the additional parameter details are shown.
- Pattern authoring view context-sensitive navigation is improved now supporting library node to source code navigation via either a double-click or context-menu action. Context-sensitive parameter node to source code navigation now directly jumps to the parameter class source code as opposed to the previous behavior of jumping to the parent pattern class source code.
- Pattern reference models are now preserved upon re-generation allowing user-created diagrams and other model elements to be dependably stored in these reference models.

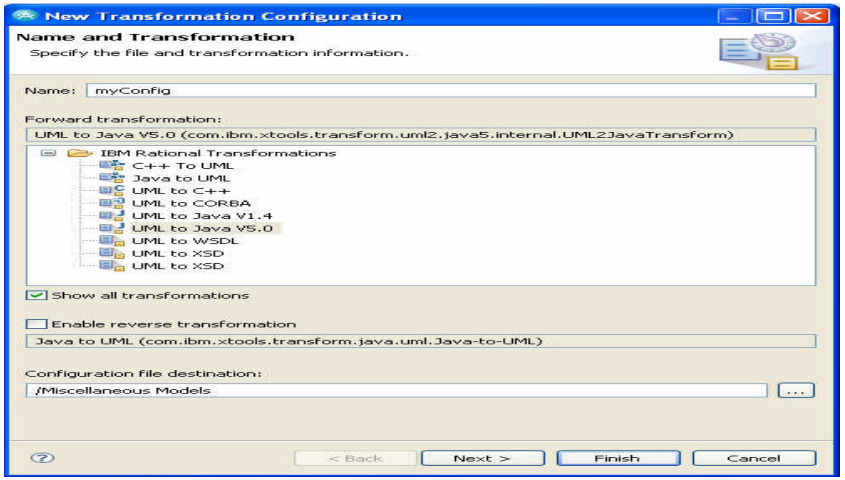
Transformation Framework

- Transformation authors can now add “constraints” to transformation properties to specify what valid property values are.
- The constraints are evaluated when the configuration is validated. Issues are reported in the problems view.



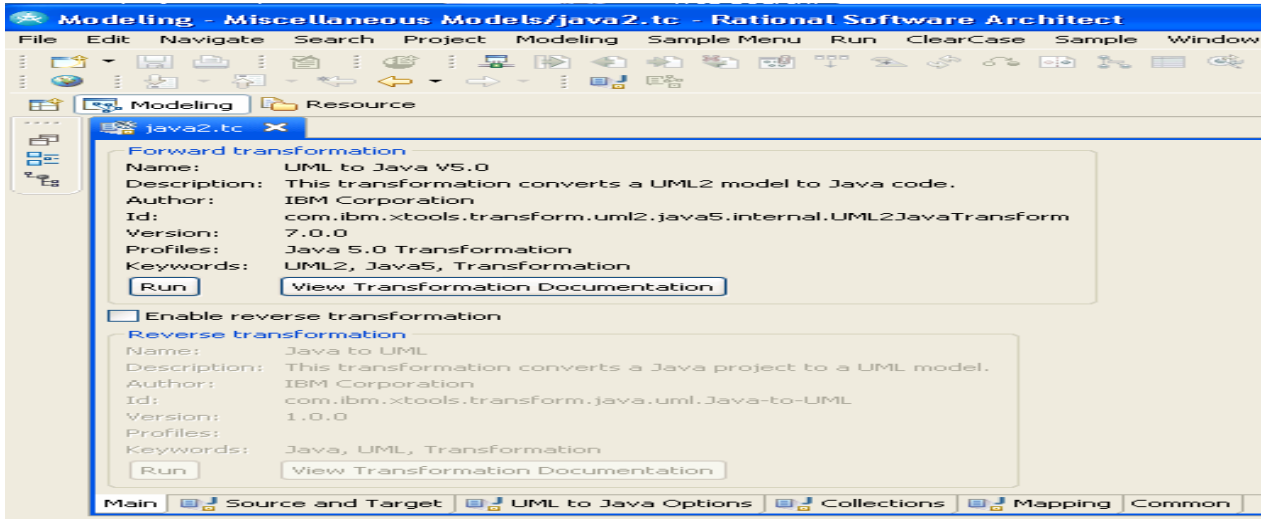
Transformation Framework

- The transformation user interface has been integrated with the Eclipse capabilities framework. Users can now filter out UI contributions at the transformation level.
- The new transformation configuration wizard by default does not show disabled transformations. The user can specify to show all transformations.
- Menu items for disabled transformations do not appear in the workbench.



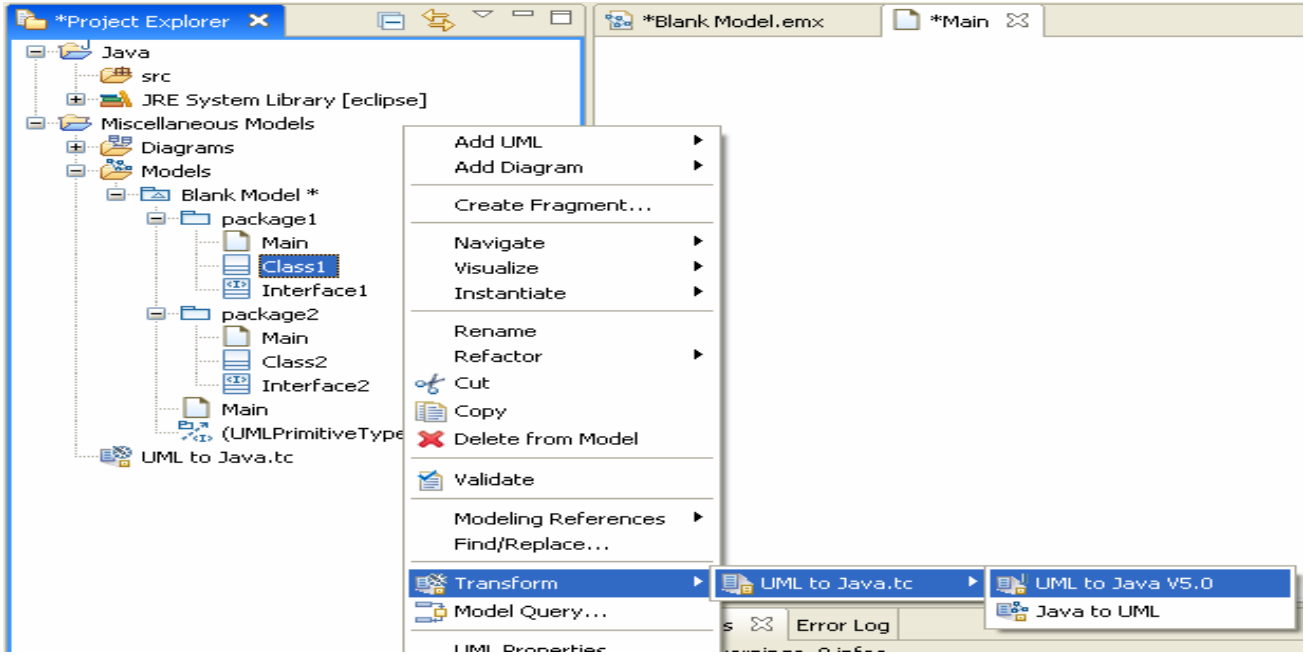
Enhanced main page of the transformation configuration editor

- The user can run the forward or reverse transformation from the main page of the transformation configuration editor.



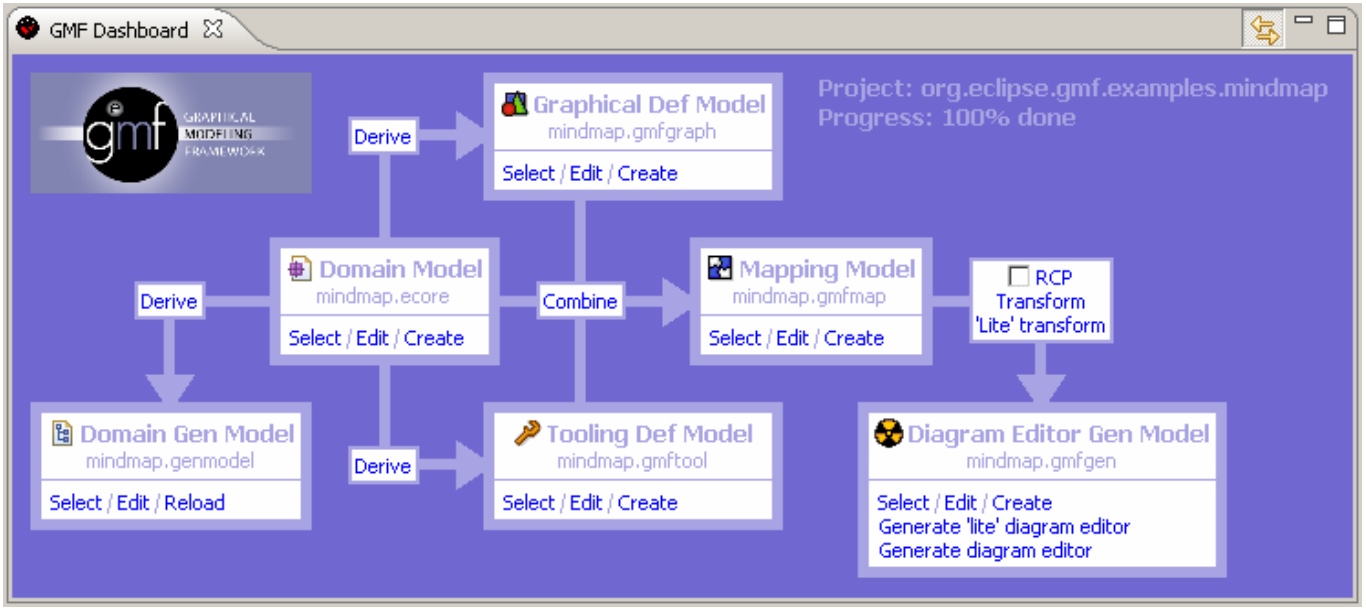
Transformation Framework

- A transform context menu has been added to the project explorer (previously only transformation configuration files had a transform context menu).



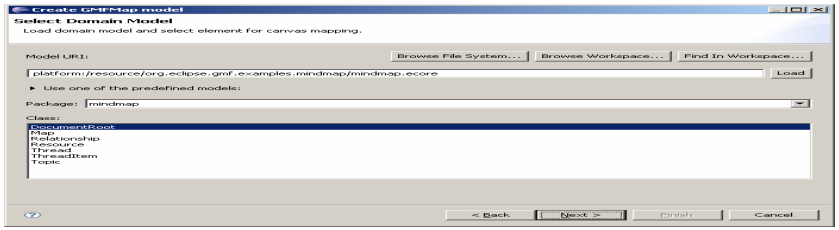
GMF 2.0 Overview

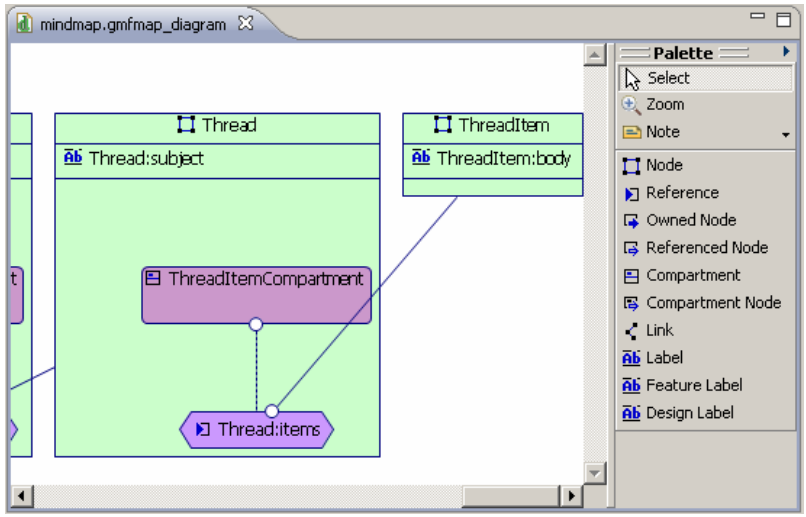
- GMF provides an MDD approach to generating graphical domain editors.
- GMF has two main components:
  - GMF Generation Toolkit
    - Models used to define graphics, tooling, mapping to domain
    - Various options to generate code for your graphical editor.
  - GMF Runtime:
    - A framework to binds EMF and GEF.
    - Provides significant diagramming capabilities out of the box.
- Diagrams in the Rational Design Products are based on the GMF Runtime.



GMF 2.0 Generation Toolkit Enhancements

- Ability to generate an Eclipse RCP application.
- Project Navigator support for diagrams and models.
- Diagram preferences generation option.
- Enhanced properties view generation support.
- xPand template engine.
- Graphical mapping model editor and graphical definition editor improvements.
- Rebuilt tooling wizards.
- Improved Mapping Model Visual Editor (built using GMF).
- Improved Graphical Definition Visual Editor (built using GMF).



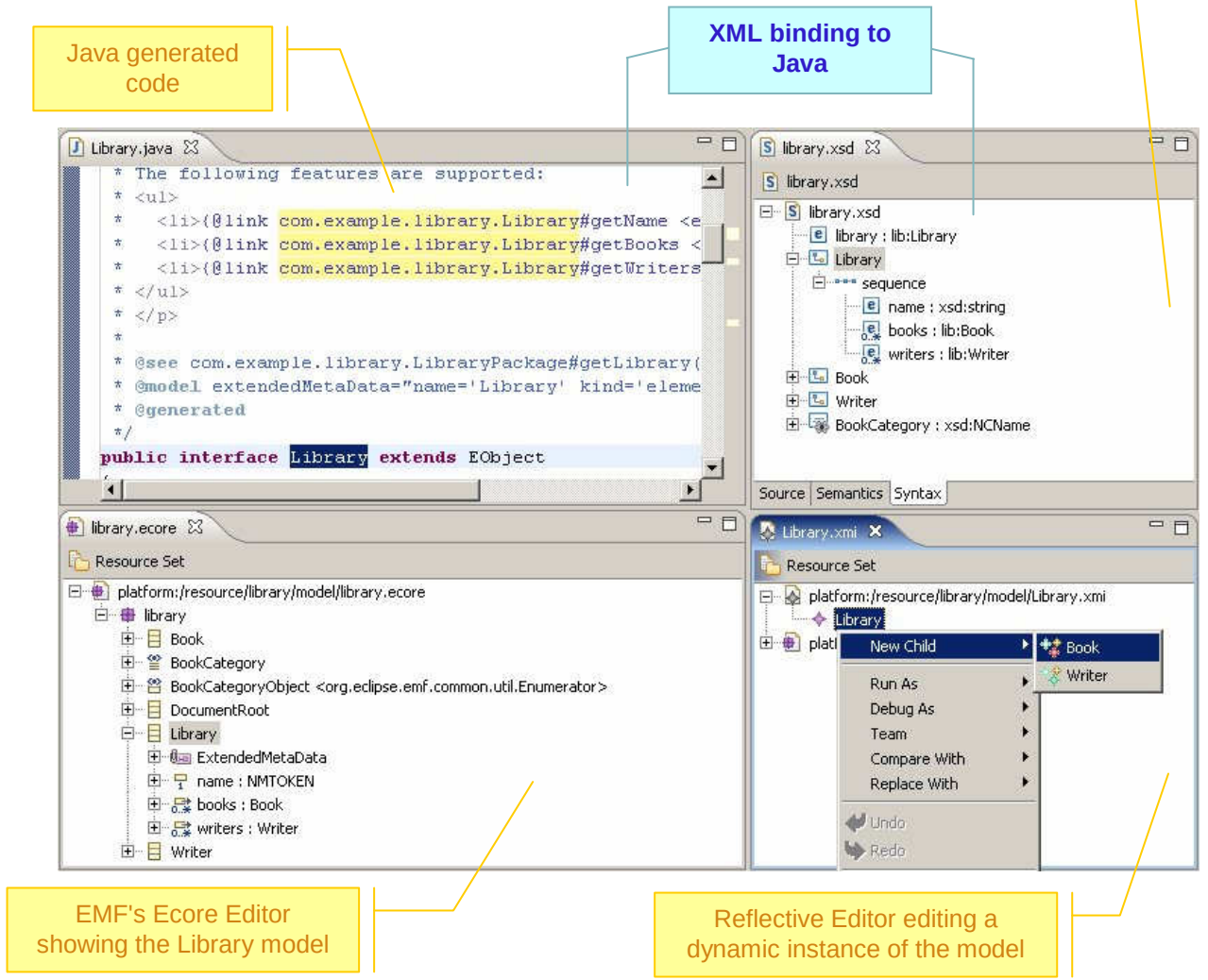


GMF 2.0 Runtime Improvements

- Printing improvements:
  - Issues with paper sizes and large diagram output resolved.
- Snap to grid improvements.
- Rectilinear router improvements for connections.
- Diagram export to image file improvements.
- Compartment scrolling improvements.
- Improvements for path maps and linked resources.
- Added improved Notational MetaModel Extensibility.
  - Ability to add generic styles.
- 342 Bugs fixed since GMF 1.0.
- No API breakages from GMF 1.0 Runtime.
- No “provisional” APIs, all non-API code in “internal” packages.

Eclipse Modeling Framework (EMF)

- Provides a uniform mechanism for describing and accessing all data to facilitate sharing
- Supports conversion to and from different models descriptions
- Developer can focus on the creative tasks rather than repeating the mundane tasks again and again
- The merging generator supports alternating between modeling and programming
- Full support for Java 5



Eclipse Modeling Framework (EMF)

- EMF 2.3 News and Noteworthy
  - Java 5.0 support
    - Allows developers to exploit Java 5 concepts and features in their models, since Generics are now fully supported in EMF's metamodel, Ecore, and in the EMF code generator
    - Other important Java 5 language features are also supported, including enumerated types, annotations, and the enhanced for loop
    - Developers still have the option to generate 1.4-compatible code, which can be run on the EMF 2.2 runtime
  - Complete Ecore model validation
  - Reference keys, which provide a more robust alternative to the default index-based reference serialization
  - EMF 2.3 will be delivered in fine-grained features, allowing for more flexible use in projects and products, without introducing unwanted code, user interface, or dependencies

UML2 2.1

- Eclipse 3.3 / EMF 2.3 Compatibility
  - Many methods within the UML API are now specified using Java 5.0 generics and are consistent with EMF's changes for Java 5.0. This could have potential impact on users calling the UML API, for example when working with lists of a specific meta-type.
  - Users can convert UML models with templates to Ecore models with generics and vice versa. UML concepts such as templates, template bindings, etc., have a direct impact on the resulting Ecore model and ultimately the generated code. Likewise, Ecore models with EGenericTypes or ETypeParameters (new meta-types in the Ecore metamodel) can be converted to UML models that make use of the corresponding UML concepts.
- Ant Task for Ecore Importer
  - A new Ant task has been created that allows users to generate code based on a UML model. From the Ant task, users can provide UML-specific code generation options for the generator model, generator package, and converter, in addition to passing an input UML model. The user is then one click away from running the Ant task and generating code directly from a UML model.

UML2 2.1 (Continued)

- Create Child/Sibling Menu Reorganization
  - The 'Create Child' and 'Create Sibling' menu items in the UML Editor menu have been reorganized so that sub-items are grouped by feature.
- Static Profile Definition
  - Users now have the option to convert their profiles to Ecore just as they would any other model. They can then generate code from that model and provide implementations for operations and derived features. If the model is then registered in the generated package registry, the UML API will discover the Ecore definition instead of looking for it in an annotation; in such a scenario, the "define" step on the profile is not needed - the conversion to Ecore replaces it.
- New API to Create Usage Dependencies
  - An additional convenience method has been introduced to create usage dependencies on named elements. Although the ability to create usage dependencies existed in the past, this new convenience method makes it even easier.

UML2 2.1 (Continued)

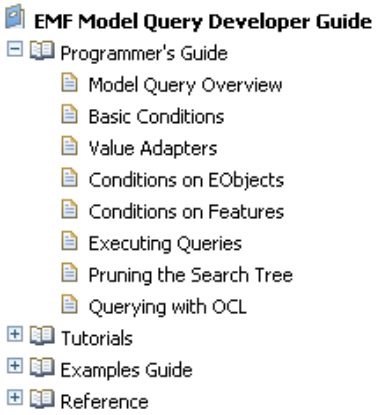
- OCL Integration
  - Now users can specify invariant constraints or operation bodies in Java or OCL and have code generated from the expressions entered in the UML model.
- Updated Schema
  - The namespace URI for the UML metamodel in UML2 2.1 has been changed to (http://www.eclpse.org/uml2/2.1.0/UML). IMPORTANT: The source for annotations such as those used to store the dynamic Ecore representation of defined profiles still use the original annotation (http://www.eclpse.org/uml2/2.0.0/UML).
  - This change was required to reflect changes to the metamodel to remove the bidirectional associations between Vertex::incoming/outgoing and Transition::source/target and between ConnectableElement::end and ConnectorEnd::role.
- Export Wizard Improvements
  - The UML export wizard used to output new resources for every nested Ecore package in a model. This no longer is the case. Also, if the ecore model that is being converted contains references to types in other metamodels, the corresponding UML representation of that referenced metamodel will also be generated.

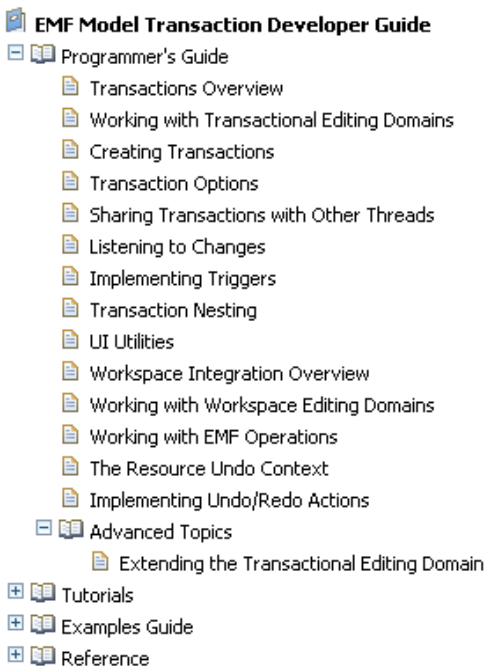
EMF Query, Transaction, Validation 1.1

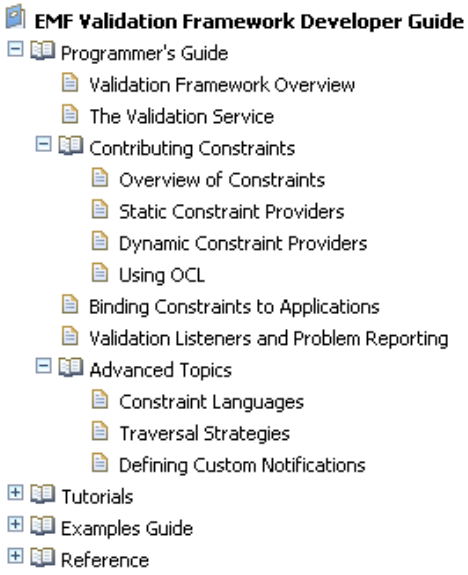
- Validation Enhancements:
  - Constraint Filtering
    - Allows clients to opt out of selected constraints
  - Constraint Listeners
    - Clients are notified of changes in the ConstraintRegistry, including constraint enablement changes, constraint registration changes, constraint category changes
  - Extensible Constraints Preference Page
    - The preference page is refactored and provides new API for clients to create their own preference pages
  - Custom Validation Event Types
    - Simplifies the triggering of “live” validation constraints by computing more abstract events according to rules provided by clients
  - Constraint Violation Reporting
    - Constraints have more control over the reporting of problems, including which elements are involved and even reporting multiple discrete results as a multi-status
  - OCL Integration
    - The support for OCL as a constraint specification language is updated for OCL 1.1 to support Ecore and UML metamodels
    - New API for applications to create OCL constraints in customized OCL environments

EMF Query, Transaction, Validation 1.1

- Better support of EMF resources lifecycle, including disconnecting them from the transactional editing domain
- Complete Programmer's Guides to the Query, Transaction, and Validation APIs







AM03

What‘s New in RSA and RSM 7.5

Deployment Architecture Platform

IBM Rational Software Development Conference 2008

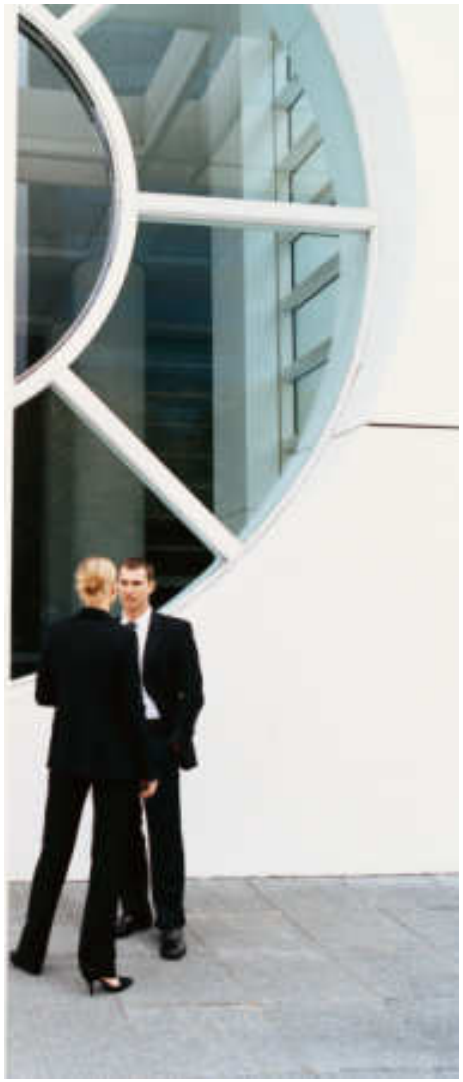
WHERE TEAMS ARE R-HEROES

IBM

Rational. software

Outline

- Goals and Vision
- Key Model Concepts
- Platform Values and Functionality
- Product Integrations



How do you capture IT architectures?

- Are diagrams your primary documentation mechanism?
  - Microsoft Visio?
  - PowerPoint?
  - Whiteboards?
- Do you struggle to keep diagrams current?
- Do you find information is not always consistently represented across diagrams?
- Do you often see constructs within diagrams being misinterpreted?
- Do you have trouble defining recommended or approved IT stacks that provide a given set of services which can be reused in solutions?

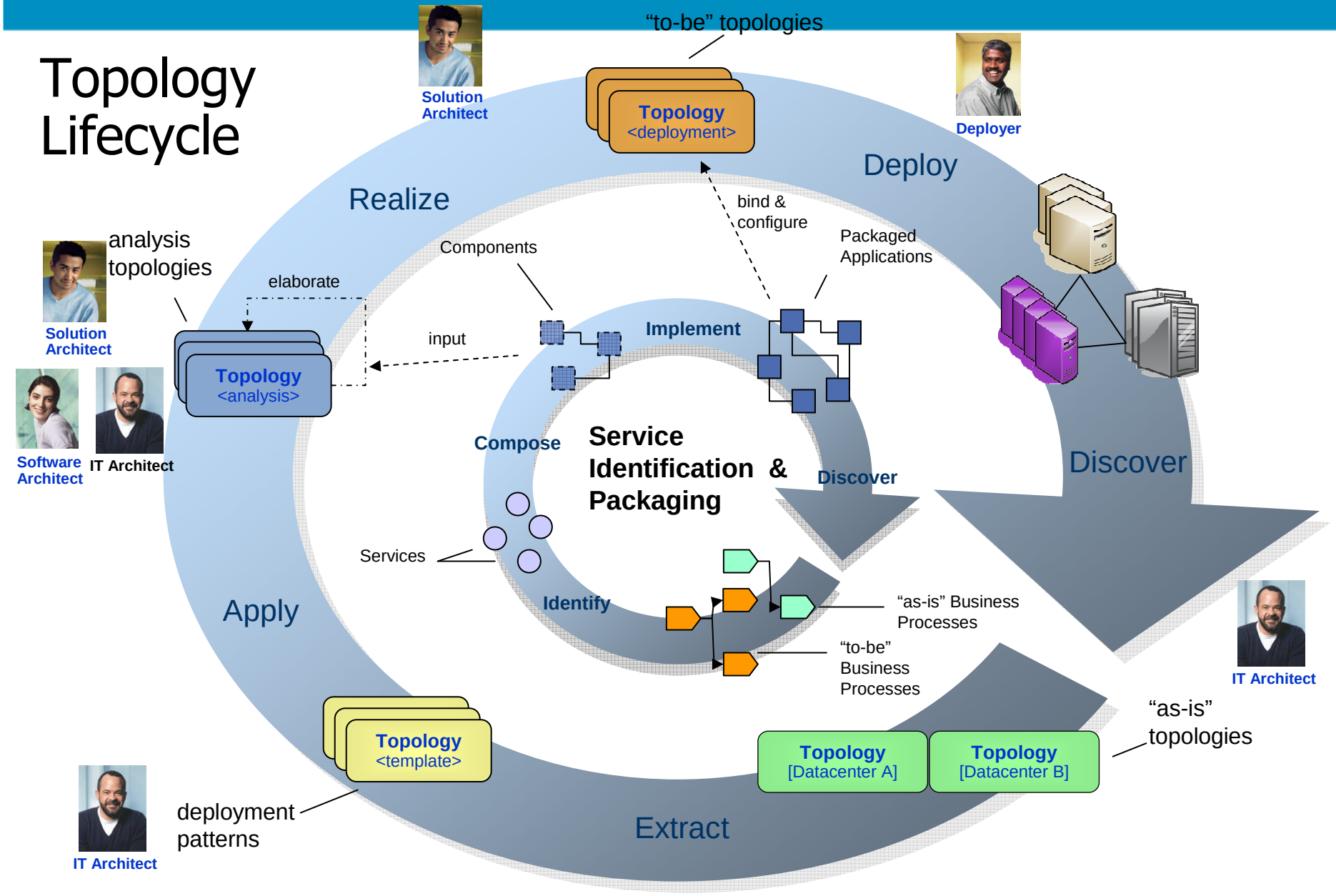
Goals and Vision

What is the deployment architecture platform trying to achieve?

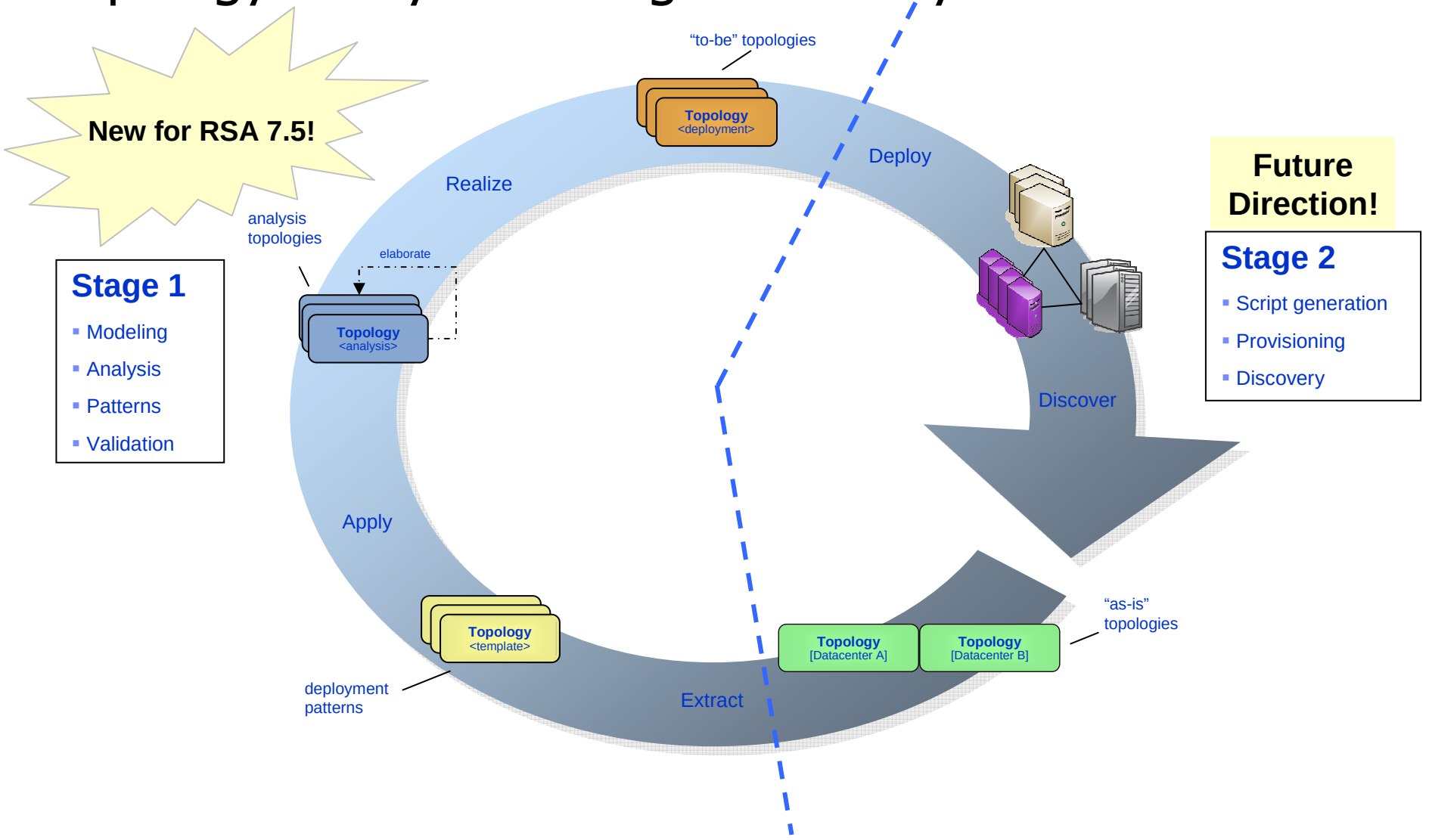
Goal of Rational Deployment Architecture Platform

- Define a common language and tooling platform to allow the creation of semantically rich IT deployment architectures.
- Document business and IT deployment needs in a way that can be validated and executed.
- Define parameterized IT topologies to improve reuse and reduce duplication.
- Provide first-class integration with the Rational modeling platform and other complementing Rational products.

Topology Lifecycle



Topology Lifecycle – Staged Delivery



Key Model Concepts

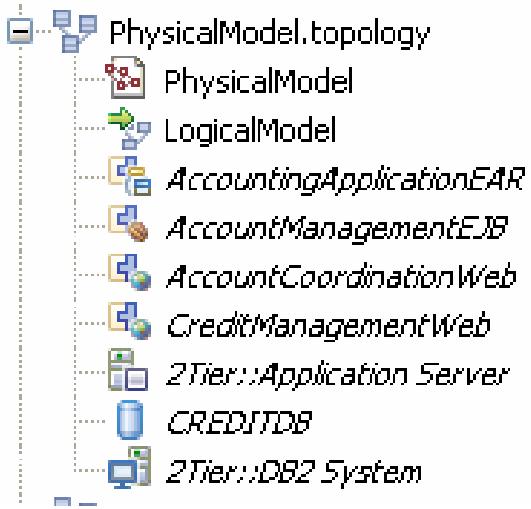
The deployment architecture platform defines a language for identifying capabilities and requirements and their linkages.

Common Language: Simple base concepts

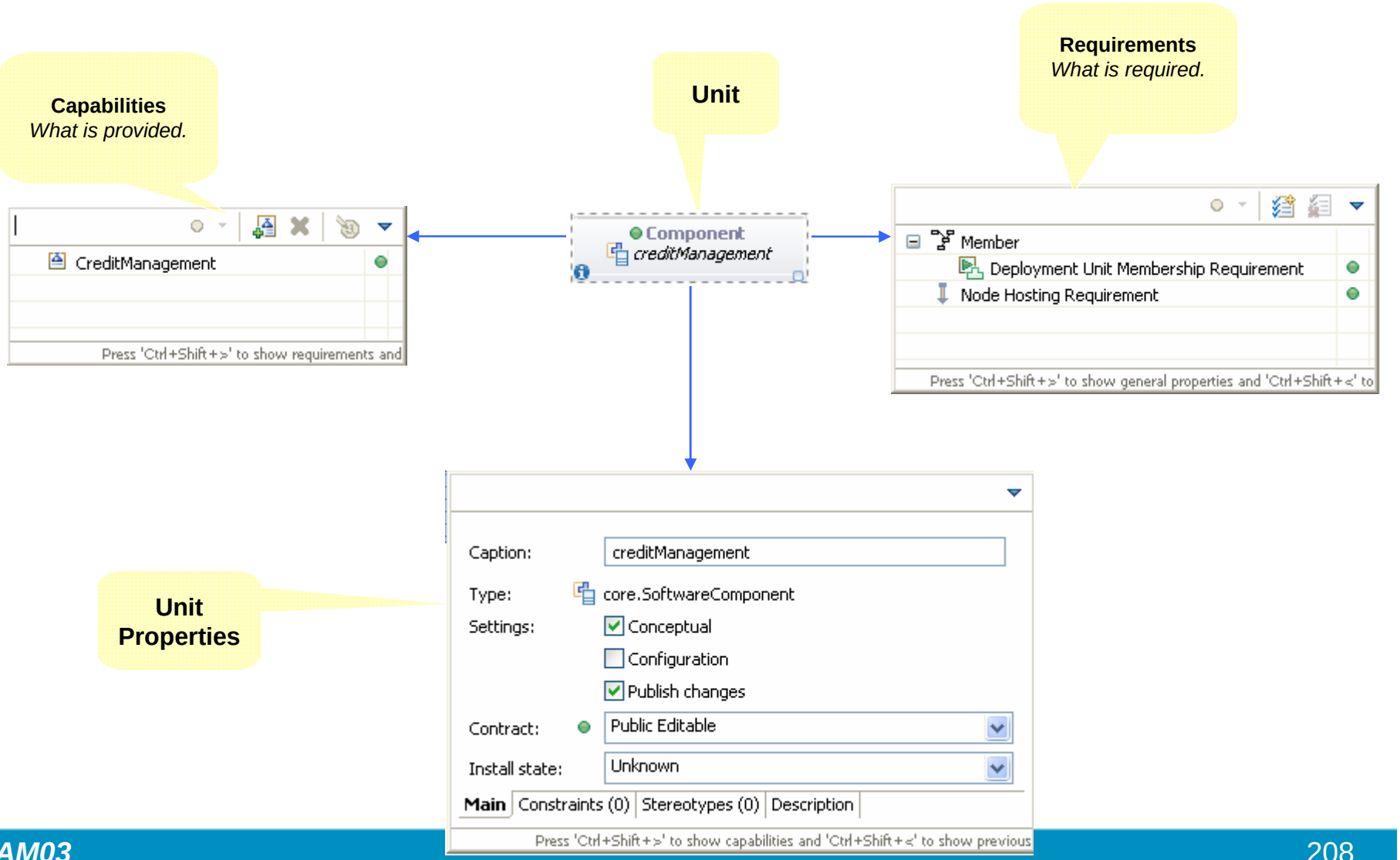
- Topology is the resource container.
- Unit defines containership, and has 0..*:
  - Capabilities which expose consumable function
  - Requirements which consume function of other Units
  - Constraints which capture and enforce architectural decisions
  - Artifacts associate on disk or external resources associated with the Unit
- Relationships
  - A Unit may be hosted on another Unit to indicate executes and installs on semantics
  - A Unit may be realized to another Unit to constrain a more detailed version of the original Unit
  - A Unit may depend on another Unit to consume a peer's function
  - A Unit may be a member of another Unit to express grouping semantics
  - A Unit may be constrained by a relationship to another Unit

Topology

- Container for Units and Links
- May import other Topologies for reuse
  - Stores overridden attribute values
- Controls visibility through contract settings

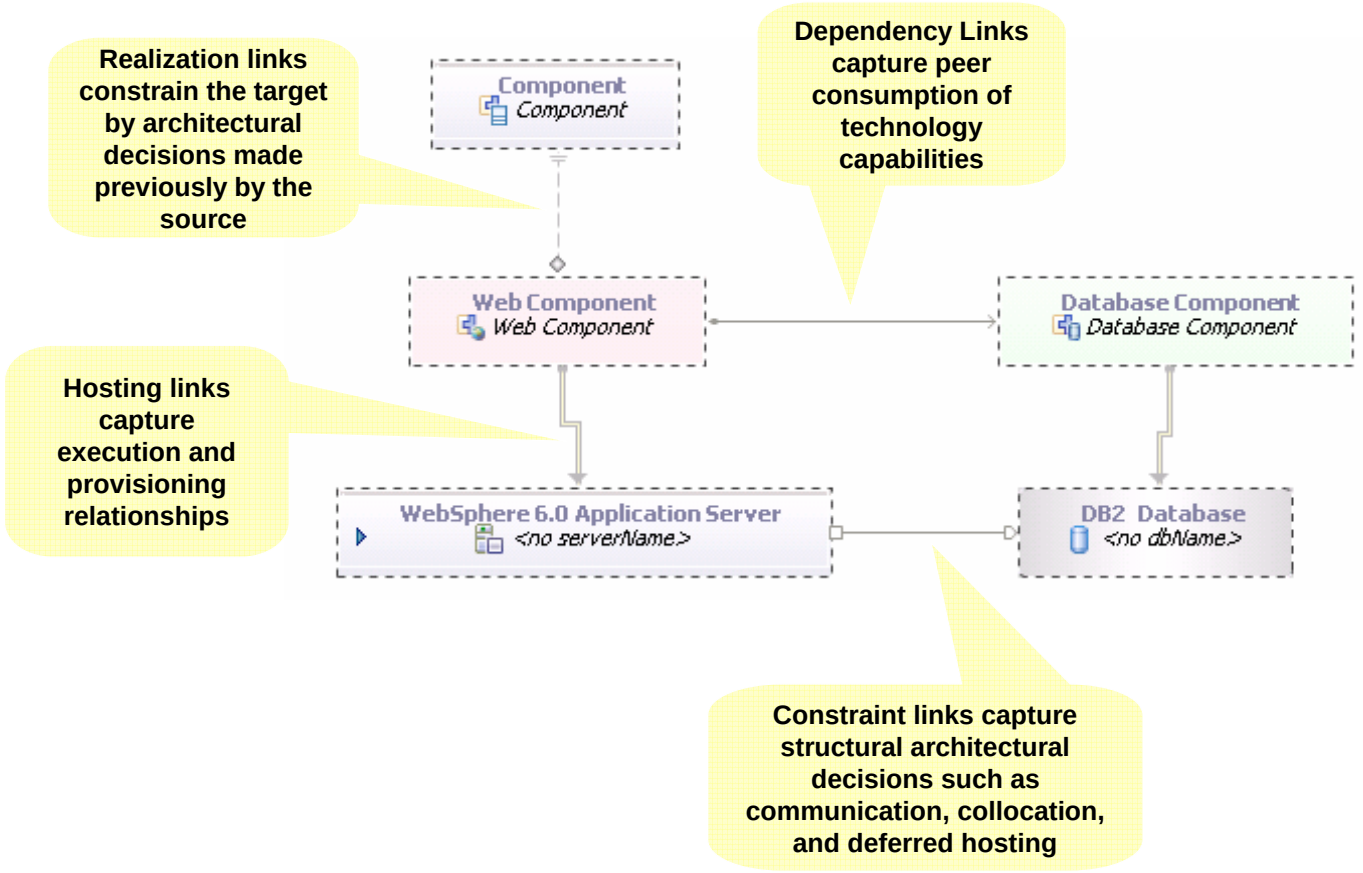


Common Language: Unit is the common base type



Common Language: Links

- Requirements are fulfilled by capabilities
- Fulfillment is accomplished via links
- Link Types
  - Dependency
  - Hosting
  - Membership
  - Realization
  - Constraint



Platform Values and Functionality

The deployment architecture platform provides valuable functionality for deployment modeling and analysis.

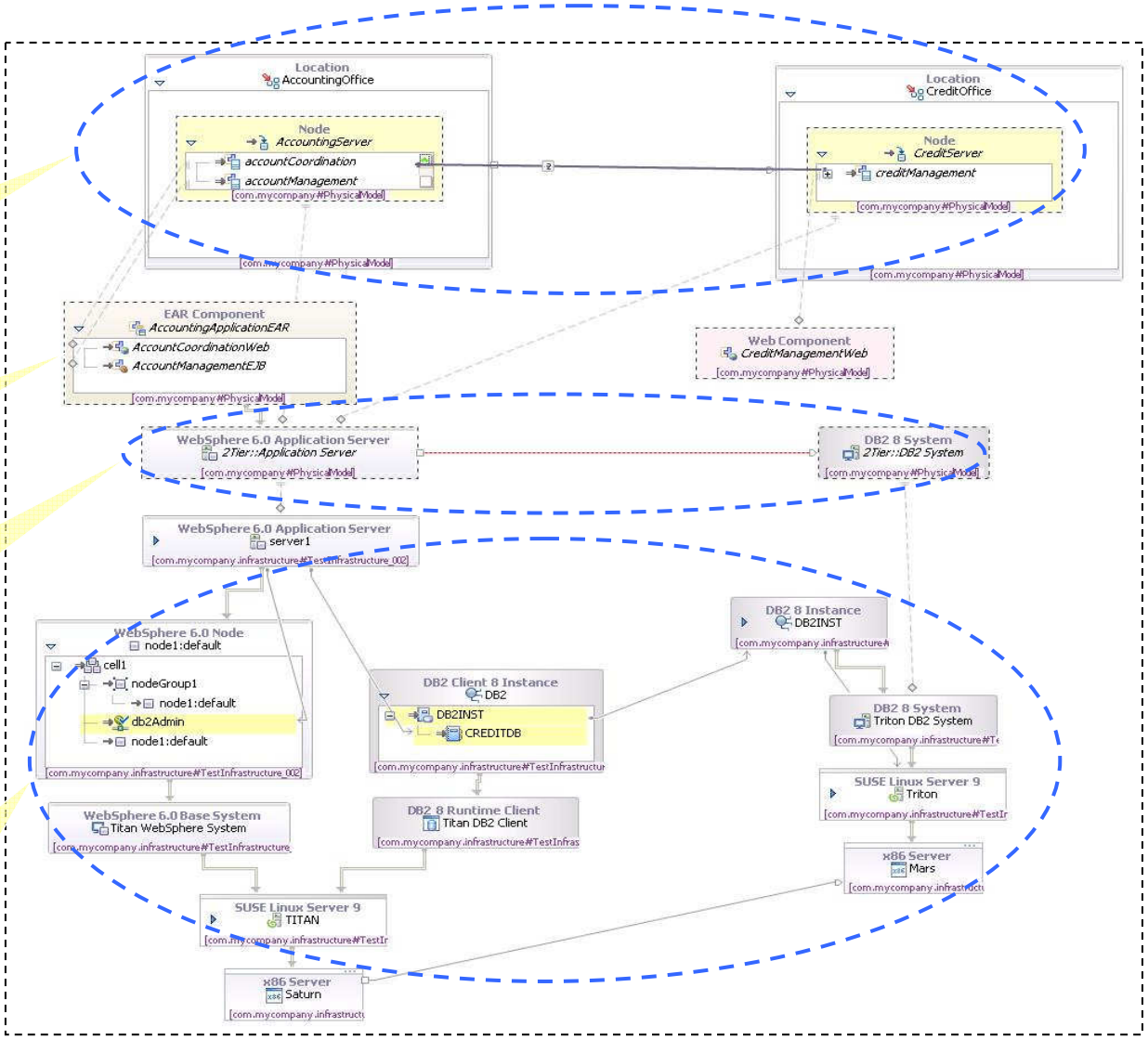
Deployment Architecture Platform

Analyze Logical Operational Model with Locations, Nodes, Components, and Actors

Capture relationships of Logical model to specific Technology artifacts (Physical Model)

Constrain final deployment environment to a best practice pattern

Map Physical Model to concrete IT Systems for Test or Production environments using type specific model for technologies like WebSphere Application Server and DB2

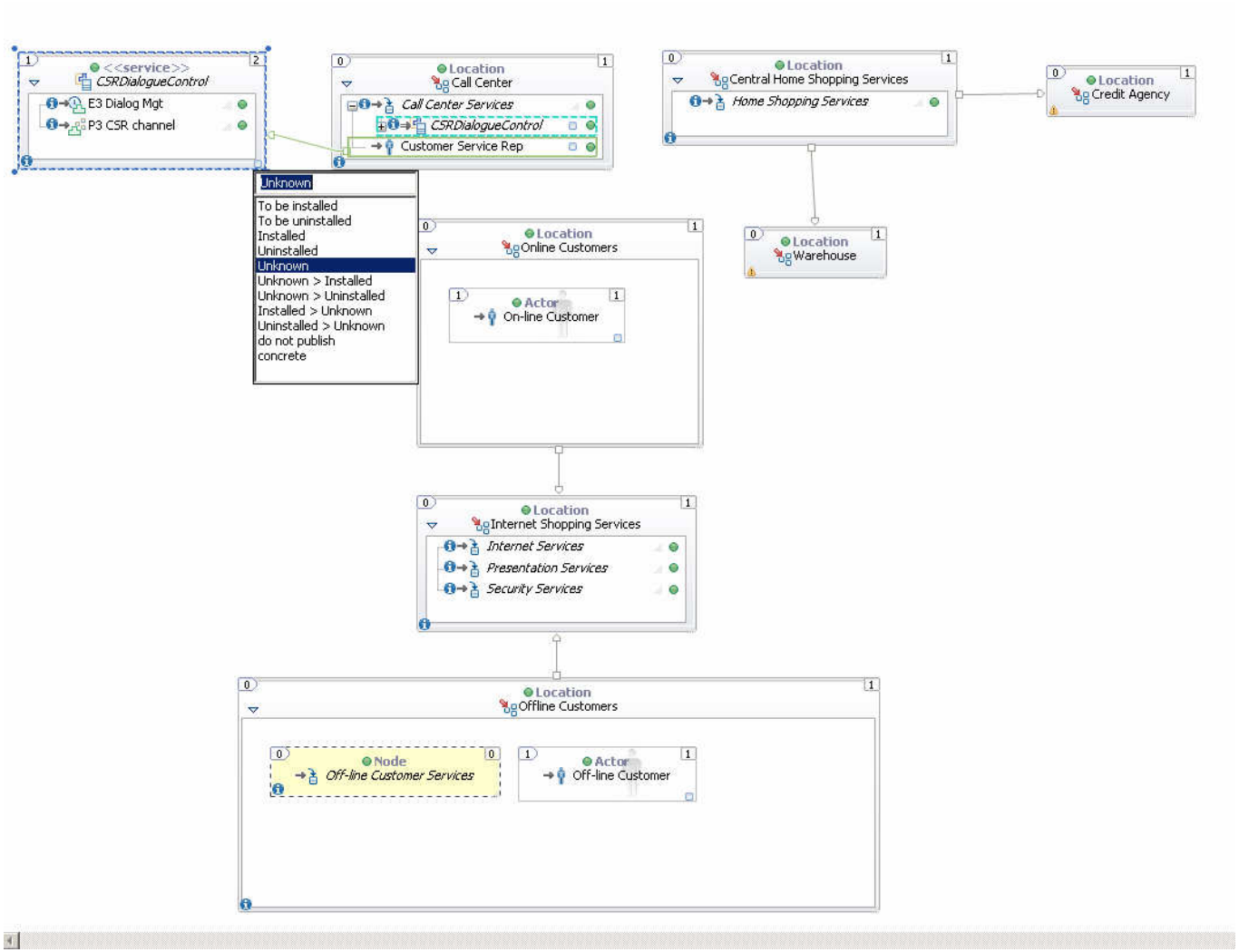


Deployment Architecture Platform Highlights

- Common language for communicating requirements and capabilities between IS and IT organizations
- Rich technology domains for an improved modeling experience
- Constraints and validations to ensure architectural decisions and requirements are fulfilled (we can show how validations are shown in the diagram with resolutions).
- Diagramming support (multiple diagrams and flexible diagramming options)
- Topology visibility settings to control topology reuse
- Support for topology imports to allow governance and reuse
- Capture standard hosting stacks and configurations and deployment patterns using Topology Templates
- Integration with other Key Products
  - Rational Asset Manager
  - Rational ReqPro

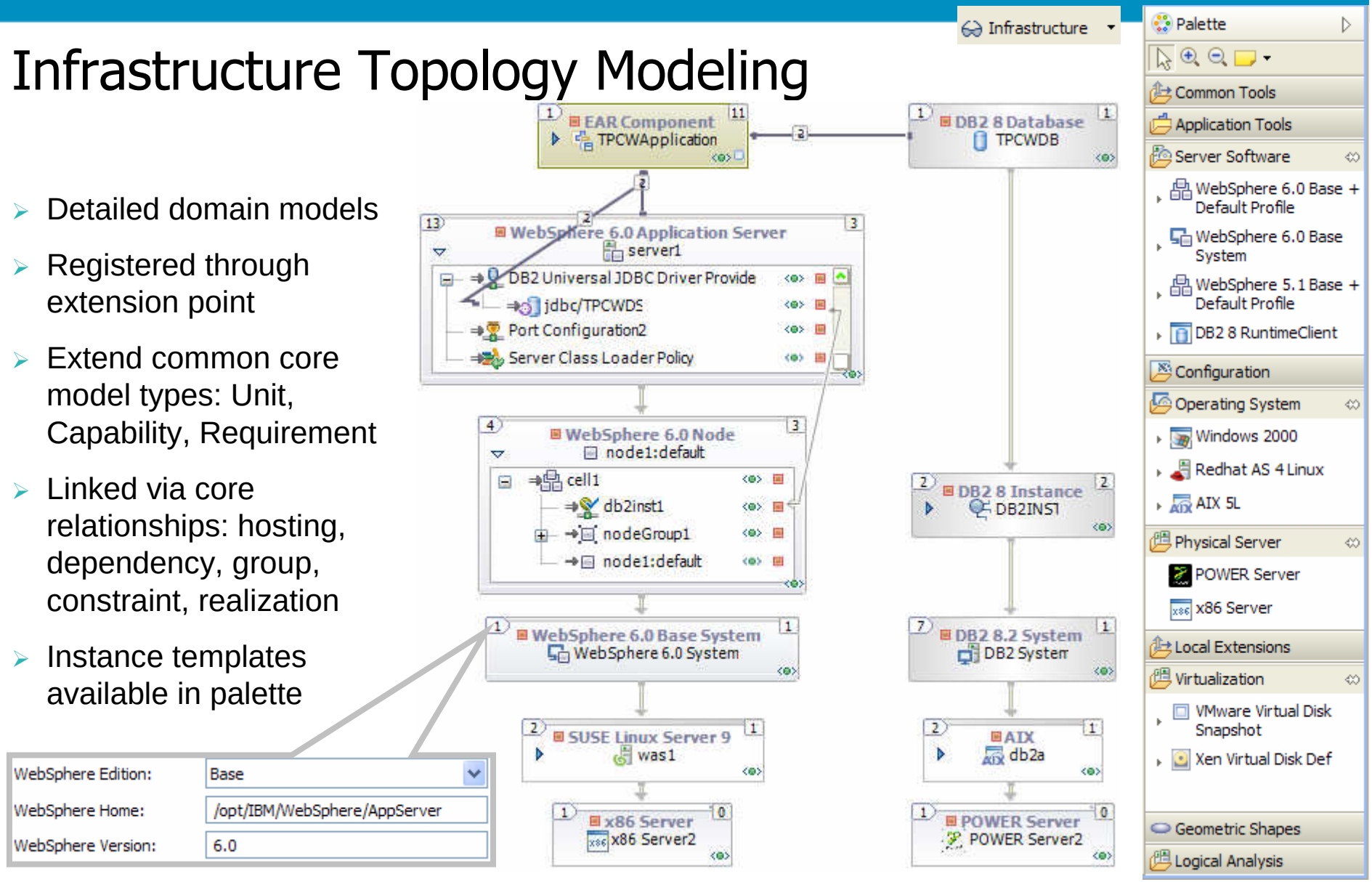
Logical Operational Modeling and UML Integration

- Analysis Domain to express operational modeling concepts
  - Locations, Nodes, Components, and Deployment Units for Non-Functional Requirements
- Multiple Layout options
- Reason about UML Component deployment instances
- Quick Action toolbars for creation of new content



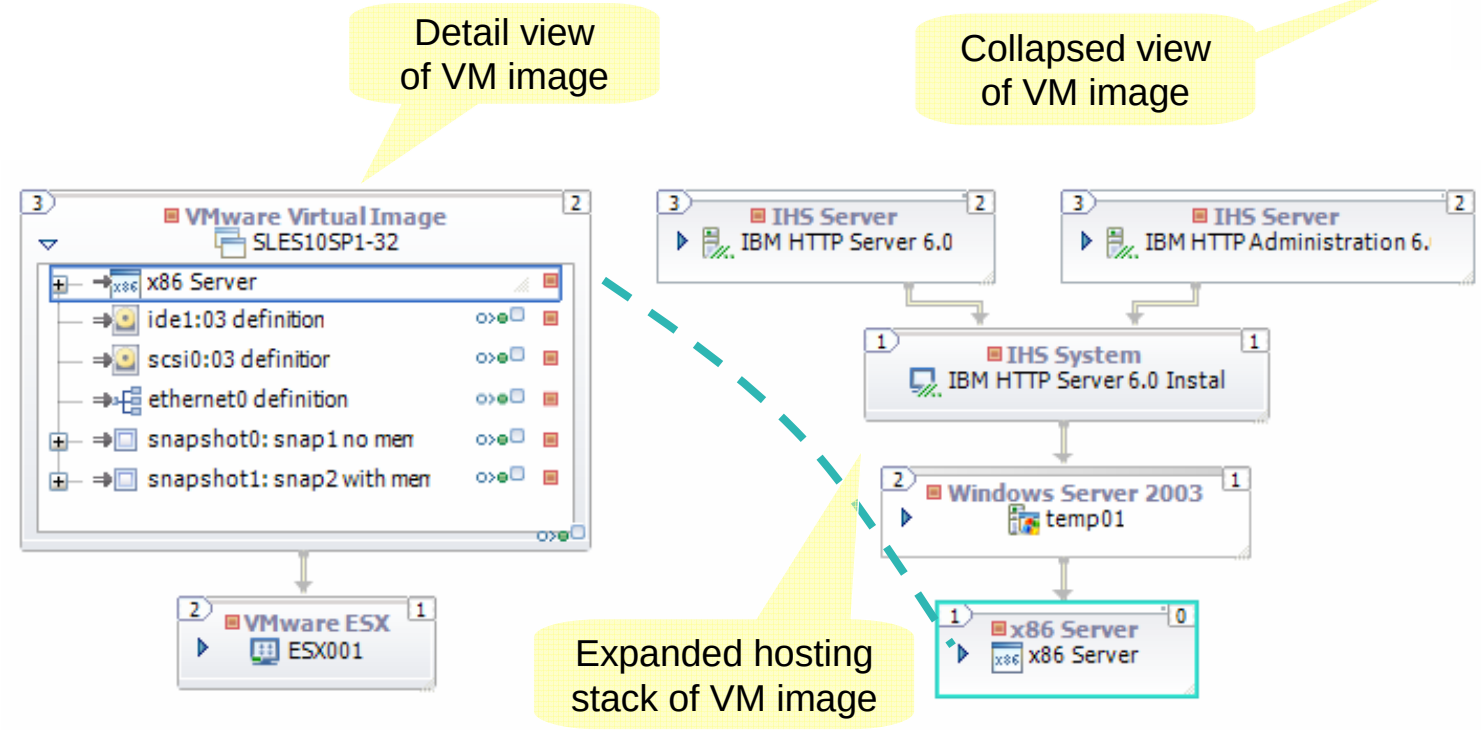
Infrastructure Topology Modeling

- Detailed domain models
- Registered through extension point
- Extend common core model types: Unit, Capability, Requirement
- Linked via core relationships: hosting, dependency, group, constraint, realization
- Instance templates available in palette



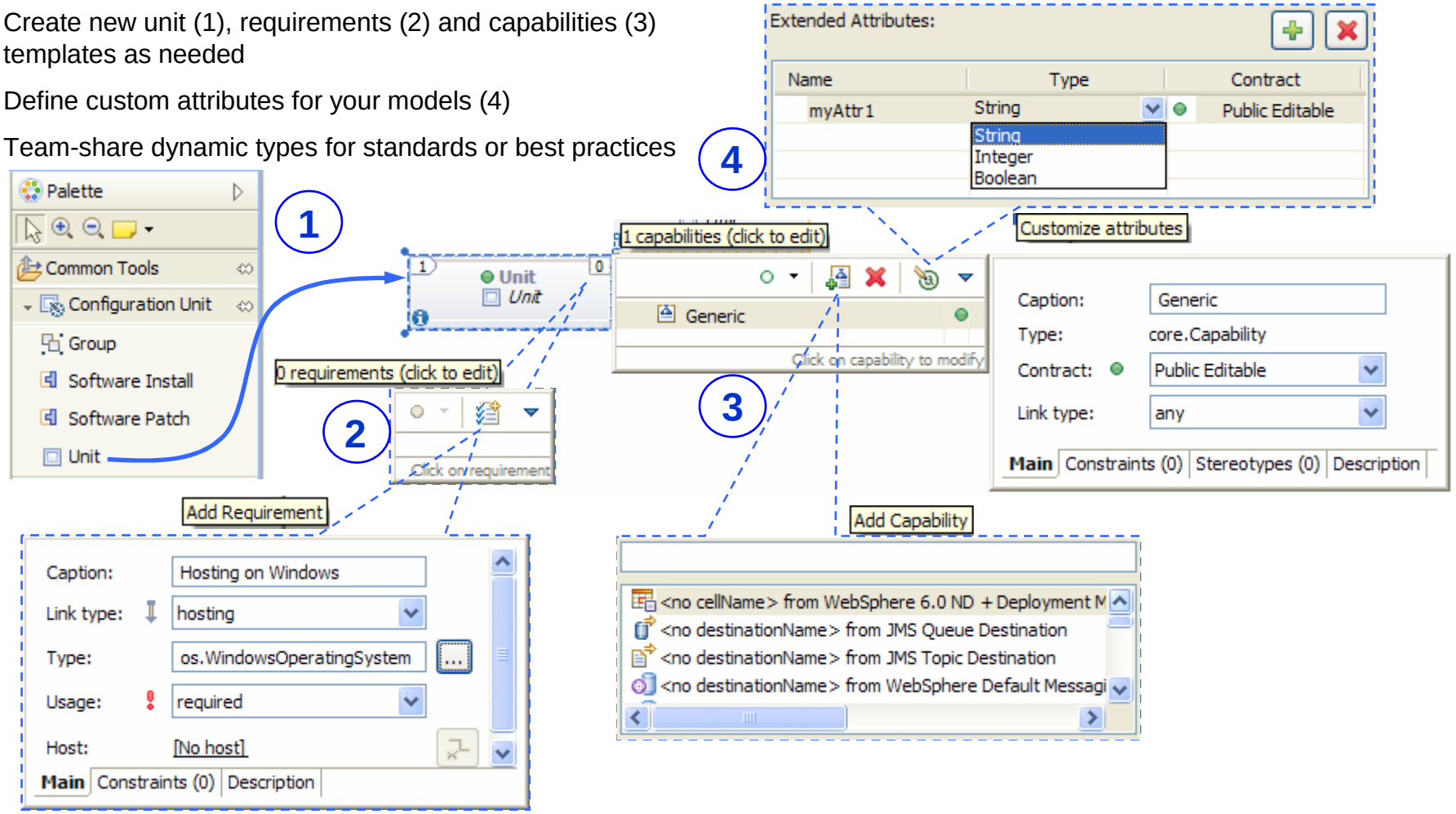
Modeling elements for virtualization

- Elements to enable detailed modeling of virtualized infrastructure
- A virtual machine image description model may be associated with a detailed model of the content of image

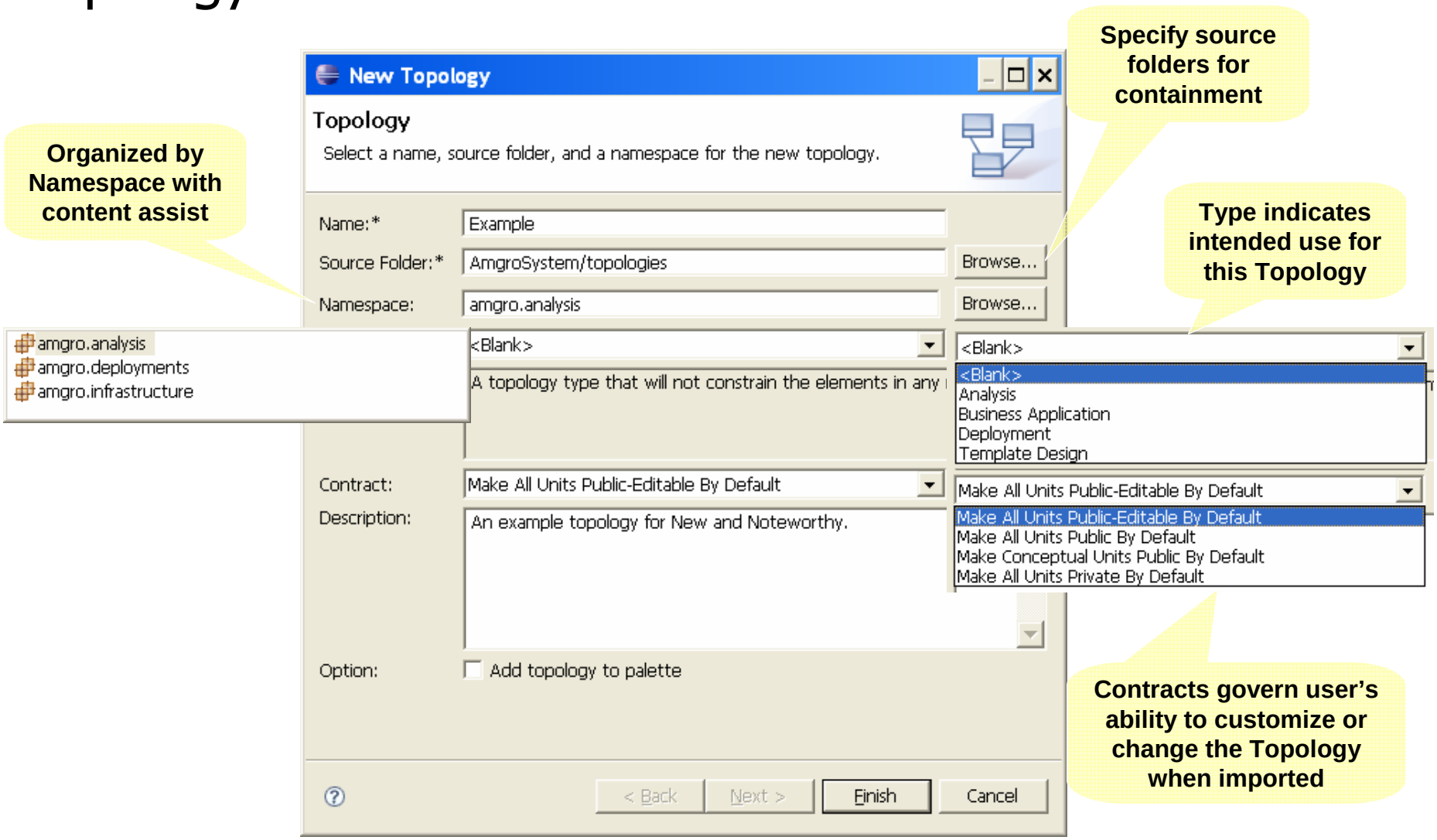


Rapid Prototyping with Generic Units

- Create new unit (1), requirements (2) and capabilities (3) templates as needed
- Define custom attributes for your models (4)
- Team-share dynamic types for standards or best practices

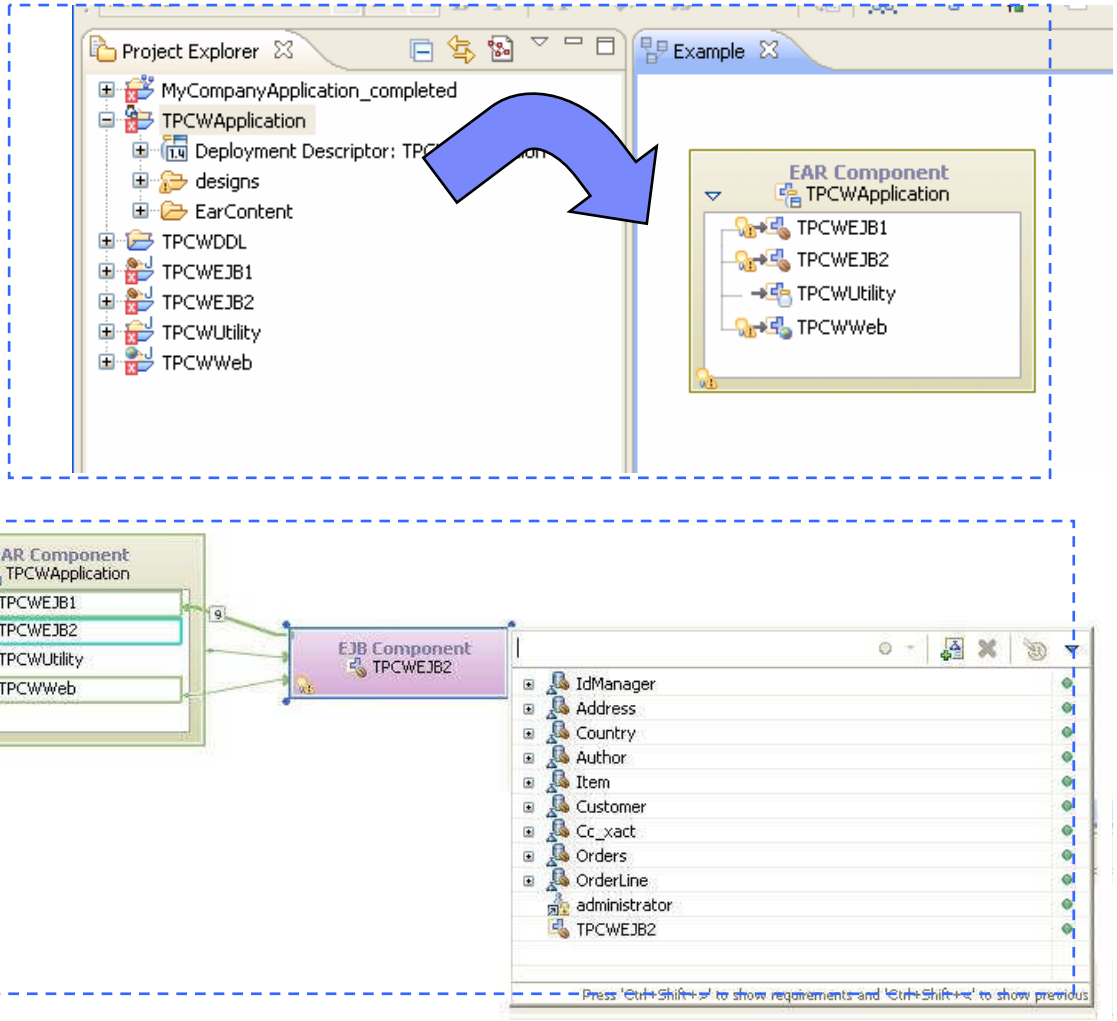


Topology Creation



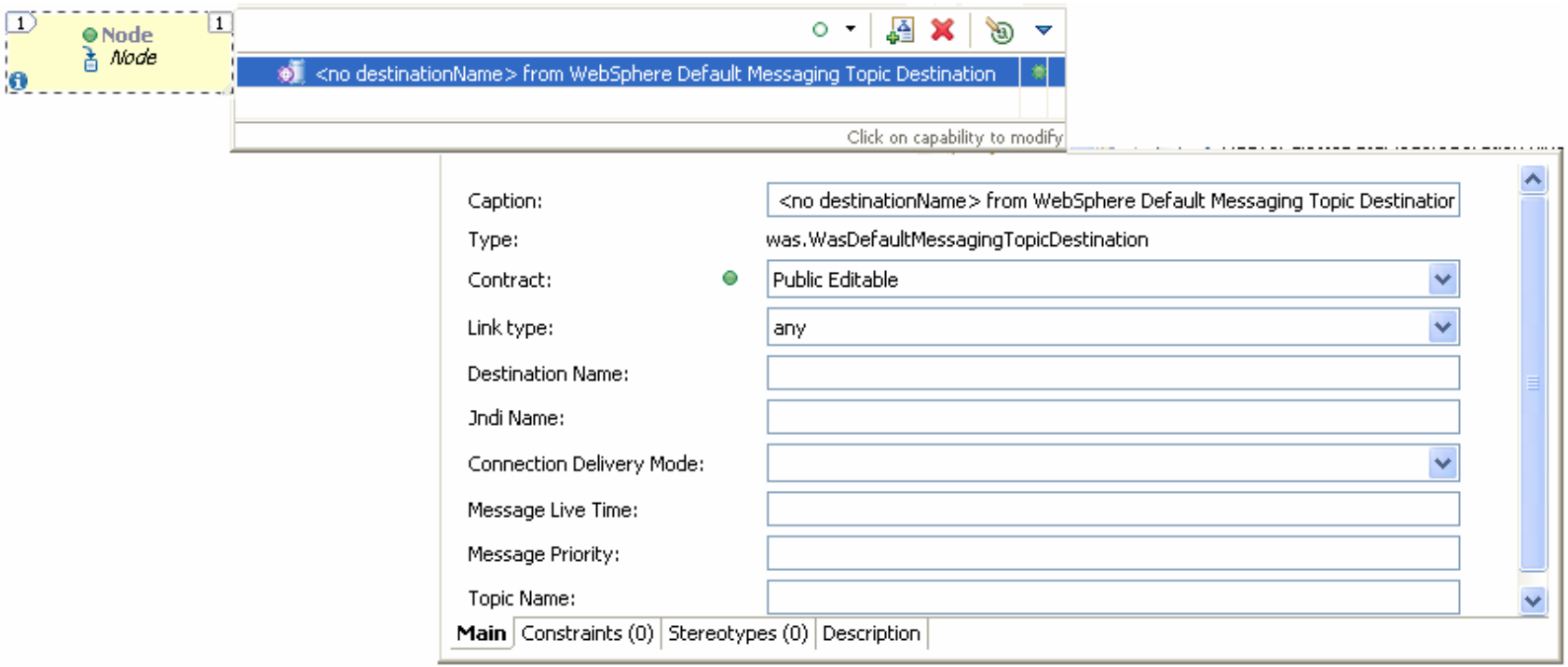
Unit Creation from Artifact Data (Java EE, DDL)

- Units for Java EE Components and *.ddl files can be created from the Project Explorer
- Extensible provider mechanism provides support for Java EE projects and archives by default
- Deep introspection reads deployment descriptor information and creates appropriate model elements



Quick Property Flyouts

- Seamless editing experience in-canvas which improves usability by focused navigation and removing the need to leave the diagram
- Tabbed folders for organization and incorporated toolbars to identify layered model modification actions
- Instant synchronization with models and validation



Property Sheet consistent with Quick Flyouts

- Property sheet reuses the same views as the Quick Flyouts giving the tooling a consistent look and feel
- Gives a holistic view of the model's attributes while providing model navigation and editing ability through tabs and table selection

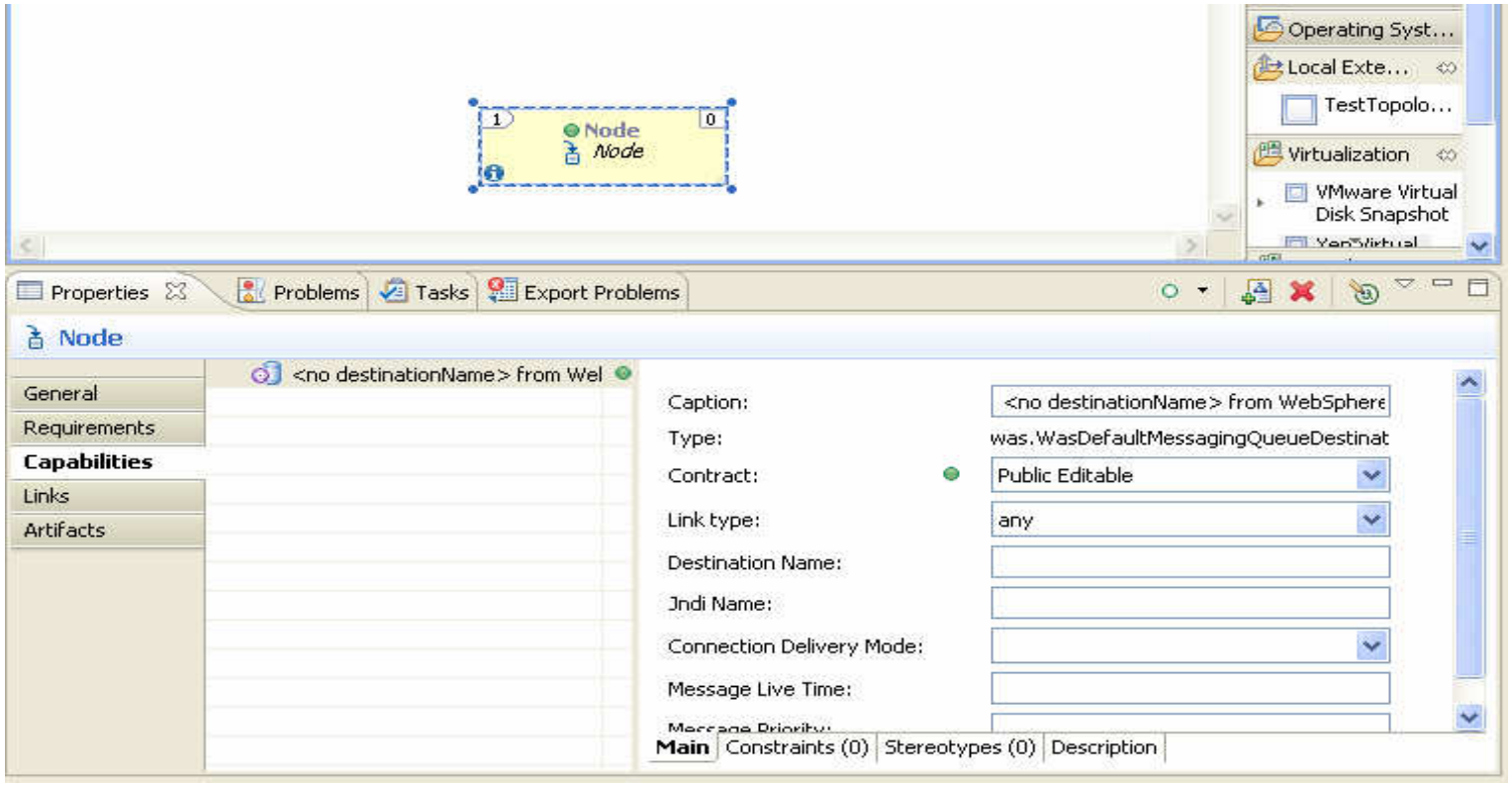
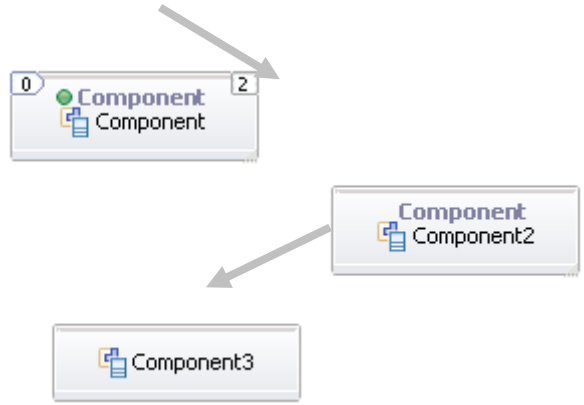


Diagram Flexibility: Figure Detail

- Figure details can be customized to suit the model.
- And the level of detail can be set for all topologies or per diagram or per figure.



- Figure color and font can also be set per figure.

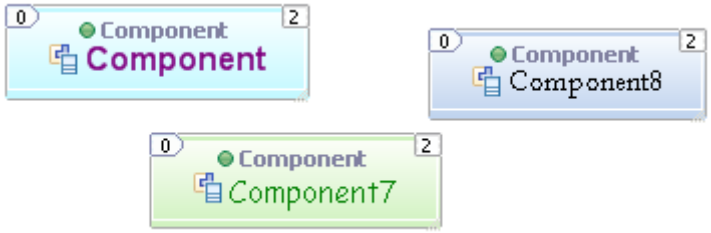
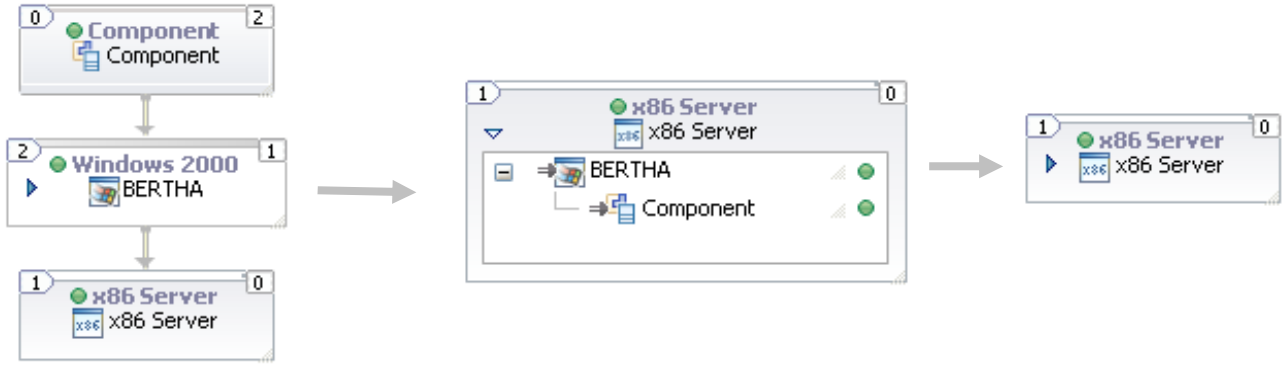


Diagram Flexibility: Support for Large Diagrams

- Linked figures can be converted into trees and collapsed to minimize diagram congestion.



- Trees can also be converted into shape containers to suit the type of model.

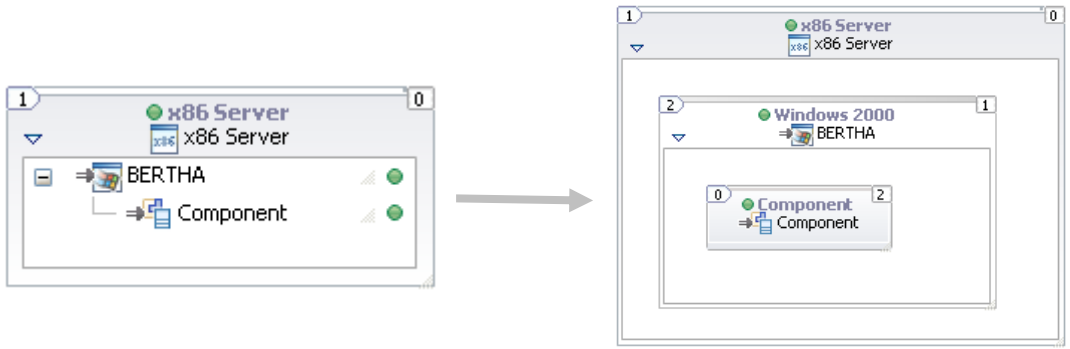
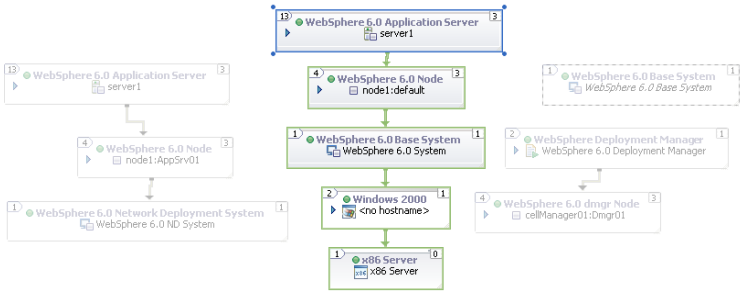


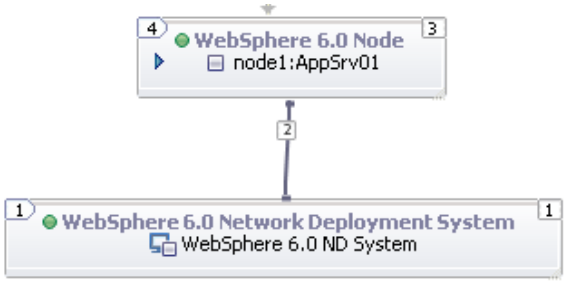
Diagram Flexibility: Support for Large Diagrams

- Selected figures are highlighted to stand out in a congested diagram.



- Links to figures outside of the screen are labeled with the target of the link.

- Multiple links between the same two figures are consolidated into one link.



- Links cross underneath any intervening figures.

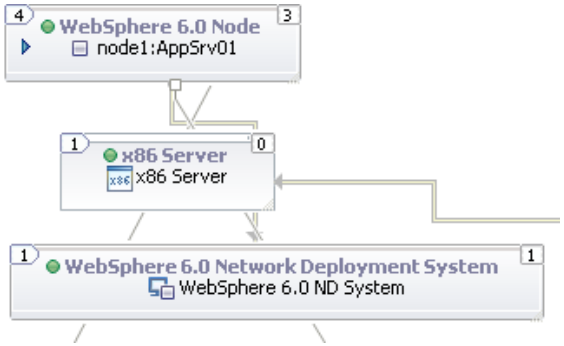
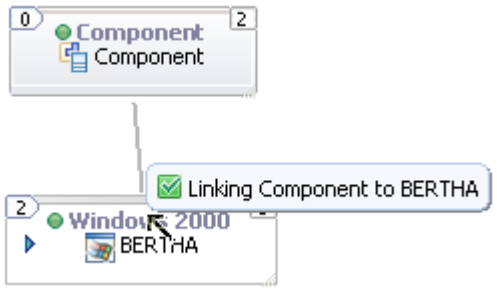
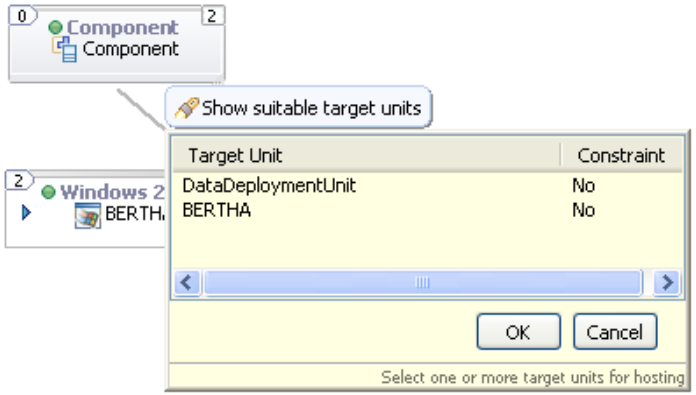


Diagram Flexibility: Support for Link Creation

- Links can be created by dragging a link tool between figures.



- Or by dropping the link tool on the diagram to search the model for eligible targets.



- Dragging a link tool to an ineligible target will open a dialog in which the target can be made eligible and the link created.

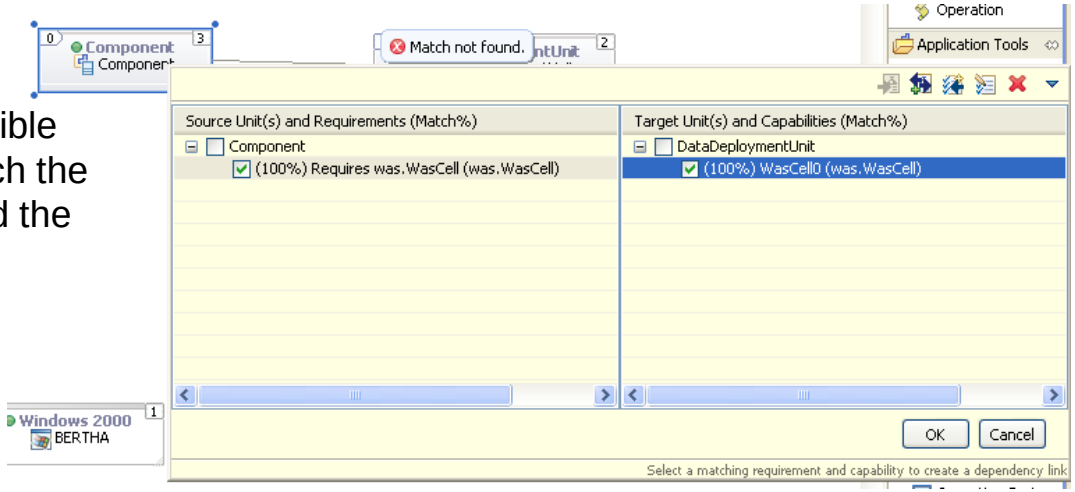
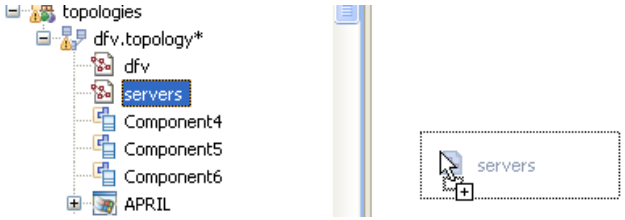
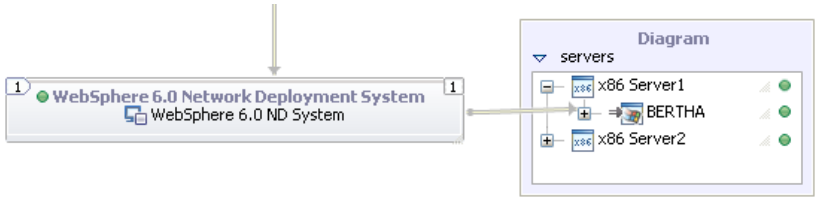


Diagram Flexibility: Multiple Diagrams

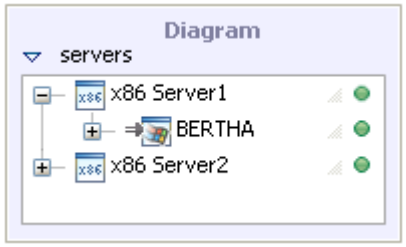
- Large diagrams can be split up into smaller diagrams.
- Any external diagram can be added to the current diagram by dragging it from the Project Explorer:



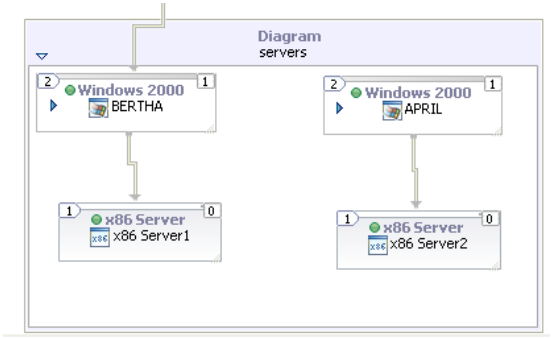
- Links can then be created to any figure in this diagram figure:



- These external diagrams can be shown as space saving trees:



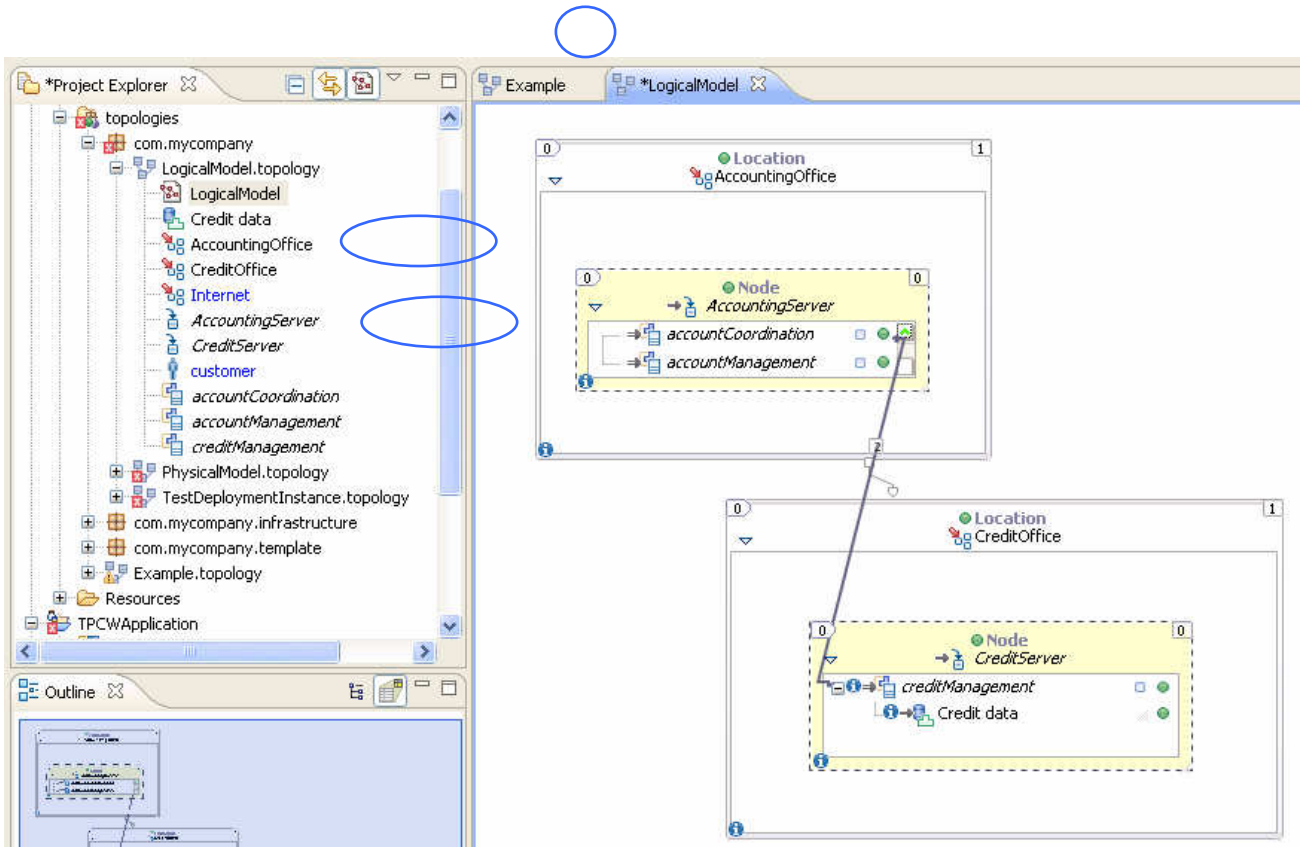
- Or as the original diagram:



- Or collapsed:

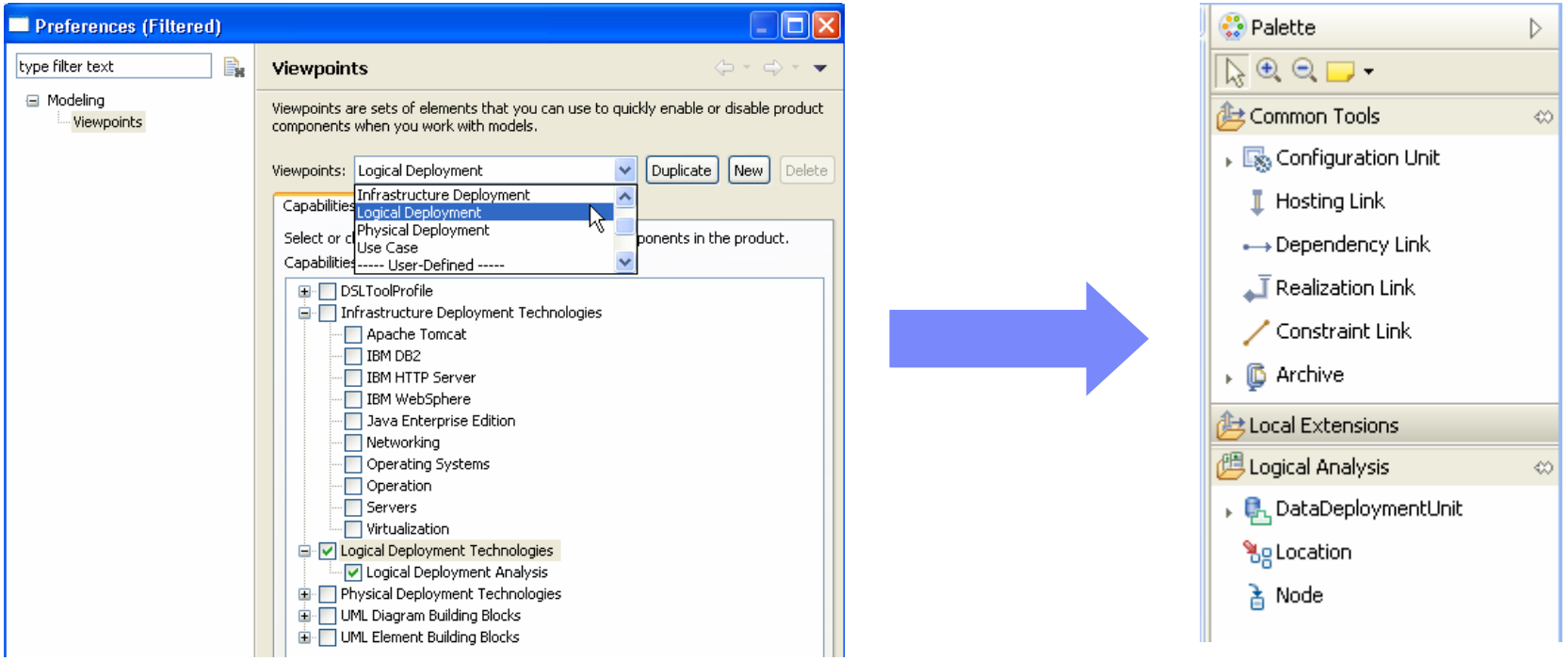
Highlight Non-Visualized Elements in Topologies

- Easily see which model elements are not shown in a diagram
- Elements which are not visualized are highlighted in blue when the toggle is depressed



View Point Integration for Palette Filtering

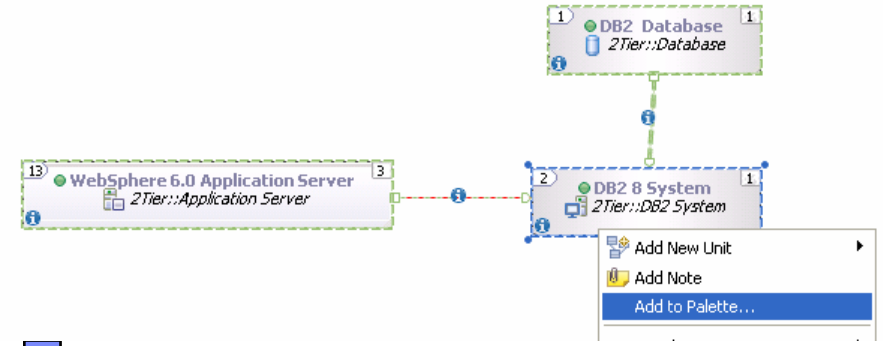
- Ability to filter palette entries based on view point selection
  - Logical filters on UML, Analysis and Core Models
  - Physical Deployment Technologies filters based on domains participating in physical topology creation
  - Infrastructure Deployment Technologies filters based on domains participating in infrastructure topology creation

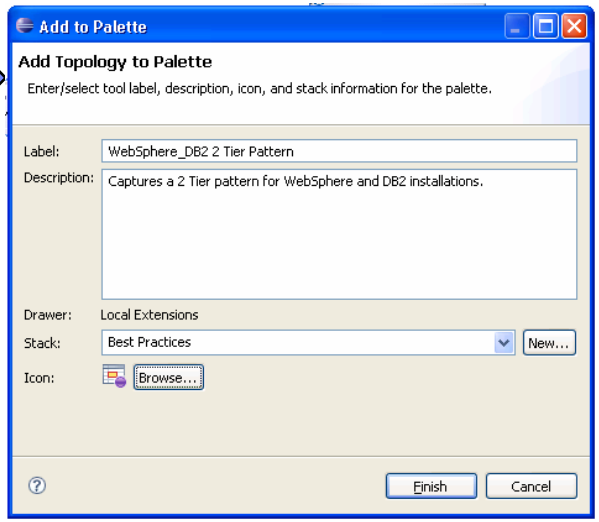


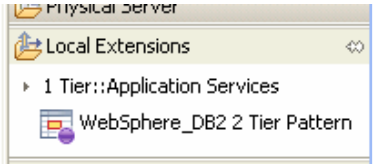
Topology Templates capture standards and common patterns

- Create a pattern that is commonly used or a best practice for your organization (e.g. a Model Snippet)
- Create templates on the fly
  - Select the related elements on the canvas
  - Right-click on one of the selected Units and choose “Add to Palette…”
- Define a new topology as a template
  - Select the “Add topology to palette” option in the topology wizard
  - Enter palette information

See RAM integration section for template topology publishing and import.



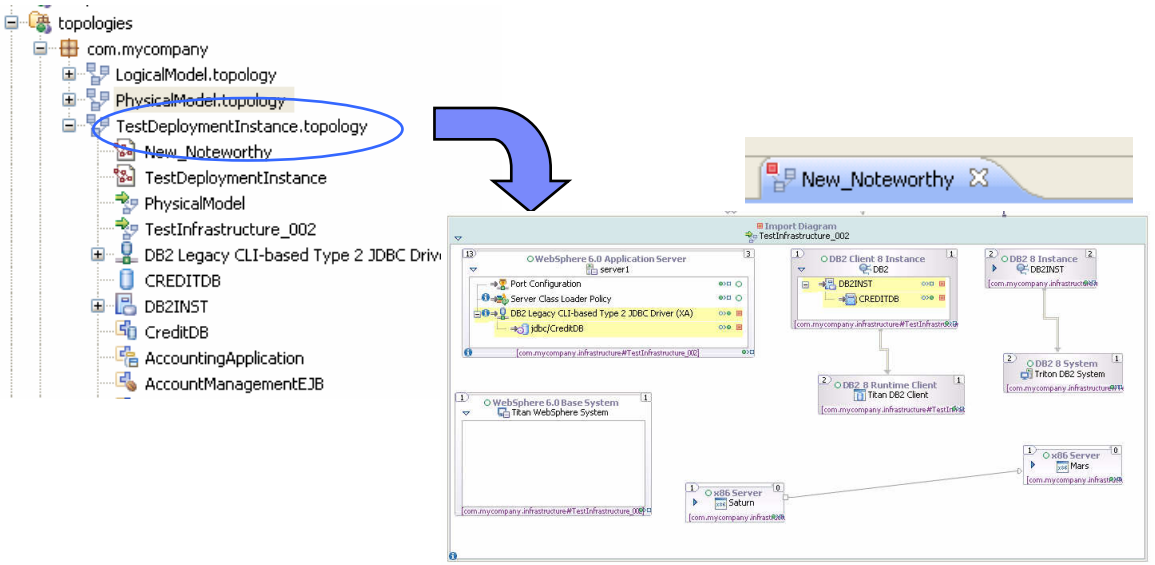


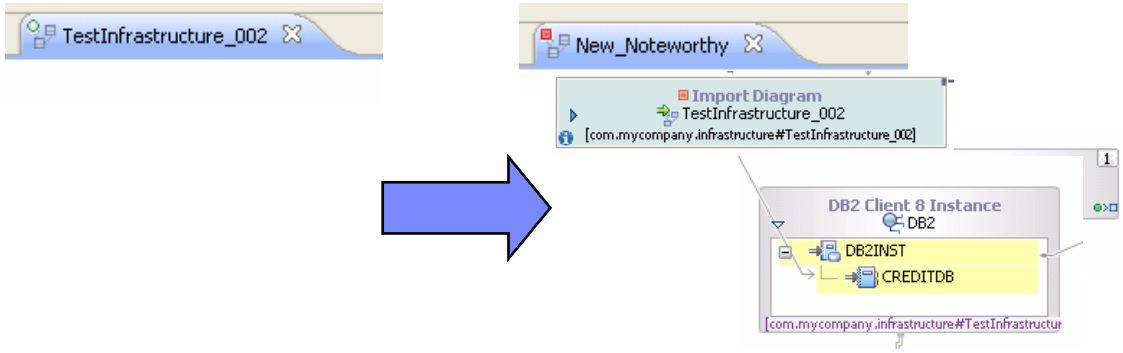


- When applied, diagram formatting and layout options are preserved
- When the project is team shared, the palette entries will be available for other users as well
- Changes made to the original template will only apply to new applications; not in cases where the template was applied previously

Model Reuse and configuration with Model Imports

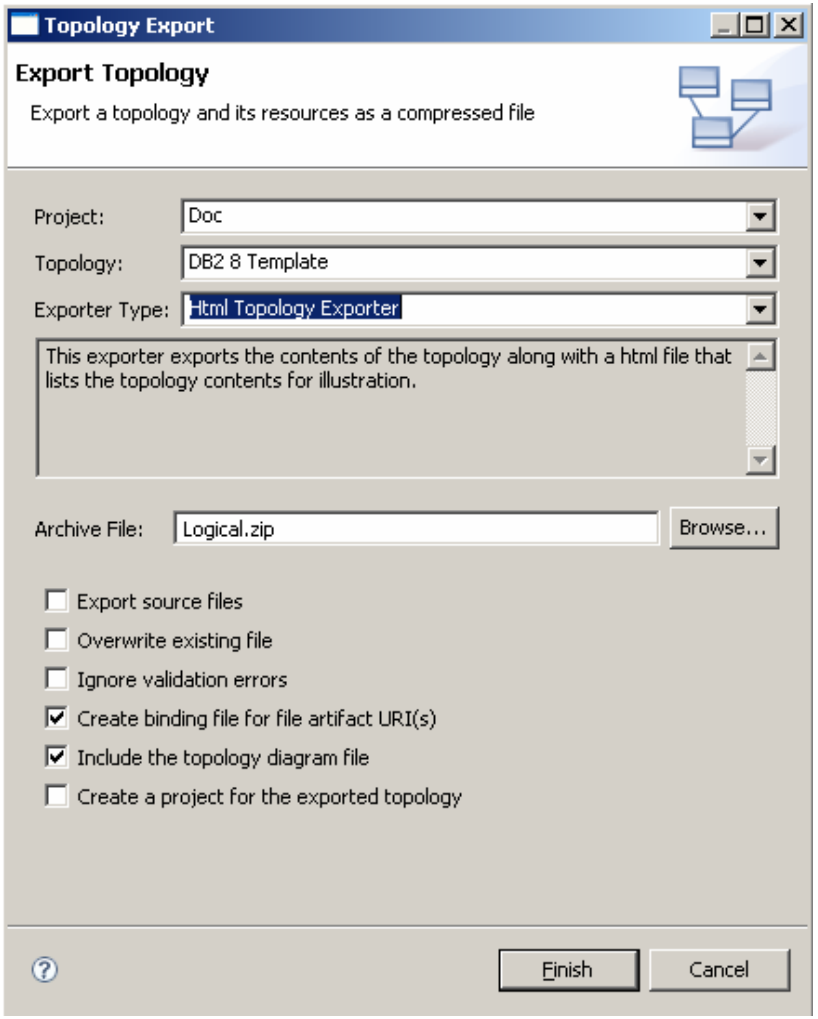
- Allows a Topology to be referenced directly
- Modifications can be made per the governance of the Topology contract
  - Designers can hide Units, fields, etc
  - Consumers can set specific values only if allowed by the contract
- Imported elements can be customized locally without affecting the original contract and the differences are highlighted



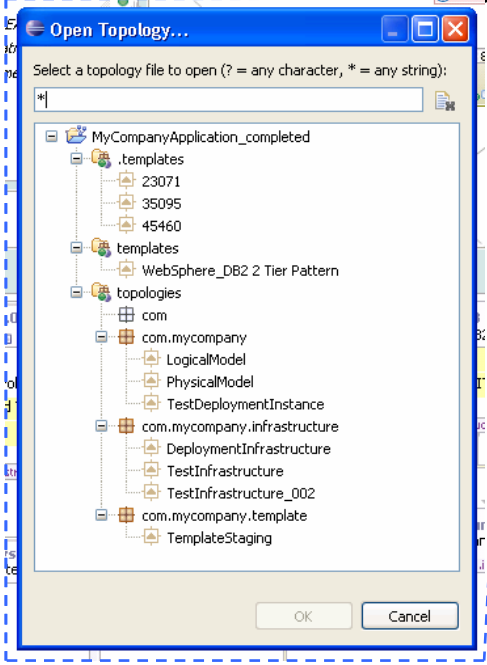


Model Sharing with Topology Export

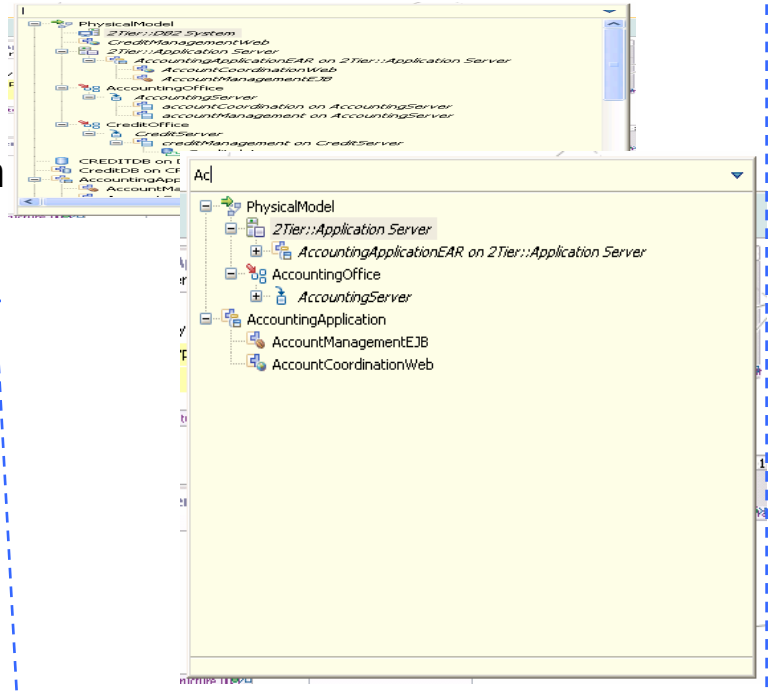
- Multiple outputs available
- Export engines can be added for greater flexibility



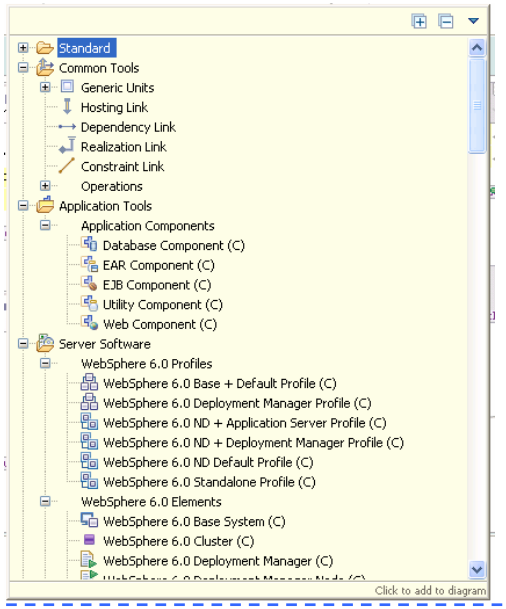
In-place Navigation helps with large models



- Any Topology can be found with Quick Open (Ctrl+Shift+I)



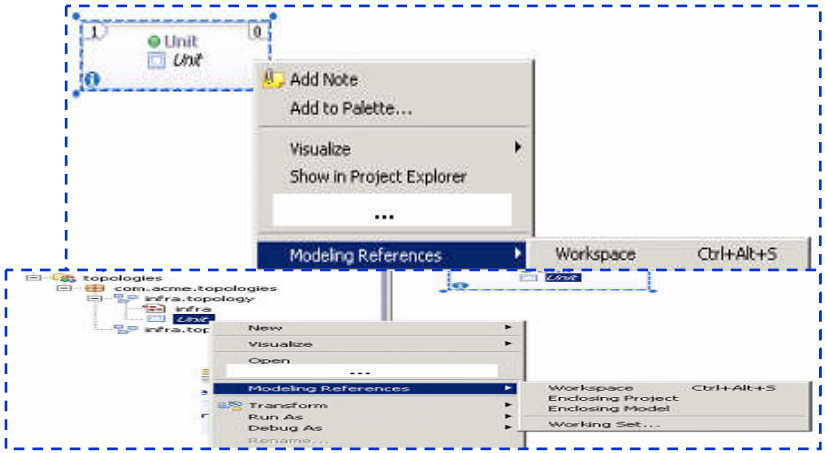
- Large diagrams are made easier to find elements with Searchable Diagram Quick Outline (Ctrl+O)

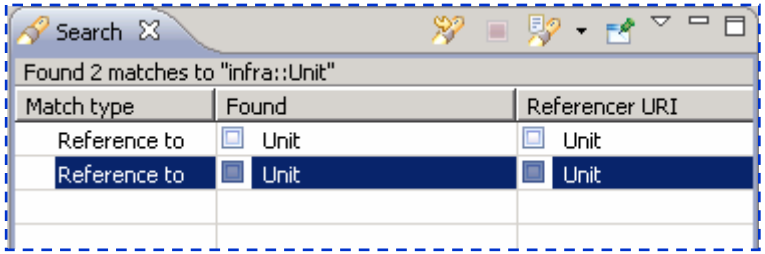


- Large palettes are made easy to navigate with Quick Palette (Ctrl+T)

Topology Search and References

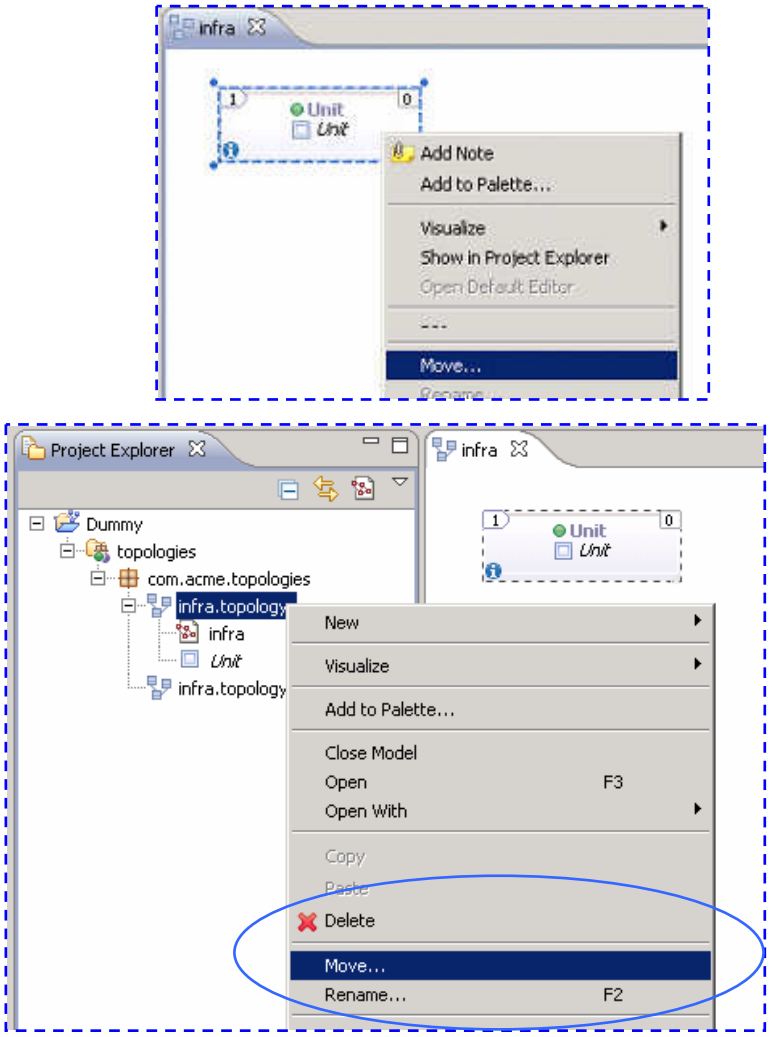
- Search for references for Topology model elements
- Invoke search from the Topology editor or the Project Explorer
- Search results similar to the existing UML Search Results view





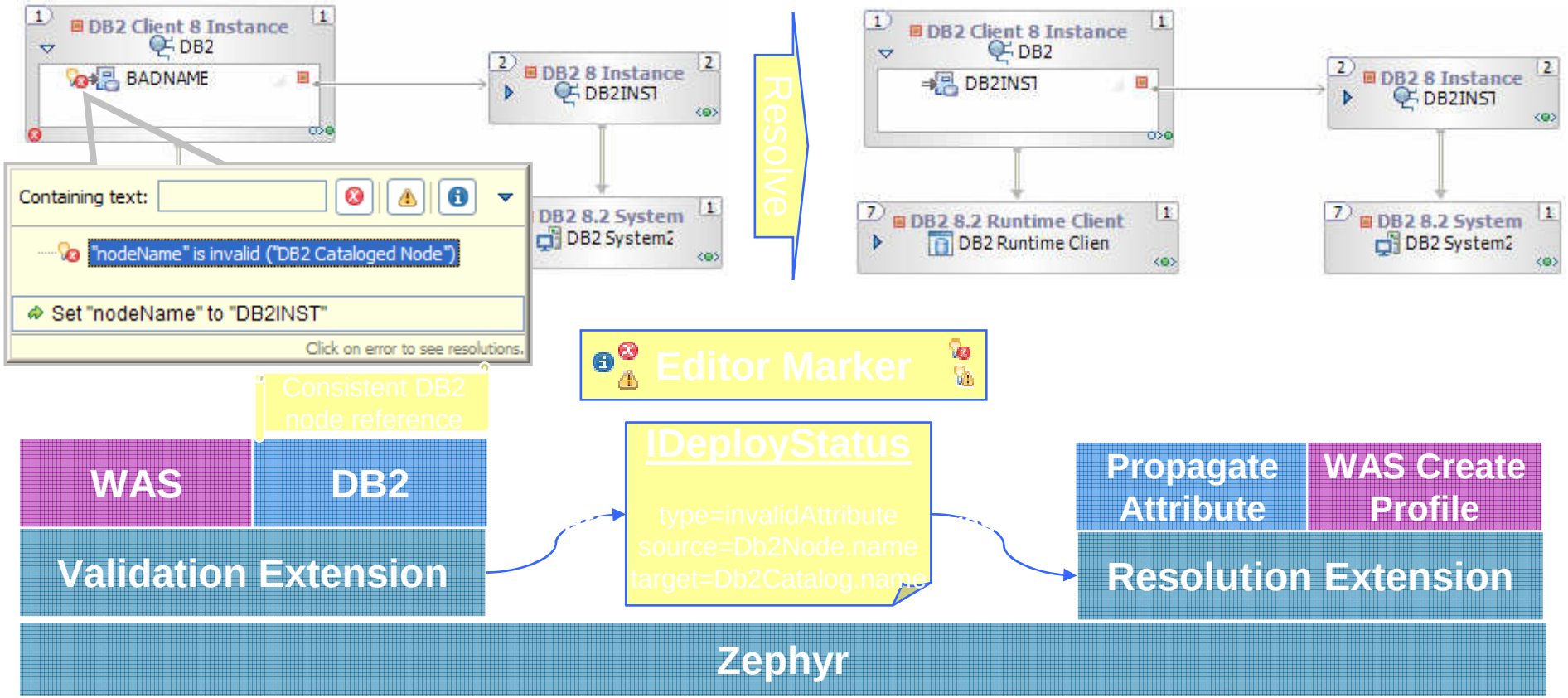
Topology Refactor Support for Move/Rename/Delete

- Integration with Refactor participant support in base Eclipse
- Move, rename, or deletion of Topology resources automatically propagates that change to referencing models and diagrams
- Individual Units can be moved out of one Topology into another
- Topology namespaces are also properly updated for move, rename, and deletion



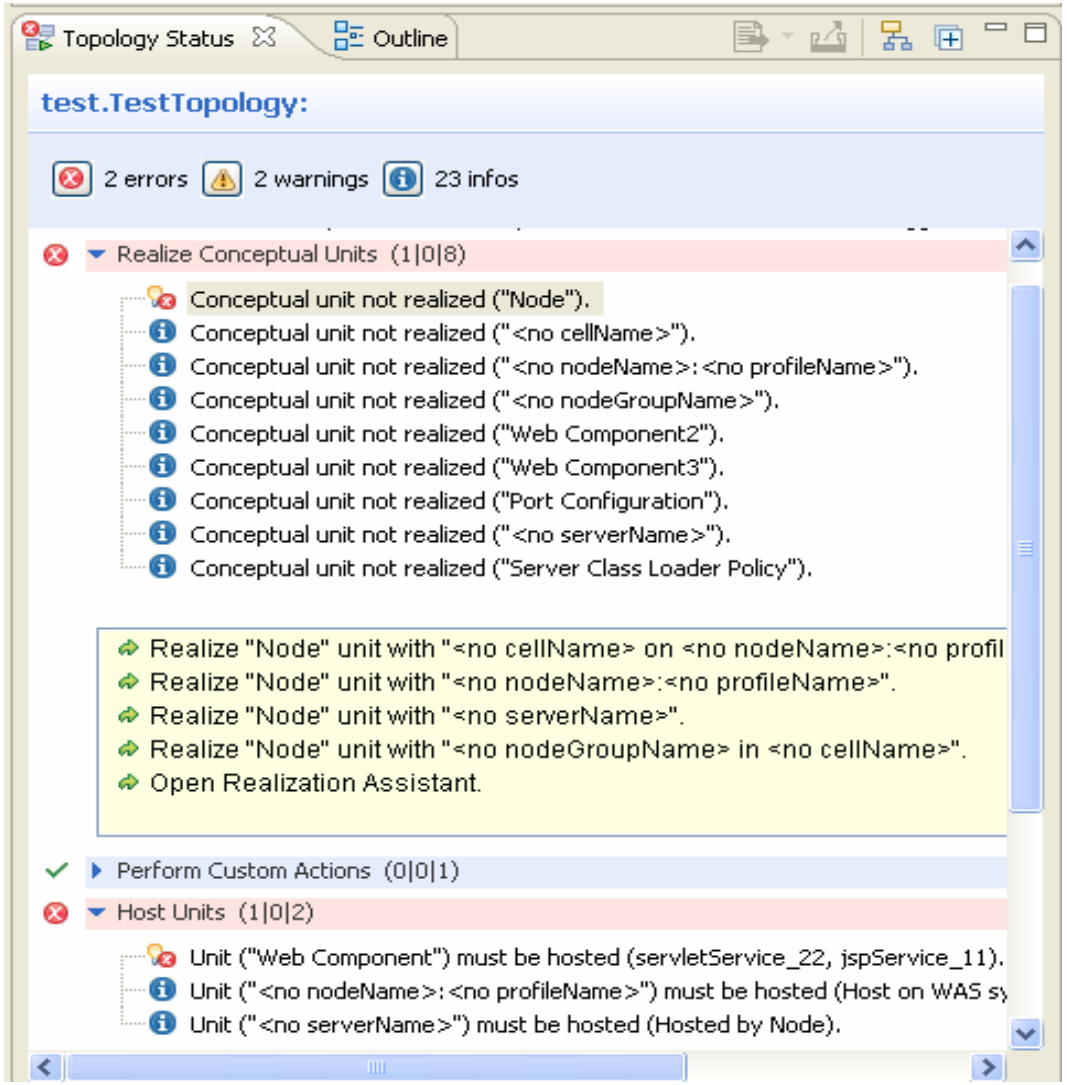
Topology Validation & Resolution

- Rich domain semantic validation
- Extensible registration
- Rich status information generation
- Resolutions register for known status
- Available to users or automation
- Operate on model to effect consistency



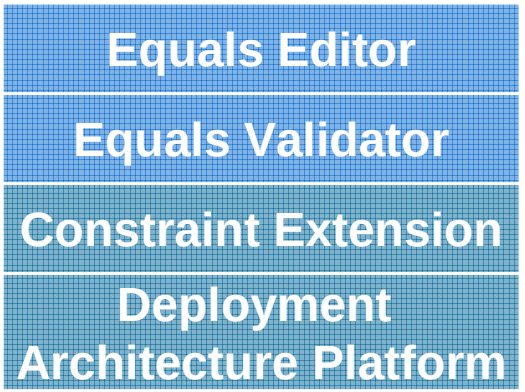
Topology Status View

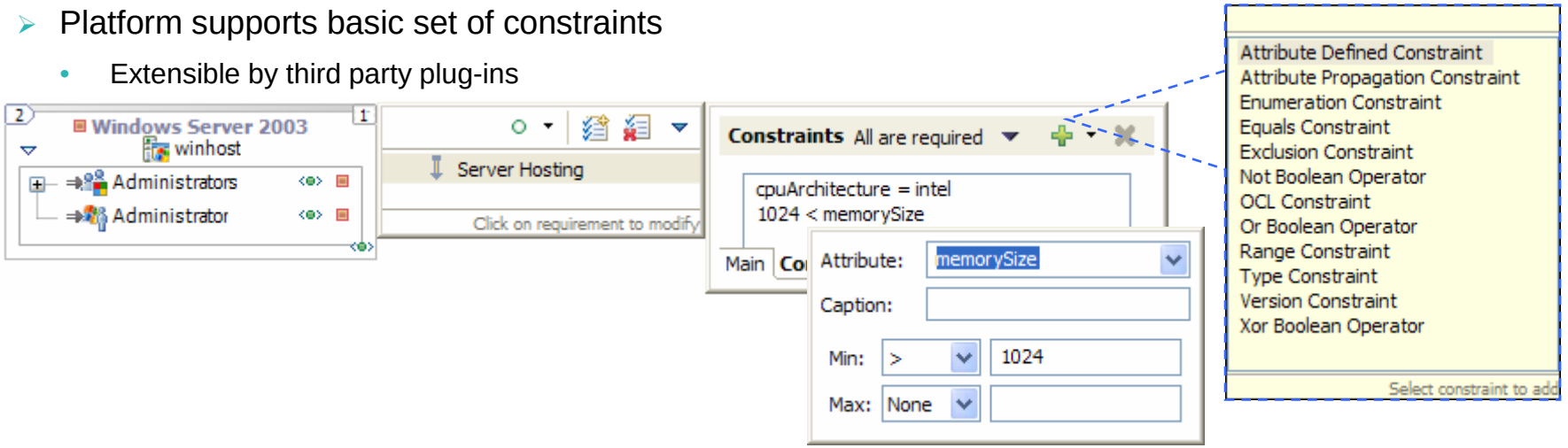
- Provides a complete overview of the topology execution state
- Prioritizes and categorizes based on problem severity in order to guide the user to resolve problems efficiently
- Extensible to support value-add from other providers via hyperlinks and resolutions
- Filtering based on problem severity
- Navigates visual models by selecting problem unit on status



Attribute Value Constraints

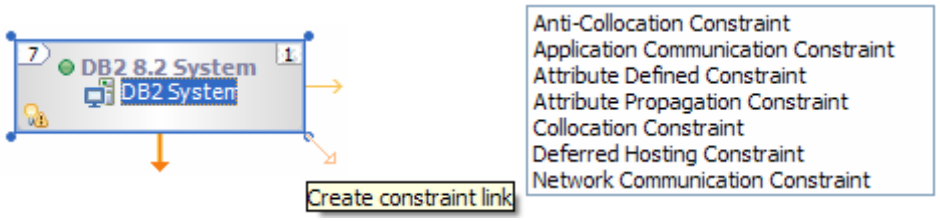
- Dynamic validation logic
  - Instance associated with model object
- Constraint unit or capabilities attributes
  - From changes or values set during elaboration
- Constraint requirement target
  - E.g. host, group, dependency target
- Constraint instances can be parameterized
  - E.g. Attribute equals constraint: architecture=intel
- Platform supports basic set of constraints
  - Extensible by third party plug-ins

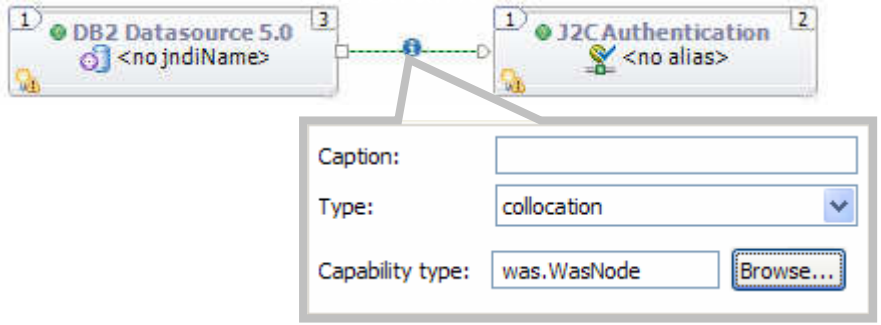


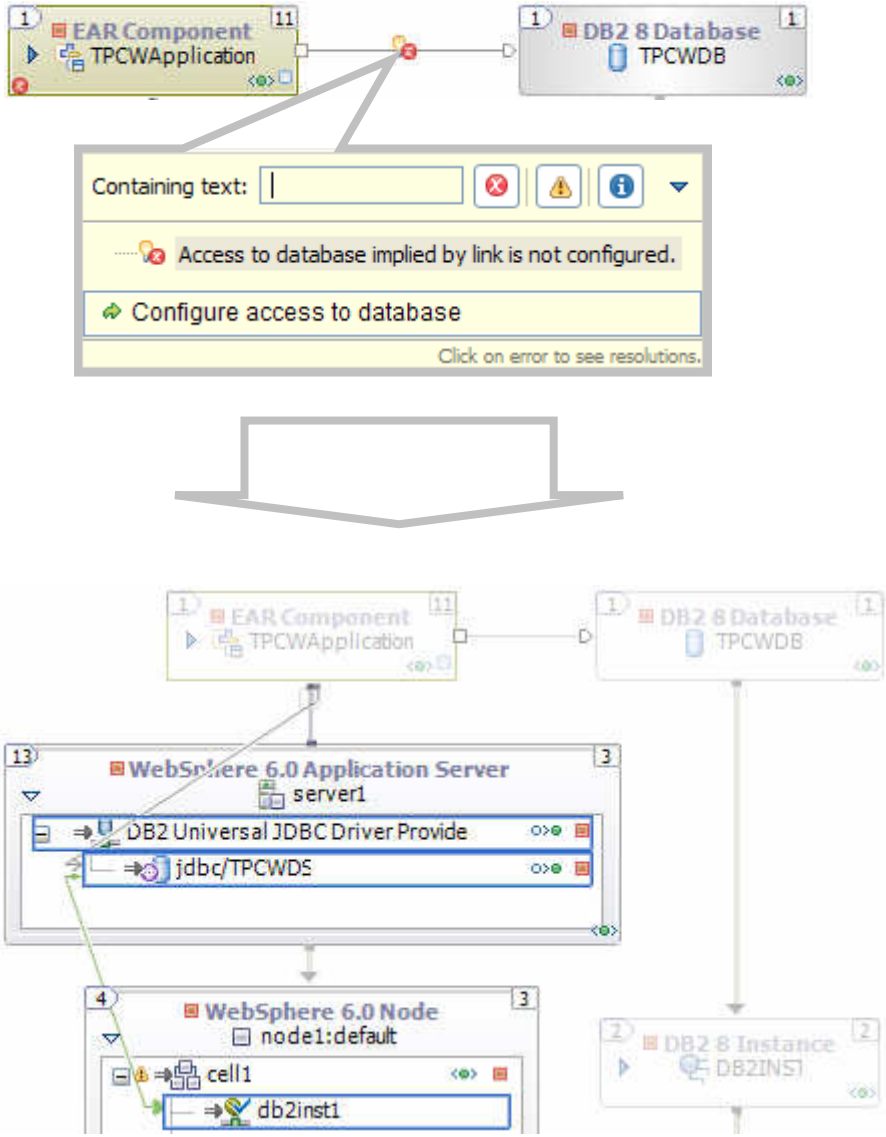


Structural Constraints

- Binary constraints over model objects
- Contained on requirement, semantic link, or constraint link
- Constraint object configurations, or linking
  - E.g. attribute propagation, anti-collocation

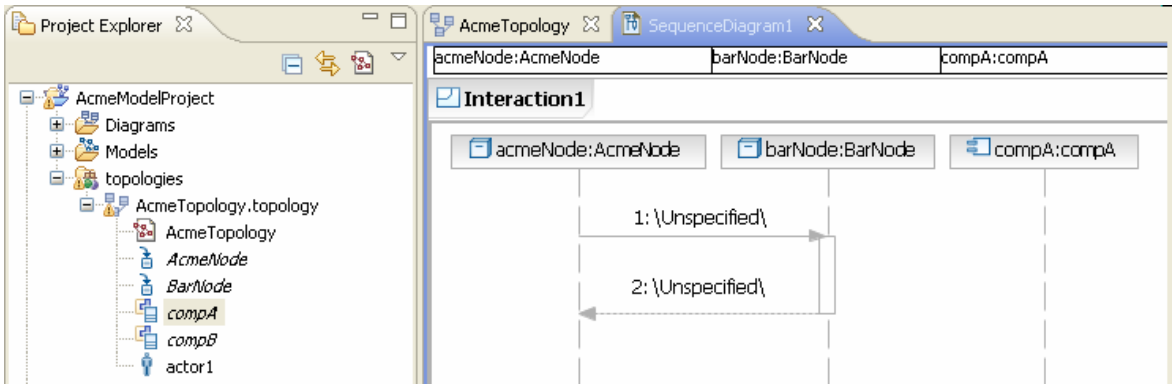




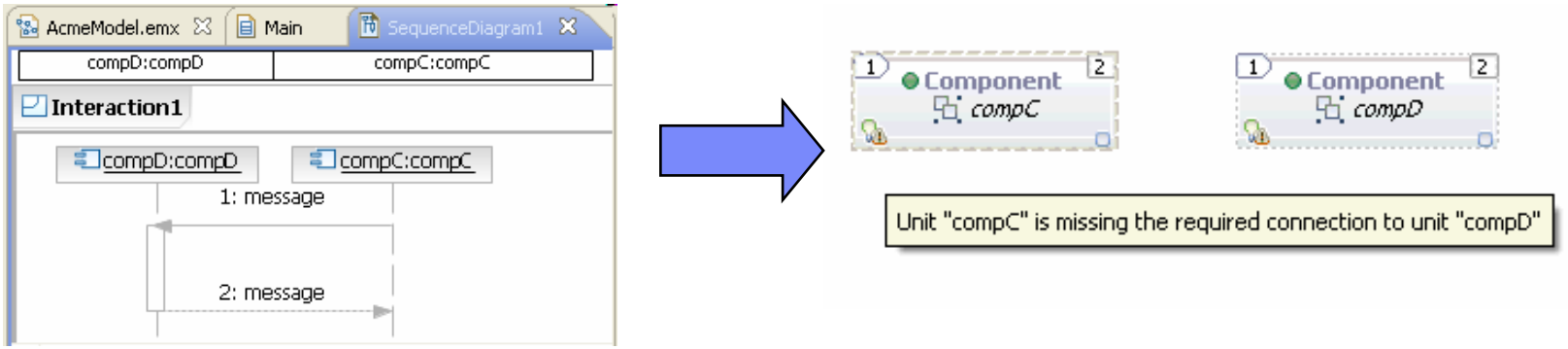


UML Interaction Models enforced in Deployment Architectures

- Deployment topology units may be visualized in UML interaction diagrams

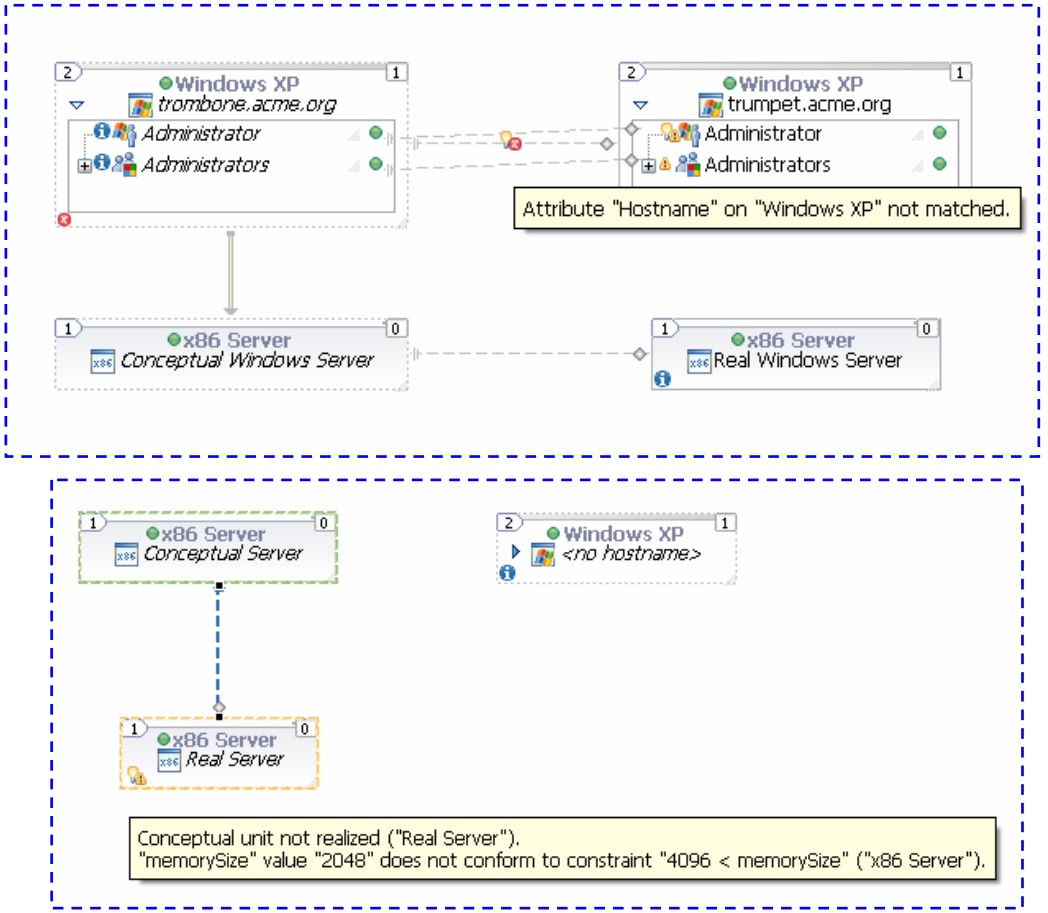


- Topologies may then be constrained by the UML modeled communications
  - Validate relationships prescribed in a sequence/interaction diagram



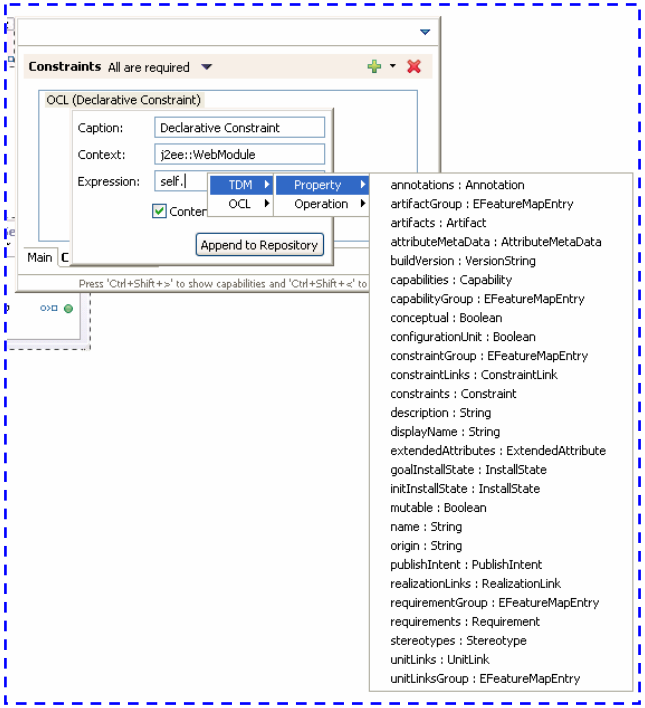
Realization Constraints

- If set, literal values must match between source and target realizations
- Validation ensures the target is consistent with the source, including attribute values and relationships (e.g. hosting)
- When a source is constrained, the target realization must conform to the limitations of the source
- Validation provides Warning statuses when values are not in the proper range, set, or value or are set to a restricted value

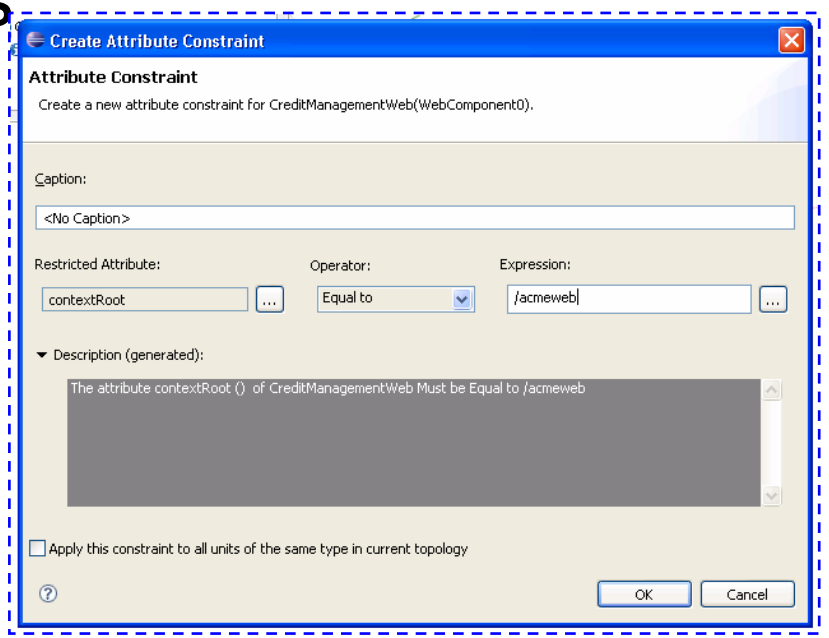


OCL Constraints for Topologies

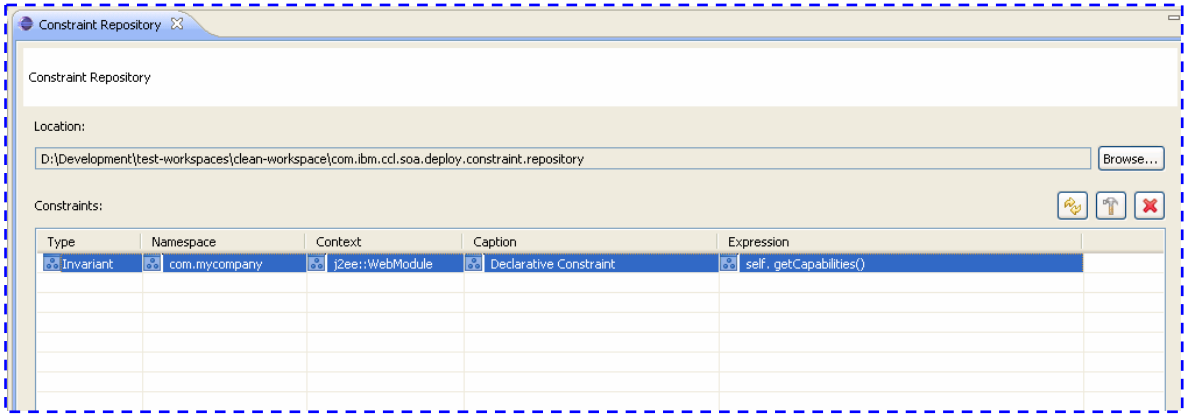
- Easy to use Wizard Dialog to create OCL Constraints



- Content assist to make model navigation easy

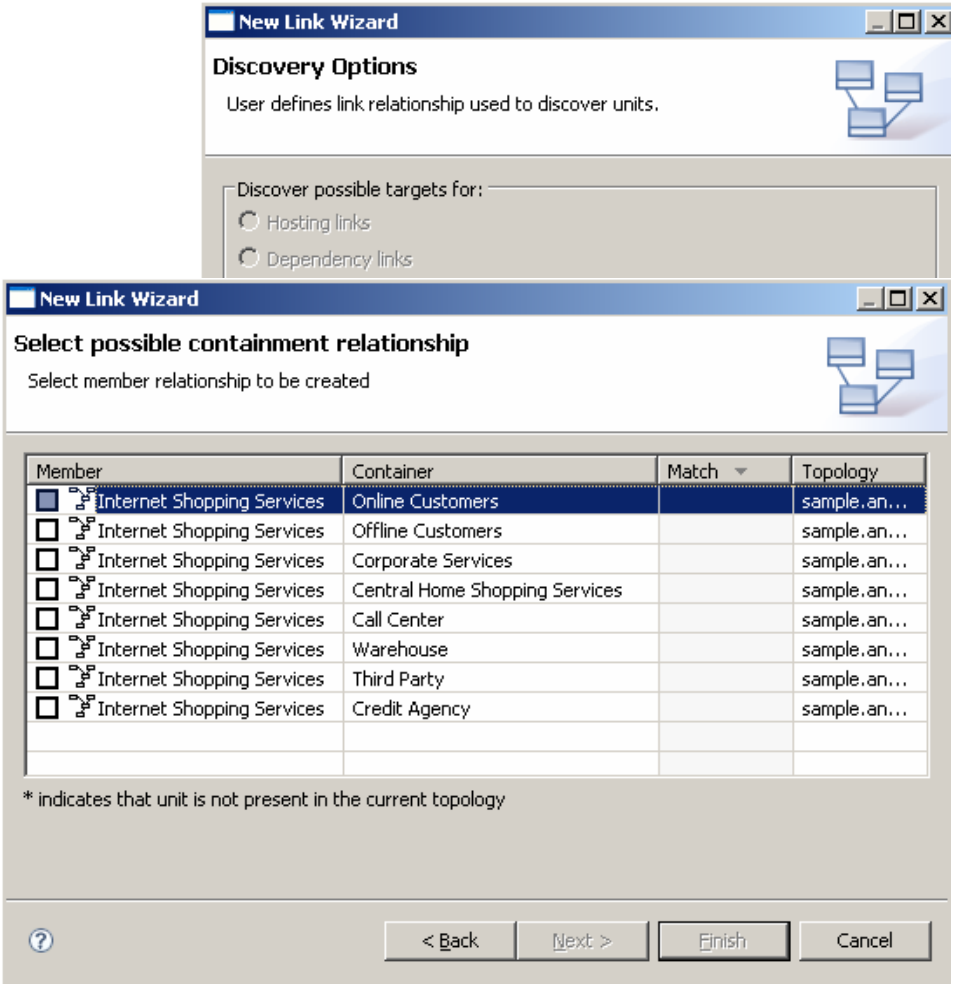


- Constraint Repository view makes it easy to reuse snippets of complex OCL expressions across models

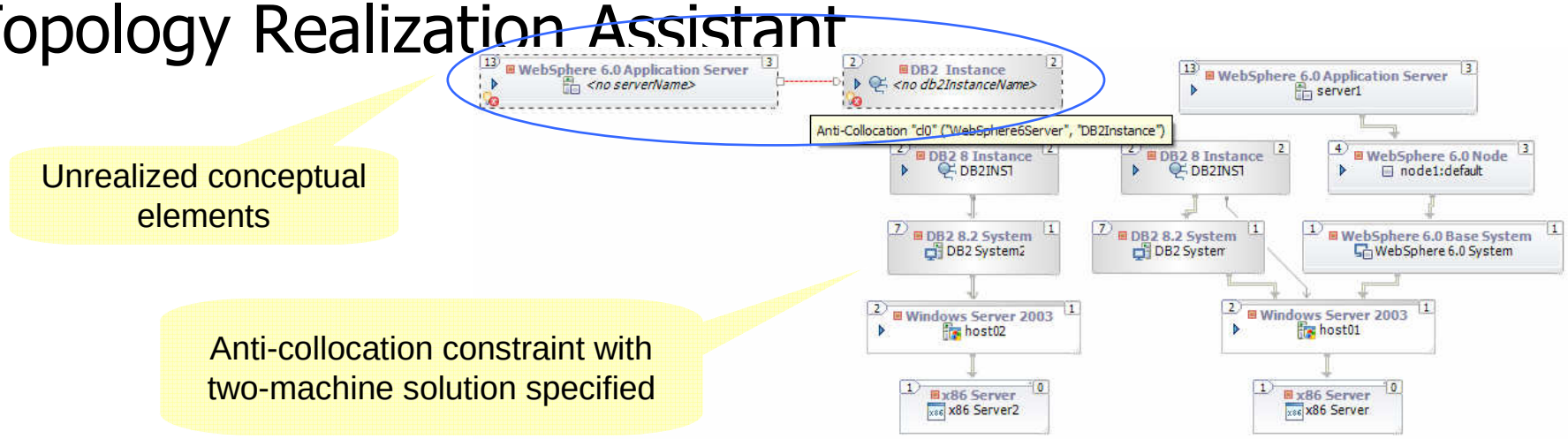


Topology Model Discovery

- Discover available targets within the Topology or from external resources in order to satisfy requirements
- Multiple relationships
  - Hosting
  - Dependency
  - Container/Member



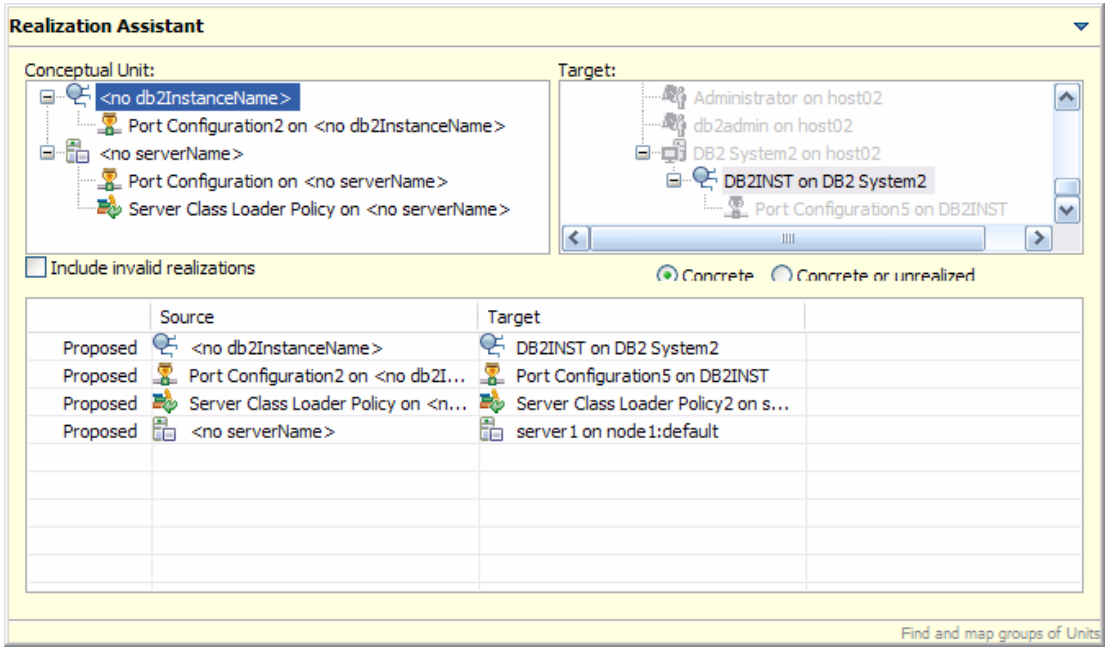
Topology Realization Assistant



Unrealized conceptual elements

Anti-collocation constraint with two-machine solution specified

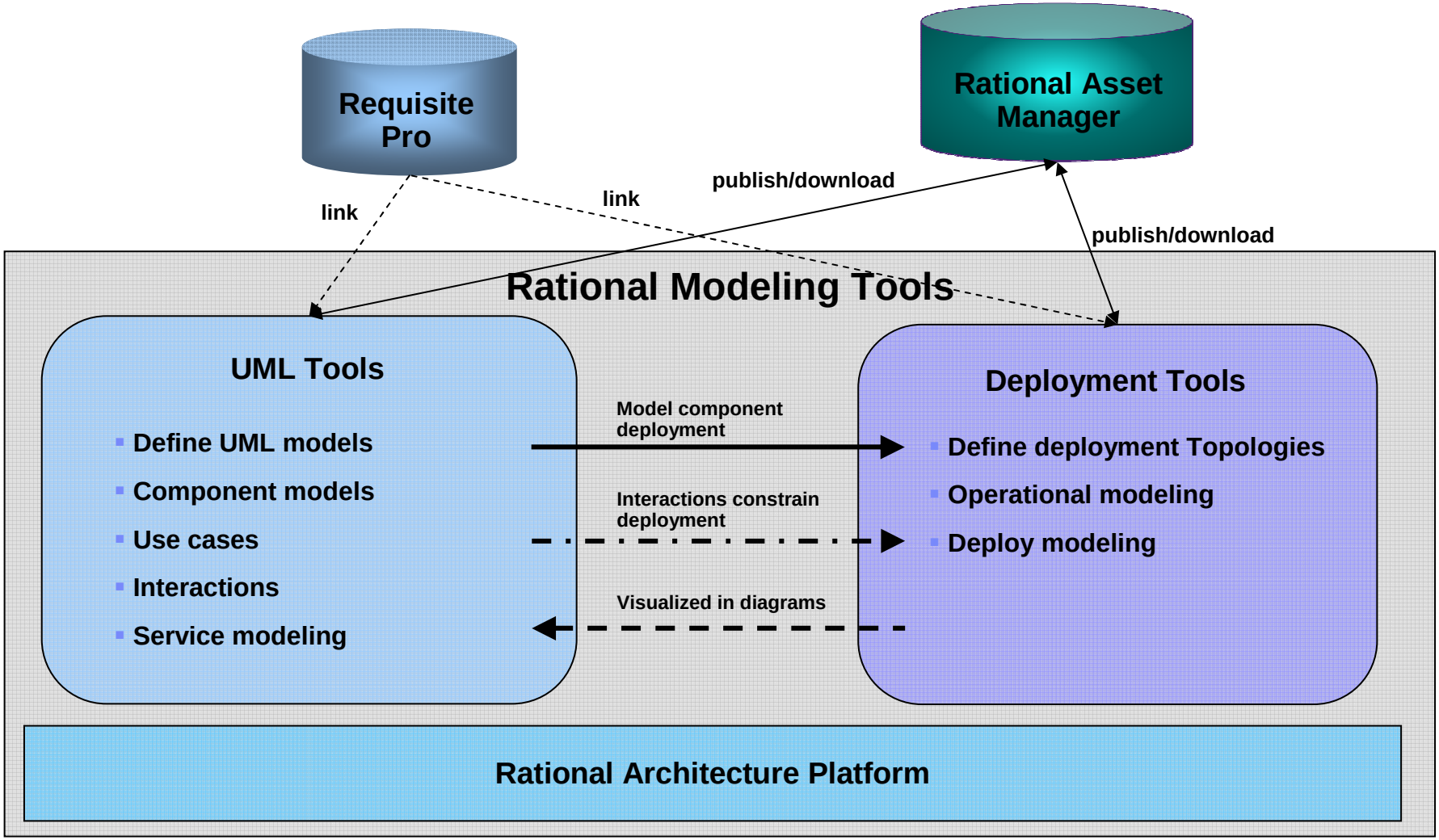
- Realization Assistant guides the user to valid realizations of an entire conceptual model to a model of similar concrete infrastructure
- *In example, the collocation constraint indicates that a two-machine solution is desired*
- *The user is not shown the invalid 1-machine mapping*



Product Integrations

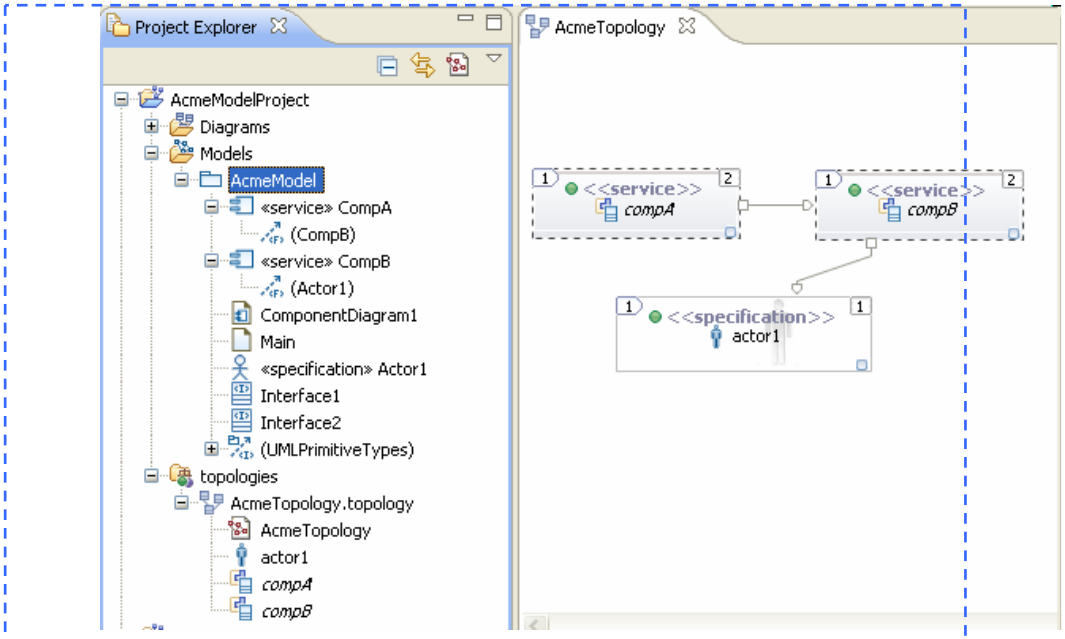
The deployment architecture platform is integrated with the Rational modeling platform as well as other Rational products.

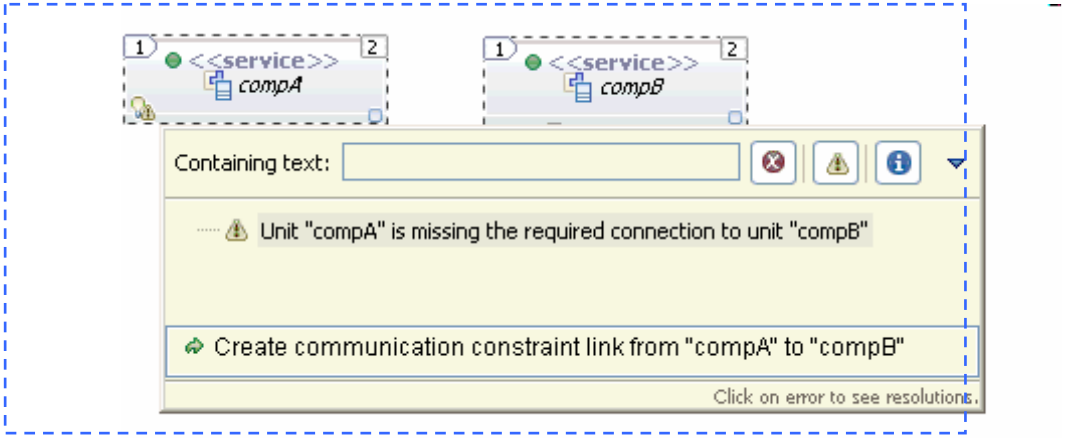
Key Rational Integrations



UML Interoperability

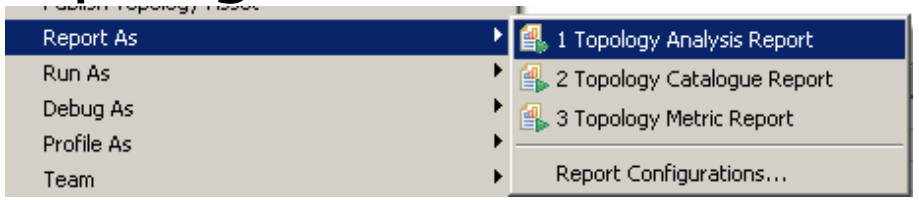
- Reflect UML modeler elements into a deployment topology
  - Components, Actors
- Validate reflected elements
  - Mismatched relationships
  - Mismatched stereotypes
- Quick-fixes to automatically resolve validation errors

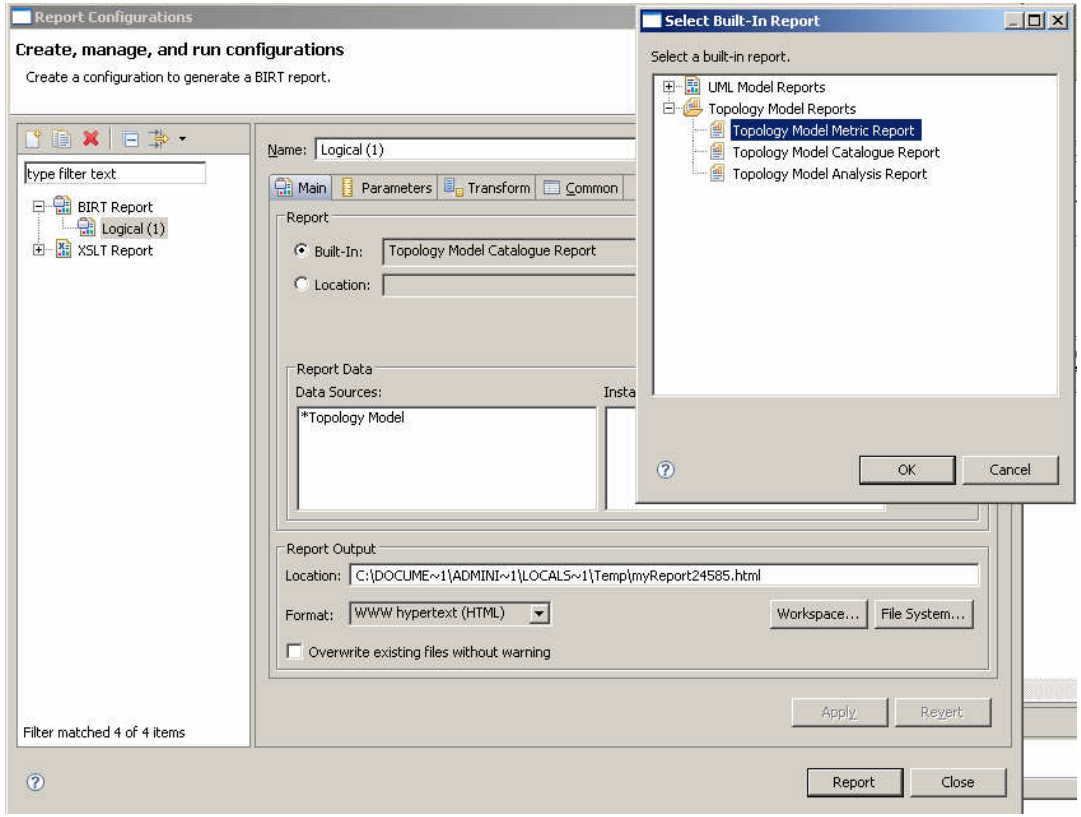




Pre-built Reports for Topologies based on BIRT

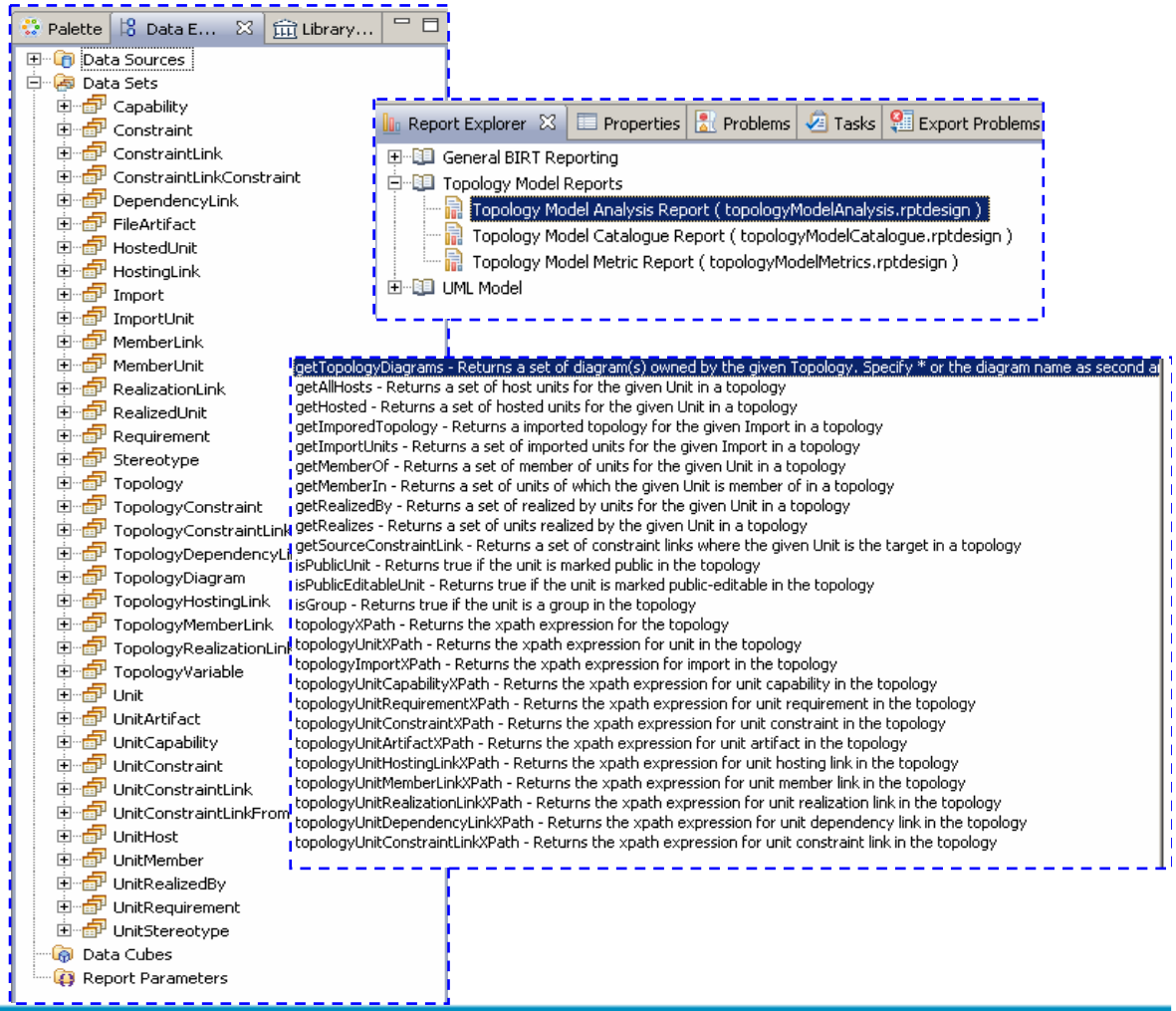
- Quick Shortcuts for pre-built BIRT reports
  - Analysis Reports focus on elements that are traditionally helpful for Operational Modeling
  - Catalogue Reports describe Topology details and relationships
  - Metric Reports provide numerical statistics on the number and kind of Units and relationships within the Topology
- Report Launch Configuration
  - Allow Custom Report based of pre-built designs





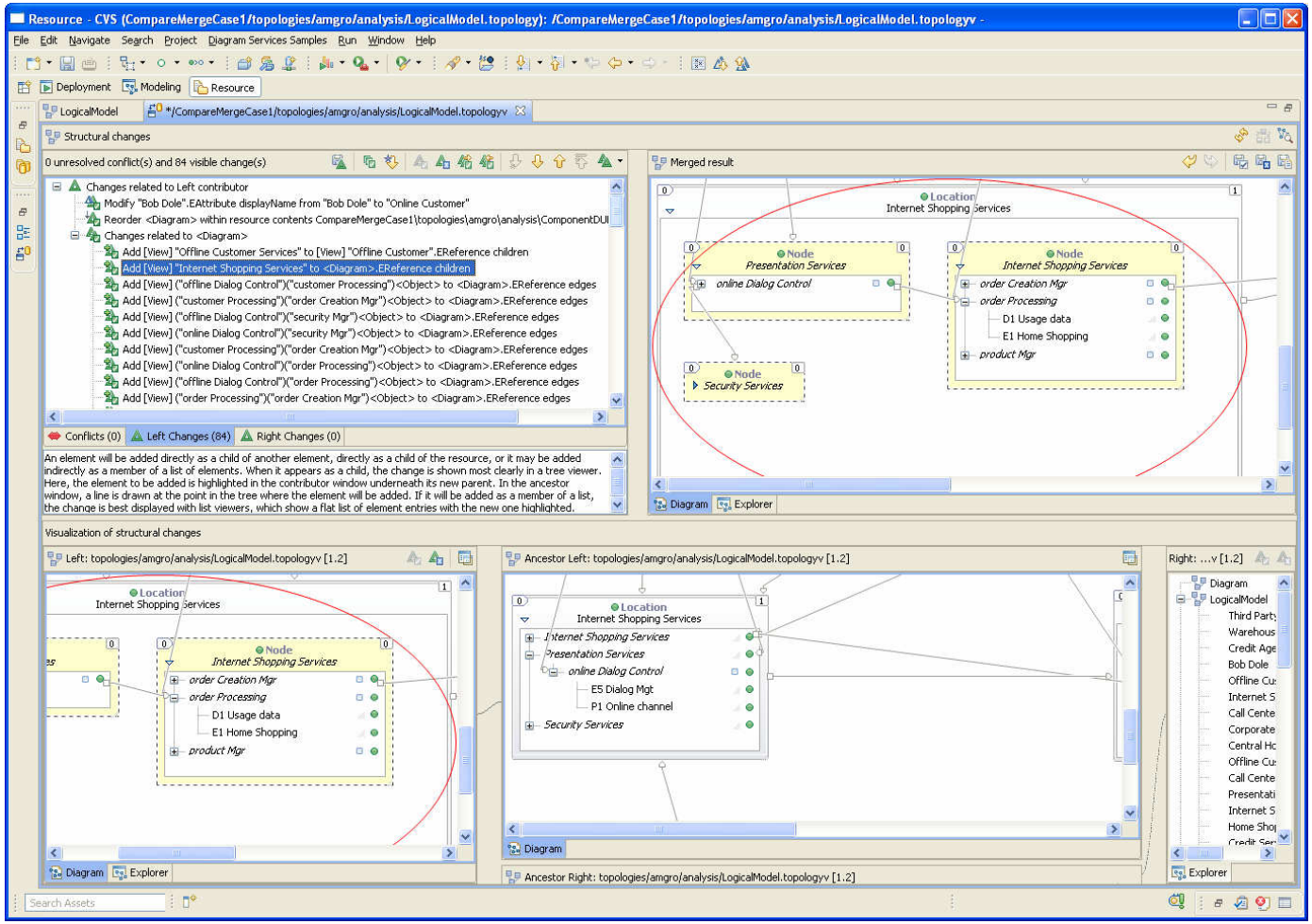
Report Configurability with BIRT Templates

- Report Templates focused on Analysis, Catalogue, and Metrics as with Reports
- Configured Topology Traversal Datasets to make navigating Topologies easy from a Report design
- Content assist for Topology XPath Functions and Expressions



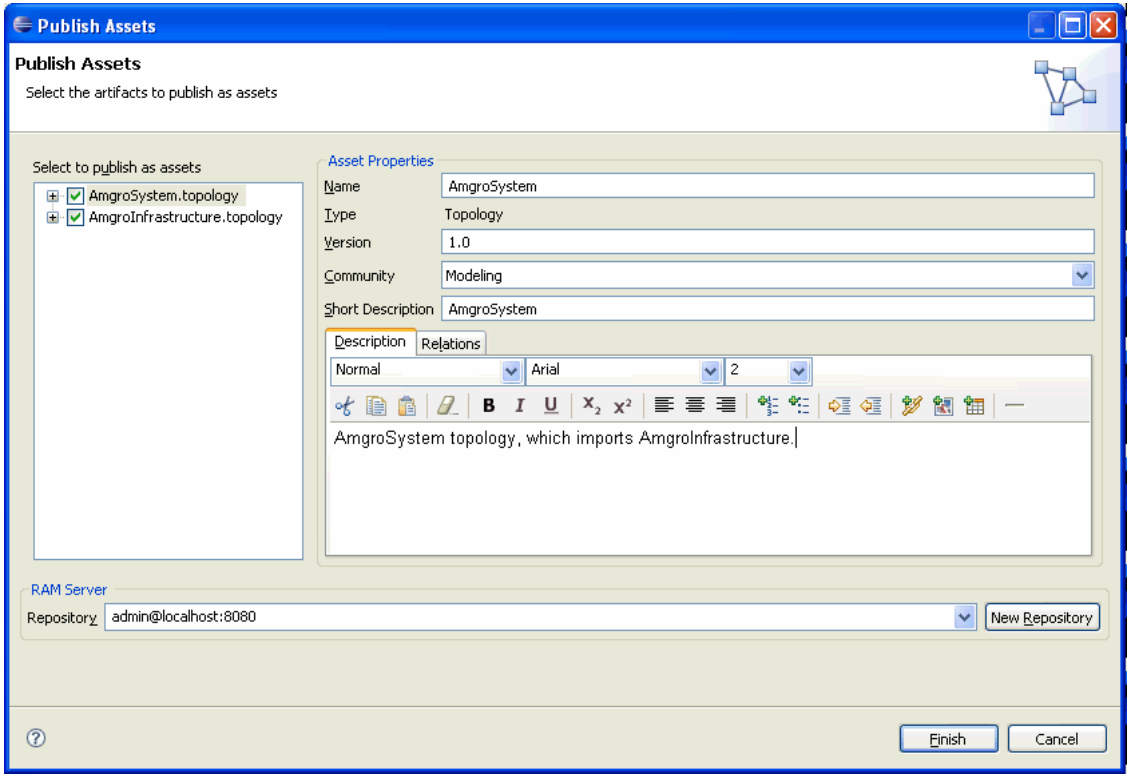
Compare/Merge support for Topologies

- Compare/Merge now supported for Topologies



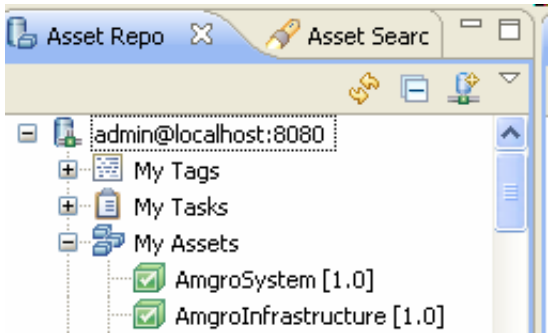
Rational Asset Manager (RAM) Publish Support

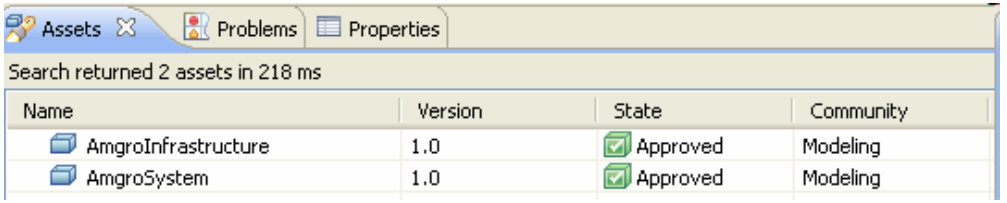
- Enable reuse of topologies through the publish and import of reusable assets
- Topology and Template Topology asset types are available
- Dependent assets and artifacts are captured with the topology
  - model files
  - diagram files
  - imported topology assets
  - palette metadata (templates)

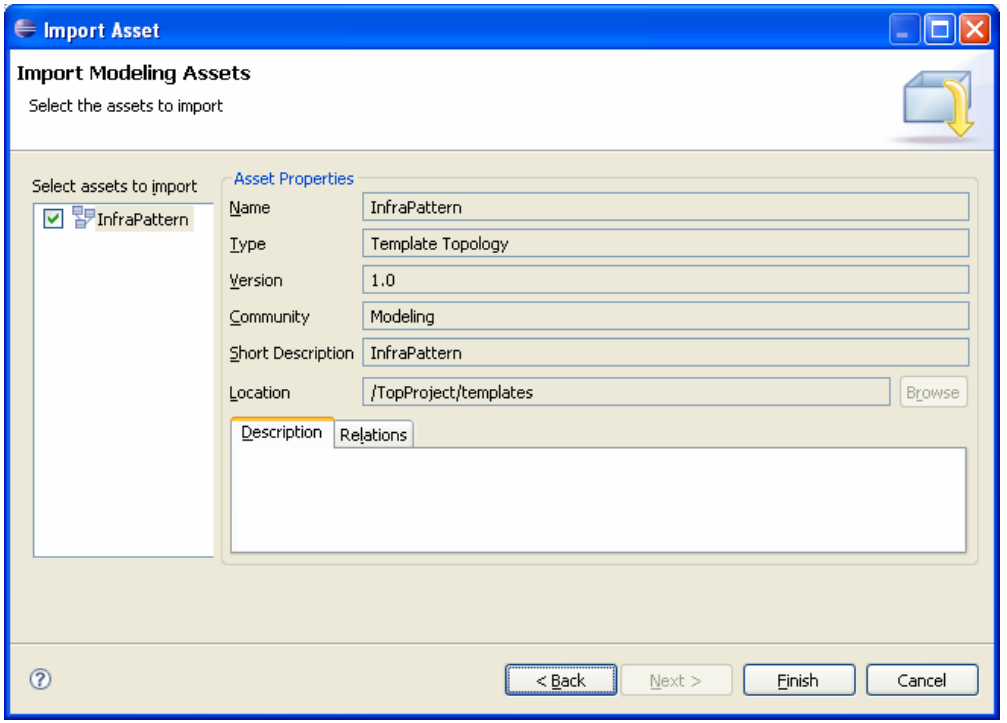


RAM Import Support for Topologies

- Browse and Search available assets
- Extends the RAM integration framework delivered in RSx 7.5

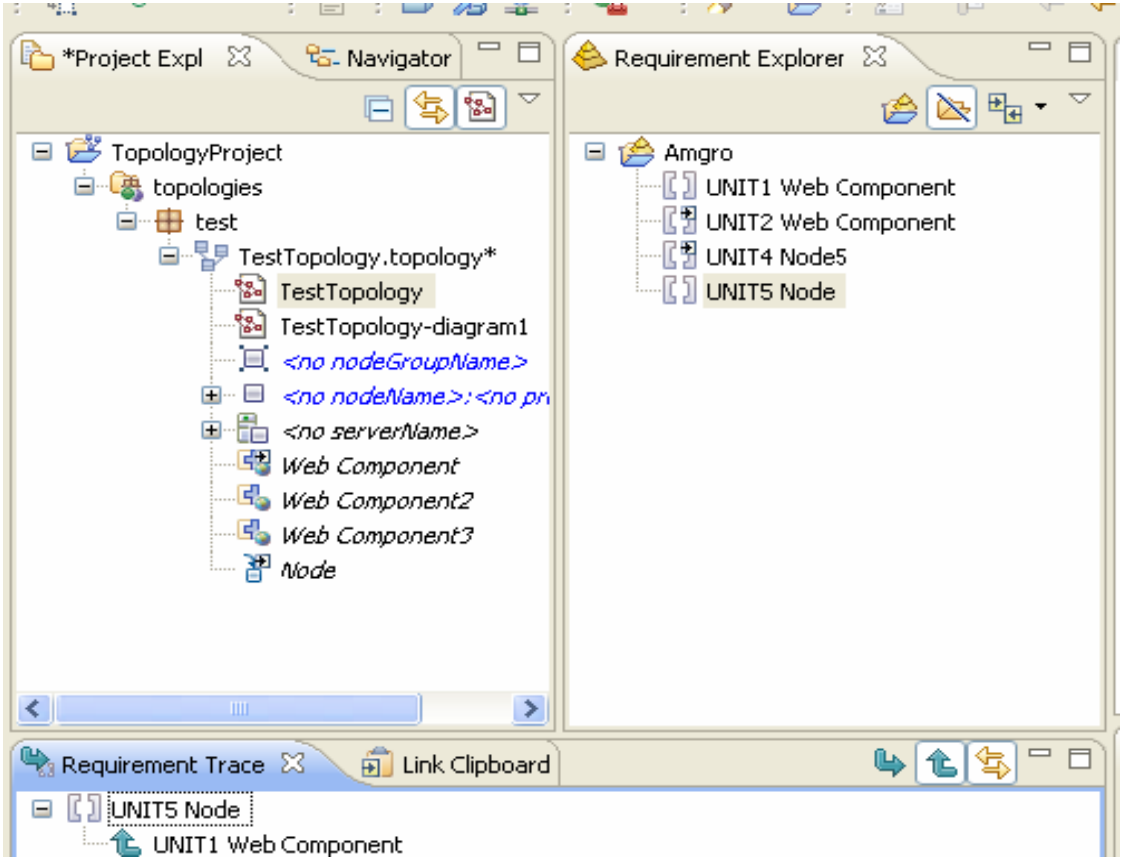






Support for Rational RequisitePro 7.0.1

- RequisitePro business requirements may be linked to Topology elements
- Traceability relationships may be created between requirements
- Linked relationships identified by decorator
- Navigation hooks guide the user between requirements and models including navigation into RequisitePro from linked models or requirements
- Creation support through Link Clipboard, drag and drop, or menu actions.



Requirements created from Topologies in ReqPro

- Link topology elements to existing business requirements
- Navigate quickly to requirements in ReqPro
- Trace relationships may be created from elements in the diagram
- Easily fix lifecycle problems with Requirement Link Problems view
- History and rollback support in Requirement Editor for Topology business requirements