Managing Eclipse Workplaces in the Enterprise

Dietmar Stoll, itemis Schweiz
Outline

Managing Eclipse Workplaces in the Enterprise

- Manual Workplace Provisioning
- Managed Workplace Provisioning
- Architecture
- Enterprise Features
- Developer’s View
- Questions
„Equipment“ for Development Teams working with Eclipse

- IDE
- Embedded Tools
- Target Platform
- Preferences
- Projects and Project Sets
- Task Data
Setting the Eclipse environment

Before getting Sphinx, the first step consists in setting the required Eclipse environment. The principle consists in using a minimal environment for the edition, and an adapted target platform to compile and run Sphinx.

Install Eclipse
No comments.

Install a Subversion client
We recommend the use of Subversion. Subversive may be installed from the following update sites:

- http://download.eclipse.org/technology/subversion/0.7/update-site/
- http://www.polarion.org/projects/subversive/download/eclipse/2.0/update-site/

Install the following feature groups:
- Subversive SVN Team Provider Plugin (Incubation)
- Subversive SVN Connectors

SVN checkout

SVN repository configuration
Create a new SVN repository with the following attributes:

- URL: https://dev.eclipse.org/svnroot/modeling/org.eclipse.m2d.sphinx (caution: https, not http)
- User: <empty>
- Password: <empty>

Import the project set
Sphinx provides a project set that lists all the Sphinx plug-ins/projects:

- Browse the SVN repository
- Check-out the project trunk/tools/org.eclipse.sphinx.psf

Import the sources
Once the project set is downloaded, it is easy to check-out the whole Sphinx project sources.

- Browse the org.eclipse.sphinx.psf project in your workspace
- Right-click on file all-subversive.psf
- Click on Import Project Set.

All the projects of Sphinx are then checked out in your workspace. They are automatically organized into several working sets.

Manually defined by one or a few ...
Manually applied by everyone ...
Manual Workplace Provisioning

Project lead

defines a set of

authors

Project Wiki

IDE, plugins, features, etc.

SCM, web server, etc.

apply to

use

Project team members

read
Manual Workplace Provisioning

Summary

- **Manual** setup procedures, documented in wikis (maybe old and incomplete)
- **Time-consuming** Eclipse installations and configurations
- „damaged“ Eclipse installations, increased **support** effort
- **Uncontrolled proliferation** of Eclipse distributions and plugins
- **Knowledge concentrated on a few people** - support
Solution Suggestion

„Manually defined by one person, automatically applied to everyone!“
Workplace Profile – „p3“ Profile

Project specific settings and initialisation of workspaces

- P3 Profile
  - Required IDE plugins/features
  - Embedded tools
  - Target platform
- Workspace preferences
- Projects and project sets from SCM
- Mylyn task repository configurations
- Eclipse Help
- Profile scope
  - per project, per team, per role
p3 at a Glance

One-click provisioning of custom Eclipse packages and workspace setups

1) **Download**, unzip and launch Eclipse IDE
   - Eclipse Classic SDK + p3 workspace provisioning tools
   - May include embedded tools like JDK, Maven, etc.

2) **Select a profile** (one-click)

3) **Automatic provisioning** of profile artifacts

4) **Automatic update** of profile artifacts at every restart of the Eclipse IDE
Eclipse p3

Architecture

p3 profile definition on

SCM

authors and provides

builds

p3 profile build on

Build server

Project lead

Project team members

deployed p3 profile on

Web server

builds

Eclipse distro with a few p3 plugins

retrieves from

use

p3 profile build on

Build server

Project lead

Web server

Apache
Outline
Managing Eclipse Workplaces in the Enterprise

- Manual Workplace Provisioning
- Managed Workplace Provisioning
- Architecture
- Enterprise Features
- Developer‘s View
- Questions
Continuous Conformance

- **Automated update of profile artifacts / restriction enforcement**
  - Check on every Eclipse restart
  - Automatic de-installation of unwanted plugins
  - Automatic update / override of Eclipse settings
  - Automatic checkout of projects into the workspace
  - Update of existing plugin configurations (e.g. Mylyn, JIRA)
  - Extensible according to customer’s needs
Standards

Provisioning artifacts may be maintained without p3

- Boot-strapped from **lightweight Eclipse classic**
  - Standard Eclipse branding by default, custom branding may be added if desired
  - Standard Eclipse file and folder structure
  - Standard Eclipse UI (Welcome page, perspectives, views etc.)
Standards

Artifacts needed for provisioning may be maintained without p3

- Reuse of **standard Eclipse formats** and editors
  - b3 update site aggregator models (*.b3aggr files) and p2 update sites
  - PDE Target definitions (*.target files)
  - Eclipse preferences (*.epf files)
  - Team project sets (*.psf files)
  - Mylyn WikiText (*.textile files)
  - Ant and Maven builds (build.xml, *.properties, pom.xml files)
- Planned: **Xtext-based profile editor, profile definition generator**
Teams and Profiles

- **Multi-team support**
  - **Master profile** for implementing cross-team development standards
  - **Project profiles** for enabling team-private customizations/add-ons

- **Profile inheritance**
  - Project profiles inherit content from master profile
  - May add/override individual items

- **Two-level profile management**
  - Central rollout of cross-team updates/changes through master profile
  - Central rollout of team-private updates/changes through project profile
Inline Documentation

- Customizable profile user documentation provided as HTML and Eclipse Help
  - Getting started guide
  - IDE and target platform description
  - How-to guidelines
Deployment – Cloud Inside the Company

- **Internal Build and Provisioning System**
  - Aggregated update sites for IDEs and target platforms
  - Zip files for workspace preferences, project sets and Mylyn task data
  - Served from company-internal profile webserver (e.g. Apache httpd)

- **Advantages**
  - No internet connection needed
  - High performance, everything inside the enterprise
  - Independence from external sites
  - Optional: Mirroring of the enterprise provisioning systems
  - Enables provisioning across world-wide distributed sites
Deployment Strategies

Non-intrusive

- Required server-side infrastructure
  - SCM repository (e.g., Git, SVN, CVS, etc.)
  - Continuous integration (CI) server (e.g., Jenkins, Hudson, Bamboo, etc.)
  - Web server (e.g., Apache httpd, Microsoft IIS, etc.)

- Supported deployment strategies
  a) Setup of dedicated SCM, CI and web server instances
  b) Seamless integration into existing server-side infrastructure
Enterprise Features Summary

- Continuous updates / restriction enforcement
- Profile inheritance (master / project)
- Standard Eclipse formats
- Internal “cloud”
- Non-intrusive (uses existing infrastructure)
Developer’s View
Developer’s View

Eclipse SDK stores your projects in a folder called a workspace. Choose a workspace folder to use for this session.

Workspace: /home/john/dev/CDO-example

Use this as the default and do not ask again

Cancel OK
Developer’s View

Workspace Launcher

Select a workspace
Eclipse SDK stores your projects in a folder called a workspace. Choose a workspace folder to use for this session.

Workspace: /home/john/dev/CDO-example

Use this as the default and do not ask again

Cancel OK

Workspace Provisioning

Choose a provisioning profile to use for this workspace.

Profile: itemis-p3-example

Cancel OK
Developer's View

[Image of Eclipse IDE showing the IDE Manager Provisioning IDE dialog box, which is computing a prerequisite plan and showing other ongoing tasks such as loading target platform and importing project sets.]
Developer's View

![Eclipse IDE interface showing Xtext editor with DomainModel.xtext file open. The editor displays code and structure of the domain model, with elements like `AbstractElement`, `DomainModel`, `Entity`, `Import`, and a structure diagram showing relationships between these elements.](image-url)
Profile Authoring: Target Definition, Preferences

Target Definition

Target Name
Enter a name for this target.
itemis p3 Master Target 3.7

Locations
The following list of locations will be used to collect plug-ins for this target definition.

https://projects.itemis.de/html/itemis-p3-site/profiles/master/targets/updates/3.7

Show location content

file_export_version=3.0
/instance/org.eclipse.jdt.ui/cleanup.add_default_serial_version_id=true
/instance/org.eclipse.jdt.ui/cleanup.add_generated_serial_version_id=false
/instance/org.eclipse.jdt.ui/cleanup.add_missing_annotations=true
/instance/org.eclipse.jdt.ui/cleanup.add_missing_deprecated_annotations=true
/instance/org.eclipse.jdt.ui/cleanup.add_missing_methods=false
/instance/org.eclipse.jdt.ui/cleanup.add_missing_nls_tags=false
/instance/org.eclipse.jdt.ui/cleanup.add_missing_override_annotestions=true
/instance/org.eclipse.jdt.ui/cleanup.add_missing_override_annotations_interface_methods=true
/instance/org.eclipse.jdt.ui/cleanup.add_serial_version_id=false
/instance/org.eclipse.jdt.ui/cleanup.always_use_blocks=true
/instance/org.eclipse.jdt.ui/cleanup.always_use_parentheses_in_expressions=false
/instance/org.eclipse.jdt.ui/cleanup.always_use_this_for_non_static_field_access=false
/instance/org.eclipse.jdt.ui/cleanup.always_use_this_for_non_static_method_access=false
Profile Authoring: b3 Update Site Editor
Profile Authoring: b3 Update Site Editor

- Custom Category: Eclipse
  - Feature: org.eclipse.sdk / 3.7.0 (Eclipse Project SDK)
  - Feature: org.eclipse.pde.visualization.feature / 1.0.0 (PDE Visualization)
- Custom Category: Application lifecycle management (ALM) tools
  - Feature: org.eclipse.egit / 1.3.0 (Eclipse EGit)
    - Available Versions
      - 2.1.0.201209190230-r
      - 1.3.0.201202151440-r (from another Repository in this Contribution)
      - 1.1.0.201109151100-r (from another Repository in this Contribution)
      - 1.0.0.201106090707-r (from another Repository in this Contribution)
  - Feature: org.eclipse.mylyn_feature / 3.8.0 (Mylyn Task List)
  - Feature: org.eclipse.mylyn.bugzilla_feature / 3.8.0 (Mylyn Tasks Connector: Bugzilla)
  - Feature: org.eclipse.mylyn.trac_feature / 3.8.0 (Mylyn Tasks Connector: Trac)
  - Feature: org.eclipse.egit.mylyn / 1.3.0 (EGit Mylyn)
  - Feature: org.eclipse.mylyn.versions / 1.0.0 (Mylyn Versions)
  - Feature: org.eclipse.mylyn.builds / 1.0.0 (Mylyn Builds)

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Categories</td>
<td>Custom Category: Application lifecycle m...</td>
</tr>
<tr>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>Enabled</td>
<td>true</td>
</tr>
<tr>
<td>Name</td>
<td>org.eclipse.egit.feature.group</td>
</tr>
<tr>
<td>Valid Configurations</td>
<td></td>
</tr>
<tr>
<td>Version Range</td>
<td>1.3.0</td>
</tr>
</tbody>
</table>