

ECLIPSE APPLICATION PLATFORM (RCP 4.8)

<p>Objectives</p> <ul style="list-style-type: none"> ✓ Understanding E4/OSGi architecture ✓ Building SWT/JFace UI ✓ Designing an Eclipse 4 based application ✓ Mastering application distribution 	<p>Target Audience</p> <ul style="list-style-type: none"> ✓ Developers ✓ Project managers ✓ Software Architects 	<p>Pre-requisites</p> <ul style="list-style-type: none"> ✓ Knowledge of Java 	 <p>Duration 4 days</p>
--	---	--	--

Eclipse Architecture

- OSGi Architecture : bundles, fragments, MANIFEST.MF
- Eclipse Architecture : plugins, fragments, features, RCP, e4
- Extensions and extension points, plugin.xml file
- Development tools : the PDE
- Eclipse 4 architecture introduction (EAP)
- Eclipse 3.x vs Eclipse 4.x

Application model

- Application model contents
- Editor for application model and the model spy
- UI structure definition
- Application fragments and processors
- Best practices to design your model

Standard Widget Toolkit (SWT)

- Simple widgets and containers (Button, Text, Group, ...)
- Advanced widgets : Tree, Table, ...
- Position management : Layouts et LayoutData
- Resources management : Font, Color, Images
- Event management : Events et Listeners
- Window Builder to design your E4 UI

JFace Library

- Introduction to JFace
- Viewers and the MVC pattern
- Input, ContentProvider and LabelProvider
- Registries : colors, fonts and images management
- Widgets to help in writing preference pages

Injection and annotations

- Injection mechanism overview
- Specific annotations for injection
- Computed annotations (preference, event)
- Dynamic instantiation

Injection contexts

- Context hierarchy
- Updating contexts
- Using the context spy to display contexts

Behavior annotations

- View and editors annotations
- Command handlers annotations
- Preferences annotations

Dialogs

- Preferences management
- Properties and wizards

E4 services

- Model component management
- Part and selection services
- Event Broker, event management and event spy
- Adapters and loggers
- Other services

Standards APIs

- Drag and drop and copy/paste
- Jobs and UIJobs
- Actions, operations and commands, undo/redo.
- Resources and workspace

Look and Feel, CSS

- Modifying code to manage CSS
- Writing the CSS, syntax, editor and the CSS Spy
- Binding the CSS in the application

Defining a new extension point

- Principle, design, definition, schema, parameters
- Java coding and extensions of extension point

Industrialization and delivery

- Feature, product, p2 repository
- Tests management (core and UI)
- Manual delivery (for multi platform using delta pack)
- Introduction to automatic delivery tools (maven tycho)

Exercises

- About fifteen exercises illustrating the concepts on the subject of a rental agency, developed using the latest Eclipse version (Photon 4.8) and managed with git.