# Eclipse IDE



Meeting (22.11.2016)

Robert, Stefan, David







## Agenda





#### TitleShort

AuthorShort

### Teil 1

Integration in Eclipse





## Integration in Eclipse

#### TitleShort

AuthorShort

 Plugins können an vordefinierten Extension Points die Funktionalität von Eclipse erweitern





#### TitleShort

AuthorShort

### Teil 2

### Architektur





#### TitleShort

AuthorShort

Bild





#### TitleShort

AuthorShort

- Definiert Extension Points
- Basiert auf OSGi Framework
- Verwaltet installierte Plugins
- Aktiviert diese, wenn sie benötigt werden





## Workspace

TitleShort

Author Short

- Definiert API, um Ressourcen zu erstellen und zu verwalten
- Ressourcen sind bspw.: Prjekte, Dateien, Ordner
- Werden im Dateisystem gespeichert





### Workbench UI

- Extension Points um UI Komponenten hinzuzufügen
- Toolkits **JFace** und **SWT** zur Erstellung von Uls





## Help

TitleShort

AuthorShort

 Definiert Extension Points um Hilfe und Dokumentationen anzuzeigen





### Team

TitleShort

AuthorShort

■ Definiert Model für das Management und die Versionierung von Ressourcen





## Debug

TitleShort

AuthorShort

■ Definiert sprachunabhängiges Model zum Debugging und UI zum Erstellen von Debuggern





### **Utilities**

- Verschiedene andere Funktionalität:
  - Suchen und Ersetzen
  - XML Konfiguration
  - Zugriff auf Server





#### TitleShort

AuthorShort

### Teil 3

### **Extension Points**





#### **Extension Points**

#### Commands

- Identifizierung durch ID
- Werden durch Menüs und Tastenkombinationen ausgeführt
- Definieren lediglich Verhalten
- Implementiert durch Handler
  - verschiedene Plugins können dasselbe Command implementieren
- org.eclipse.ui.commands





### **Extension Points**

#### Handler

- Implementiert das Verhalten eines Commands
- Workbench entscheidet, welcher Handler ein Command behandelt
- org.eclipse.ui.handlers





### **Extension Points**

## Key Binding

- Assoziation zwischen Command und Tastenkombination
- Tastenkombinationen haben einen Kontext in dem sie aktiv sind
- org.eclipse.ui.bindings





#### TitleShort

AuthorShort

## Teil 4

Workspace UI





TitleShort

AuthorShort

BILD





#### Seiten

- Eine Seite (Page) kann veschiedene Komponenten enthalten
- Pages müssen typischerweise nicht programmiert werden





## Perspektive

TitleShort

AuthorShort

■ Eine **Perspektive** definiert eine der Aufgabe angemessene Menge von Views, deren Layout und ausführbare Aktionen





#### View

- Eine **View** wird verwendet, um durch eine Informationshierarchie zu navigieren, Editoren zu öffnen oder Einstellungen des Editors zu bearbeiten
- org.eclipse.ui.views





### Editor

- Ein Editor wird verwendet, um ein Dokument oder ein anderes Objekt anzuzeigen und zu bearbeiten
- org.eclipse.ui.editors





### **Preferences**

- Plugins können Einträge zur Baumstruktur der Einstellungen hinzufügen
- org.eclipse.ui.preferencePages





#### Tool Bar und Menu Bar

- Menu Bar: die globale Tool Bar von Eclipse
- Weitere Tool Bars und Kontextmenüs können erstellt werden
- Einträge führen Commands aus oder beinhalten weitere Menüs
- org.eclipse.ui.menus





## Dialoge und Wizards

TitleShort

AuthorShort

#### ■ Dialoge:

- Anzeige von Meldungen (z.B. Fehlern)
- Bestätigung einer Aktion
- Eingabe von Werten

#### Wizard:

- Geführte Eingabe von Daten in einer Sequenz verschiedener Formulare (z.B. Erstellen eines Projekts)
- Besteht aus Wizard Dialog und mehreren Wizard Pages
- Vordefinierte Extension Points existieren für Wizards zum Erstellen, Importieren und Exportieren von Obk





### Literature

#### TitleShort

AuthorShort

