

# Proposal

- Overview:
  - Move WSDM Tooling, WSDM extensions to Managed Agent Explorer (MAX) and WS-Notification provider to the TPTP Notification Broker to COSMOS incubator project
  - Leave base MAX framework and Notification broker in TPTP
- URL for Voting:
  - [https://bugs.eclipse.org/bugs/show\\_bug.cgi?id=206632](https://bugs.eclipse.org/bugs/show_bug.cgi?id=206632)

**WSDM = Web Services Distributed Management, an open standard/specified ratified by OASIS**

# Value Proposition

- WSDM tooling enables rapid creation and testing of management interfaces based on open standards
- WSDM extension to Managed Agent Explorer enables testing and runtime introspection of resources with WSDM based management interface
- WS-Notification provider enables workbench to receive WS-Notification from various resources (both WSDM and non-WSDM)
- All three are related to standards based systems management and are valuable additions to COSMOS

## Release History

- Managed Agent Explorer Tech Preview in TPTP 4.2 (2Q2006), 4.2.1 and 4.2.2
- Managed Agent Explorer GA in TPTP 4.3 (4Q2006), 4.3.1
- WSDM Tooling technology preview in TPTP 4.3 (4Q2006), 4.3.1
- Managed Agent Explorer and Notification Broker GA in TPTP 4.4 (2Q2007)
- WSDM Tooling and WS-Notification provider technology preview in TPTP 4.4 (2Q2007)

## Overview of WSDM Tooling and WSDM extension for Managed Agent eXplorer

WSDM tooling enables the complete build-package-deploy-test cycle for manageability endpoints through artifact editors, validators, code generation, deployment editors and integration for testing with the Managed Agent eXplorer in TPTP.

Targeted towards architects and developers; Allows users to,

- Rapidly define manageability characteristics of the resource

- Assemble capabilities into manageability endpoint

- Generate code for endpoint, proxies and deployment bundles

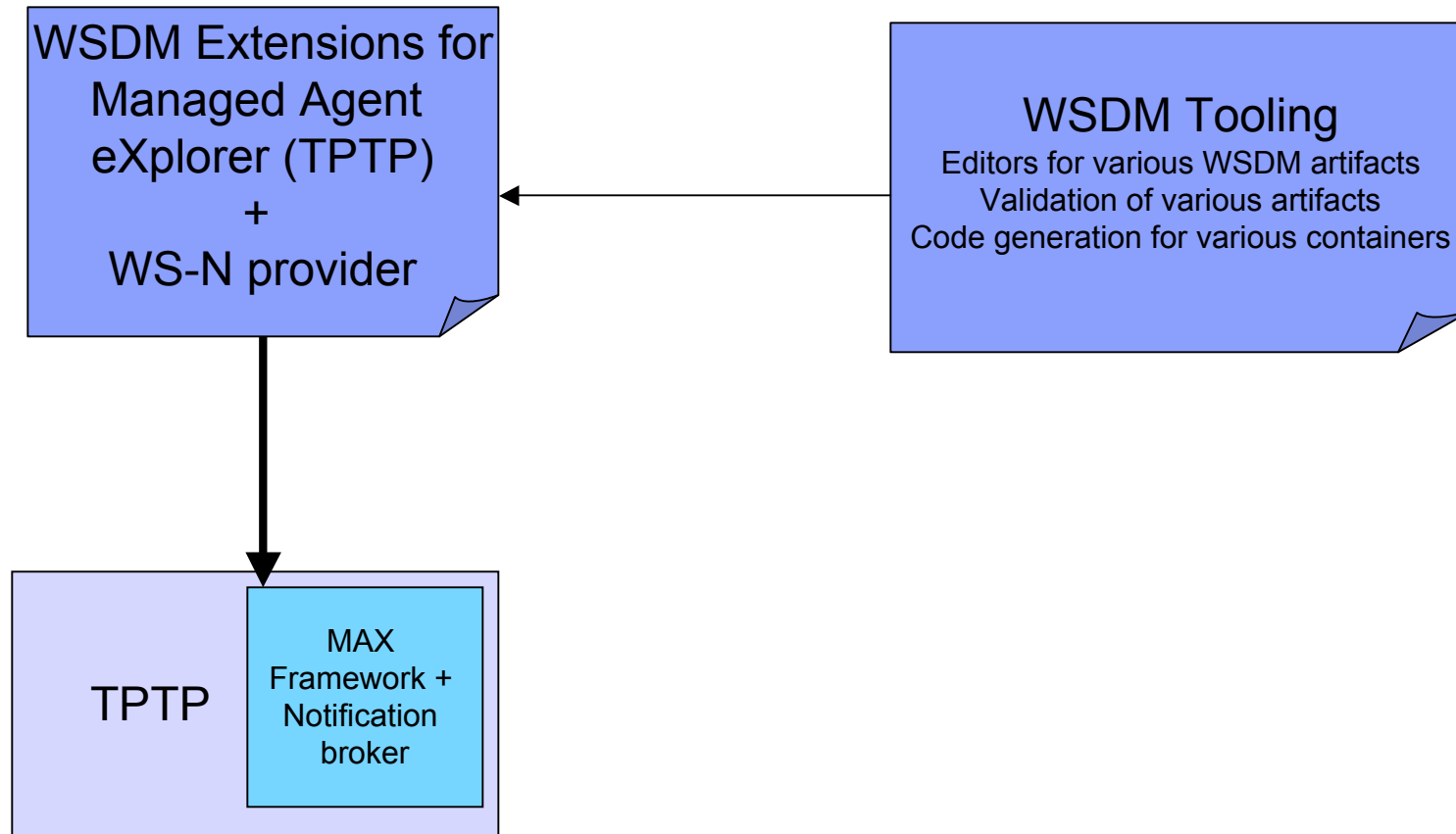
- Test the implementation using WSDM extension to the Managed Agent eXplorer

WSDM extension for the TPTP Managed Agent eXplorer allows introspection and interaction with WSDM endpoints

The WSDM extension also uses the WS-N notification provider to receive property change notifications, service group membership updates and other notifications from WSDM endpoints; any workbench bundle can subscribe to WS-N producers through this provider

## High level overview of proposed contribution :

The following components in the blue boxes are proposed to be moved to COSMOS from TPTP



## **Move related work items**

## Bugzilla changes

Create following components in COSMOS bugzilla:

ManagementEnablement.Tooling.WSDM  
ManagementEnablement.MAXExtensions

Remove the following components from TPTP bugzilla:

Monitor.UI.WSDMTooling

31 features from Monitor.UI.WSDMTooling and 3 features from Monitor.UI.ManagedResourcesExplorer will be moved to ManagementEnablement.Tooling.WSDM and ManagementEnablement.MAXExtensions respectively. 52 Outstanding bugs will be reclassified as well.

See [this slide](#) for details on packages being moved

## Newsgroup and mailing list changes

The components proposed to be moved to COSMOS did not have their own mailing list or newsgroup. All discussions were carried out in the *eclipse.tptp* newsgroup and the *tptp-monitoring-dev* mailing list.

We will post messages on the above newsgroup and mailing list pointing users and developers to the COSMOS *eclipse.technology.cosmos* newsgroup and *cosmos-dev* mailing list

## Website updates

1. Create Management Enablement section on main COSMOS page (<http://eclipse.org/cosmos>)
2. Announce addition of WSDM tooling component to above page
3. Move WSDM related content from Eclipse TPTP Wiki to <http://wiki.eclipse.org/CosmosBuildToManageComponent>
4. Move design documents from Eclipse TPTP Wiki ([http://wiki.eclipse.org/Build\\_to\\_Manage](http://wiki.eclipse.org/Build_to_Manage)) to [http://www.eclipse.org/cosmos/mgm\\_enablement/documents/index.php](http://www.eclipse.org/cosmos/mgm_enablement/documents/index.php)
5. Create new demos / viewlets showcasing WSDM tooling and add to Management Enablement section or the Demo downloads page (<http://www.eclipse.org/cosmos/downloads/?ver=eclispecon07>)
6. Provide link to Apache Muse on Management Enablement main page
7. Add contributors and committers to [http://www.eclipse.org/cosmos/project\\_info/team/](http://www.eclipse.org/cosmos/project_info/team/)
8. Update release plan at [http://wiki.eclipse.org/index.php/Cosmos\\_Release\\_Plan#Release\\_Deliverables](http://wiki.eclipse.org/index.php/Cosmos_Release_Plan#Release_Deliverables)
9. Update TPTP BtM page at [http://www.eclipse.org/tptp/monitoring/resources/download/btm/index\\_4\\_4.html](http://www.eclipse.org/tptp/monitoring/resources/download/btm/index_4_4.html) to point to COSMOS for WSDM tooling
10. Download



# Committers and contributors

## Approved Committer:

- Balan Subramanian (approved by PMC)
  - Committer since April 2006 on Eclipse TPTP project, lead committer for WSDM tooling
  - Commits: 414, Lines of code: 185, 729
  - Contributed 6 features, fixed 104 bugs

## Additional Proposed Contributors:

Saurabh Dravid (Active contributor to Eclipse TPTP project)

- Contributed 3 features, fixed 65 bugs

Hariharan L Narayanan

- Contributed 2 features and fixed 22 bugs in TPTP
- Proposed parttime committer for MAX framework maintenance

## Other Work items

### 1. Move code

Refactor package names under root *org.eclipse.cosmos.me.tooling.wsdm* package

Plugins with EPL code to be checked into COSMOS CVS:

*org.eclipse.tptp.wsdm.editor*  
*org.eclipse.tptp.wsdm.tooling.doc.user* (to be linked to higher level COSMOS TOC if available)  
*org.eclipse.tptp.wsdm.model*  
*org.eclipse.tptp.wsdm.tooling.codegeneration*  
*org.eclipse.tptp.wsdm.tooling.codegeneration.j2ee*  
*org.eclipse.tptp.monitoring.managedagent.wsdm*  
*org.eclipse.tptp.monitoring.managedagent.wsdm.muse*  
*org.eclipse.tptp.monitoring.managedagent.wsdm.doc.user* (remain linked to higher level TPTP TOC)  
*org.eclipse.tptp.monitoring.wsdm.muse.notification*

Plugins with non-EPL JARs and binaries to be checked into COSMOS CVS following parallel IP process or on EMO Legal Approval:

<i>org.apache.muse.api</i>	<i>org.apache.muse.impl</i>
<i>org.apache.muse.tools</i>	<i>org.apache.muse.utils</i>
<i>org.apache.muse.core</i>	<i>org.apache.axis2</i>
<i>org.apache.muse.osgi.core</i>	<i>org.apache.muse.osgi.wrapper.core</i>
<i>org.apache.muse.osgi.wrapper.mini</i>	<i>javax.activation.api</i>
<i>javax.mail.api</i>	<i>javax.saaj.api</i>
<i>javax.stax.api</i>	<i>org.apache.commons.httpclient</i>
<i>org.apache.ws.commons.schema</i>	<i>org.apache.xalan</i>
<i>wstx-asl</i>	

Regenerate Javadoc; Ensure no workspace compile issues

## Other Work items (Continued)

### 2. Build integration

Update COSMOS dependencies to include:

- Eclipse Equinox 3.3 (for building OSGi endpoints, consuming WS-N based notifications)

- Eclipse Web Tools Project 2.0 (for building J2EE endpoints)

- New packages from Eclipse Orbit

Update COSMOS build

- PDE build script used by the WSDM tooling team can be reused

Test the build

### 3. Integrate test suites

Check in following test suite plugins:

- org.eclipse.tptp.wsdm.editor.test

- WSDM extension specific test cases from org.eclipse.tptp.monitoring.managedagent.test

- WS-N provider specific test cases from org.eclipse.tptp.monitoring.notifications.test

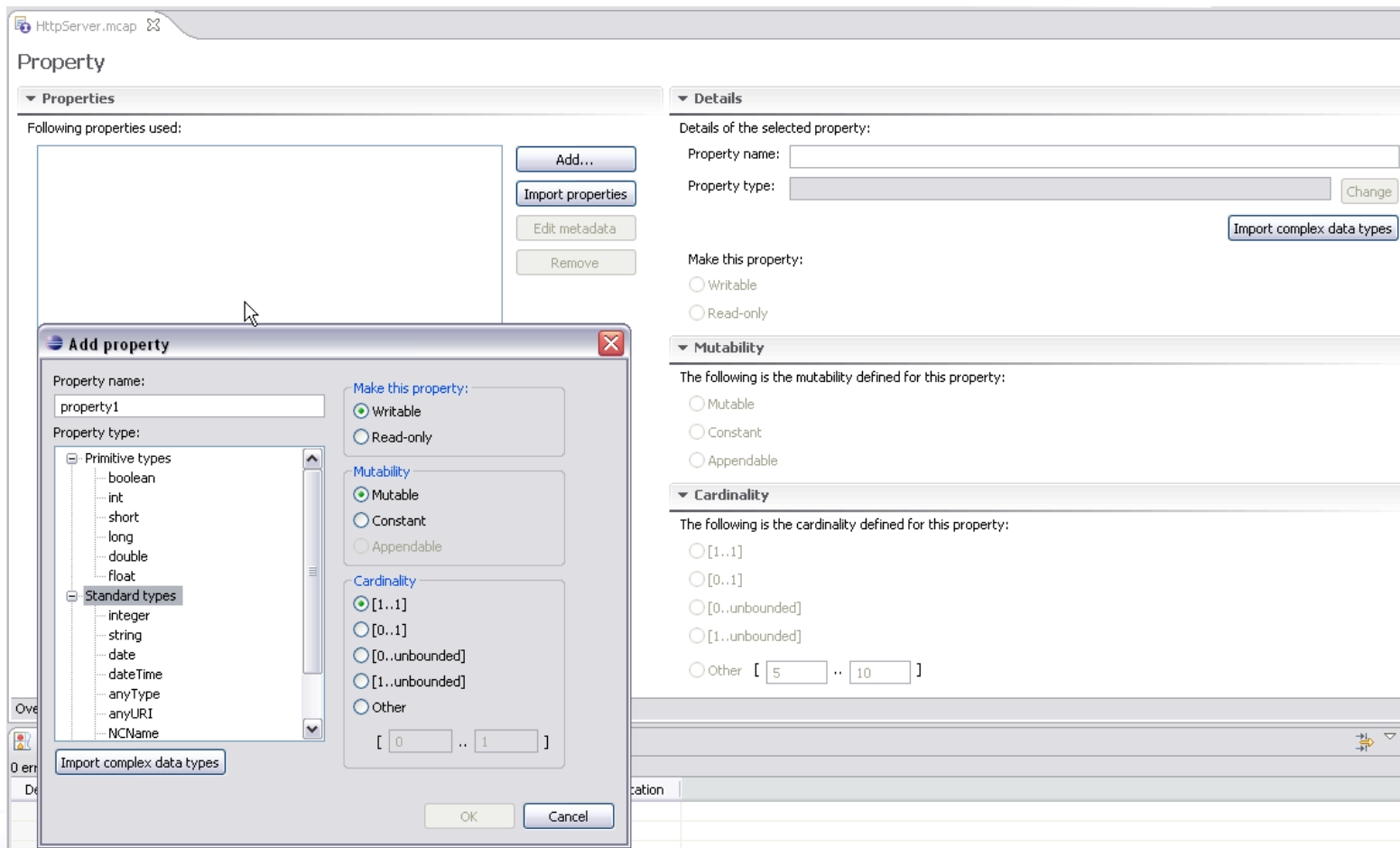
Integrate with top level test suites in COSMOS

Some manual test cases have been automated with the TPTP AGR and will be used for future testing.

## **Functional Details and screenshots**

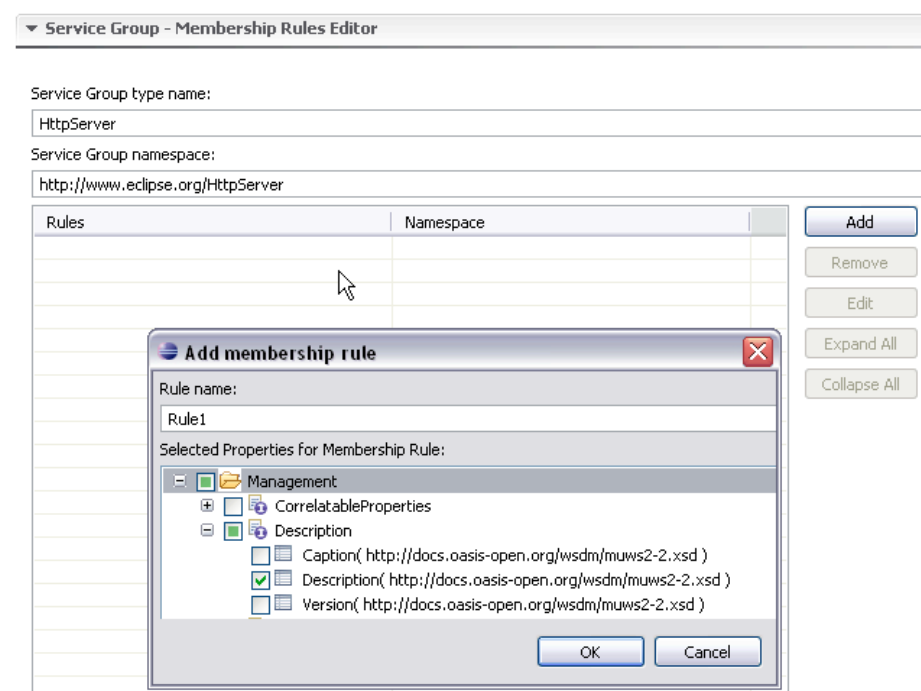
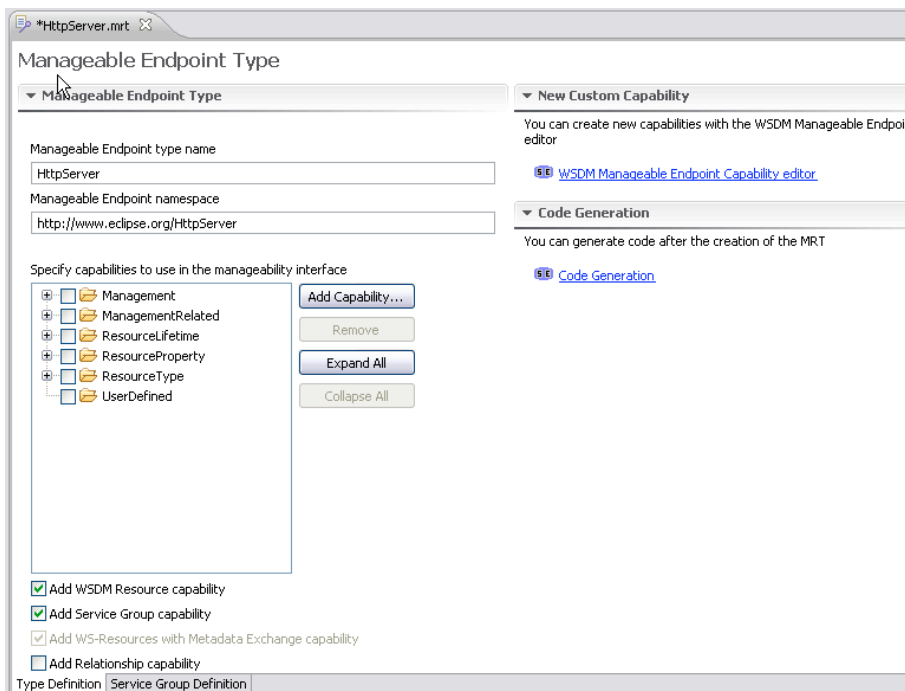
## WSDM Tooling Editors – Capability Editor:

- Define properties, meta-data for properties, import complex types (XSD) for property types, identify metrics, specify metrics settings, import WSDLs and XSD Schemas as capabilities, import Java classes
- Define operations, their parameters, return types and faults; import XSD types for parameters and return
- Define topics in a hierarchy



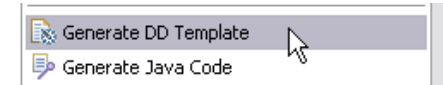
## WSDM Tooling Editors – Manageable Resource Type (MRT) Editor:

- Define different resource types using a combination of standard and custom, user-defined capabilities
- Automatically add standard capabilities for certain types of resources
- For service group resource types, edit the set of membership rules by visually looking at various capabilities both standard and user defined in the workspace
- New capabilities can be added to the list through provided extension point



## WSDM Tooling Editors – Deployment descriptor template editor:

- Create deployment descriptor from scratch or generate from resource definition
- Provide initialization parameters, initial instances, custom serializers for complex types
- Control code generation



HttpServer.dd

**Resource Types**

Following resource types used:

Resource ...	MRT File	WSDL File ...	Context F
HttpServer	HttpServer...	HttpServer...	/HttpServi

Add Edit Delete

Select a resource type to view its parameters:

Parameter Name	Parameter Value
validate-wsrp-schema	false

Add Delete

**Capabilities**

Following capabilities available in selected resource type:

Capability Name	NameSpace URI	Class Name	Persiste
Get	http://docs.oas...	<Generate>	
Identity	http://docs.oas...	<Generate>	
ImmediateResou...	http://docs.oas...	<Generate>	
ManageabilityCh...	http://docs.oas...	<Generate>	
mex	http://schemas...	<Generate>	
NotificationProd...	http://docs.oas...	org.eclipse.tptp...	
Query	http://docs.oas...	<Generate>	
ScheduledResou...	http://docs.oas...	<Generate>	
ServiceGroup	http://docs.oas...	<Generate>	
ServiceGroupRe...	http://docs.oas...	<Generate>	
Set	http://docs.oas...	<Generate>	

Edit

Parameters for the selected capability

Parameter Name	Parameter Value
----------------	-----------------

Add Delete

Overview ResourceTypes Deployment Layout Custom Serializers

## WSDM Tooling – Code generation:

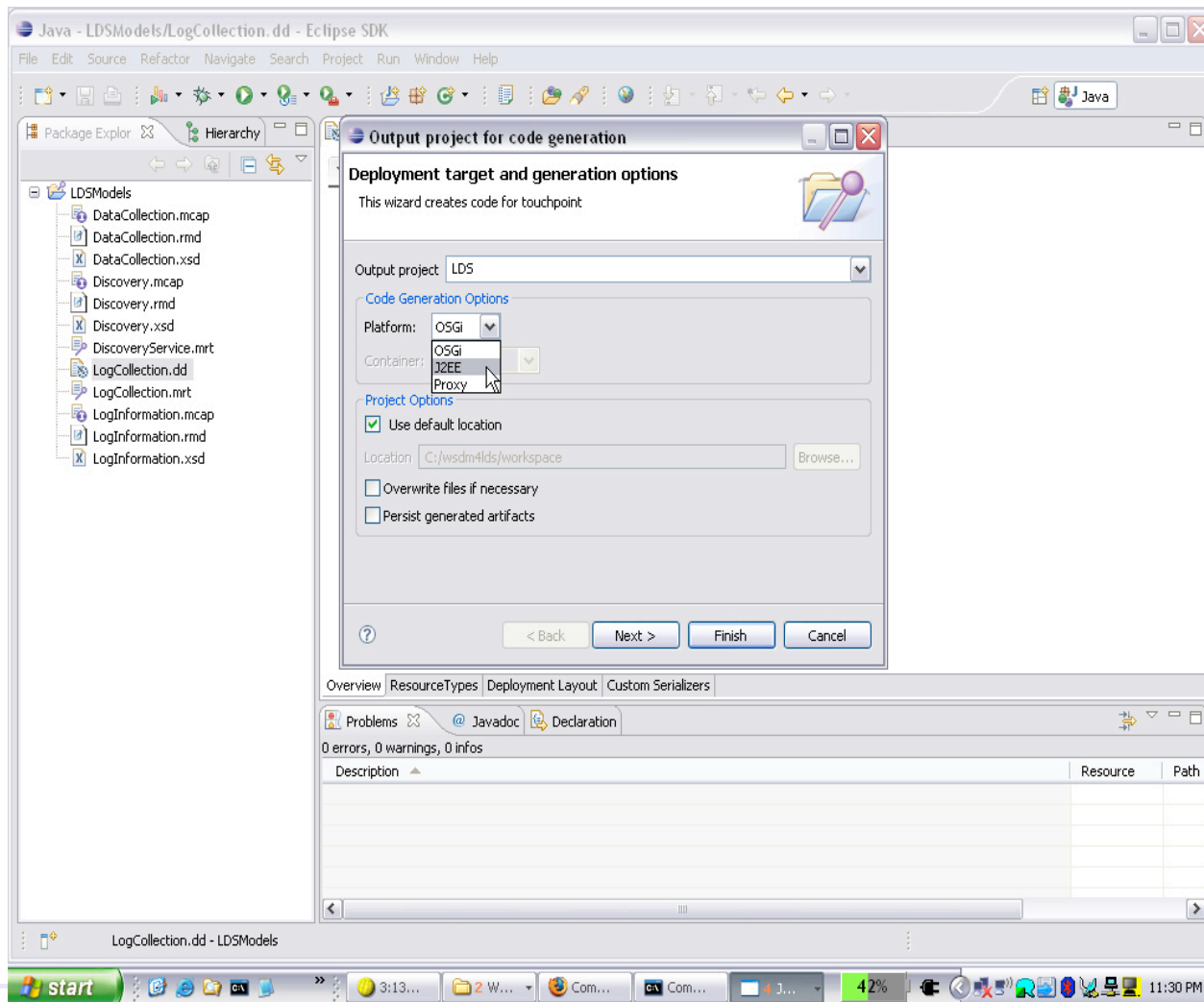
- Create code for OSGi deployment using Axis2 or Muse Mini servlet engine
- Create code for Java Proxies to interact with resources
- Apache Muse code generation invoked to generate all web services artifacts and backend code





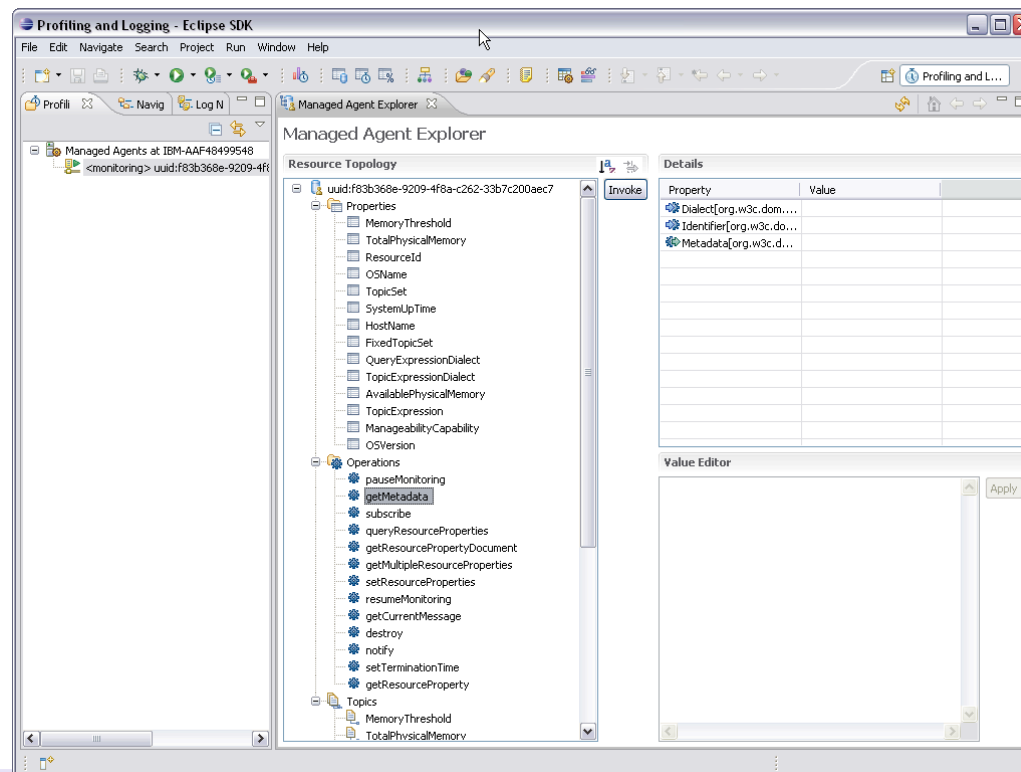
## WSDM Tooling – Code generation for J2EE endpoints:

- Create code for J2EE deployment using Axis2 or Muse Mini servlet
- Uses WTP to “webify” the generated project so that it can be exported to WAR, run on server, validated



## WSDM Extension for MAX

- Connect to different resources by providing a configuration for that particular resource type (EPR)
- View the properties and operations provided by a resource for management; refresh on demand
- Inspect individual properties; Change the property values
- Invoke operations on the resource; ability to key in parameters and view returned results
- View the topology of a group of endpoints in a service group
- Connect directly with any member of such a displayed group and further inspect/interact with it
- View and traverse relationships between WSDM enabled resources
- View topics and subscribe; also subscribe to property change notifications



## WS-N provider

- Allows workbench to subscribe to topics and receive WS-N based notifications
- Any plugin can register with this provider; the provider manages subscriptions intelligently
- MAX registers with this provider for property change notifications and service group membership updates
- Other notifications are currently shown in error log; plan was to integrate with log view in TPTP
- This provider plug into the notification broker in TPTP

