

<p><b>Objectifs</b></p> <ul style="list-style-type: none"> <li>✓ Comprendre l'architecture 4.X</li> <li>✓ Etudier la portabilité 3.X-&gt;4.X</li> <li>✓ Bonnes pratiques de développement</li> </ul>	<p><b>Public</b></p> <ul style="list-style-type: none"> <li>✓ Développeur RCP</li> <li>✓ Chef de projet</li> </ul>	<p><b>Pré-requis</b></p> <ul style="list-style-type: none"> <li>✓ Avoir suivi la formation 3.X</li> <li>✓ ou avoir une expérience significative de développement sur l'architecture 3.X</li> </ul>	 <p>Durée 2 jours</p>
--	--	--	--

### Architecture Eclipse 4

- Présentation de l'architecture Eclipse 4 (EAP)
- Comparaison avec Eclipse 3.X

### Le modèle d'application

- Contenu du modèle d'application
- L'éditeur de modèle et le model spy
- La définition de la structure de l'IHM
- Création de parts (vues et éditeurs)
- Création de commandes
- Les bonnes pratiques de conception

### L'injection et les annotations

- Description du mécanisme d'injection eclipse 4
- Les annotations spécifiques de l'injection
- Les annotations complémentaires (préférence, event)
- Dynamique d'instanciation

### Les contextes d'injection

- Hiérarchie des contextes d'injection
- Mettre à jour un contexte d'injection
- Visualiser les contextes avec le context spy

### Les annotations de comportement

- Gestion des vues et des éditeurs
- Gestion des handlers de commande

### Addons

- Rôle des addons
- Utilisation pour l'initialisation et le portage
- Gestion des registries Jface

### Dialogues

- Gestion des préférences
- Gestion des property et des wizards
- Problématiques de portage

### Les fragments et les processors

- Définition des fragments de modèle d'application
- Définition d'un processor pour enrichir un modèle
- Intégration dans une architecture Eclipse

### Les services E4

- Le ModelService et la mise à jour du modèle
- Le PartService pour gérer l'IHM
- Le SelectionService et la gestion de la sélection
- L'EventBroker, la gestion des événements et l'event spy
- Les autres services

### Look and Feel, CSS

- Modification du code pour gérer les CSS
- Écriture de la CSS, syntaxe et éditeur et CSS spy
- Intégration de la CSS dans l'application
- Modification de la CSS au runtime

### Couche de compatibilité

- Rôle de la couche de compatibilité
- Mise en œuvre dans une application
- Migration progressive vers une application pure 4.X
- Ajout de composants 4.X en utilisant les fragments

### Application Eclipse 4.X

- Création d'une application pure 4.X
- Target platform 4
- Utilisation des spies pour la mise au point
- Règles de développement
- Règles de portabilité Eclipse 3.X → Eclipse 4.X
- Le portage d'une application, outils d'aide au portage

La formation est illustrée d'exercices de migration entre une application basée sur Eclipse 3.X vers une application avec couche de compatibilité, puis purement 4.X en utilisant la version Oxygen 4.7.

L'historique des évolutions des exercices est également stocké sous git