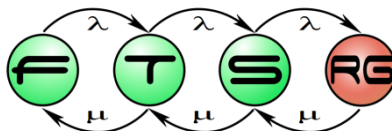


# Plug-in fejlesztés



# Plug-in architektúra

# Plug-in rendszer

- Contribution rule
  - „everything is a contribution”
- Az Eclipse minden eleme egy plug-in
  - Nincsenek „bevasalt” elemek
    - Sok plug-in (kb. 60 nagyobb az alap SDK-ban)
    - Még több lehetséges bővítés
      - (pl. WebSphere Application Developer fejlesztőkörnyezet kb. 500 plug-in)

# Plug-in rendszer

- Plug-in kezelés
  - Ha több ezer plug-int kell kezelni, kérdéses a teljesítmény
  - Jó lenne csak az éppen használtakkal törődni
- Lazy loading rule
  - „Contributions are only loaded when they are needed.”

# Plug-in kezelés

## ■ Probléma

- Lazy loading vagy
- Plug-in információk állandó elérése?

## ■ Megoldás

- Deklaráció/implementáció szétválasztása
  - Csak a plug-inok jellemzőit olvassuk be
    - Leíró fájlok (manifest.mf, plugin.xml)
  - Implementáció lusta betöltése
    - Java kód

# Plug-in projekt

- Kiterjesztések plug-in projektbe csoportosíthatóak
  - Hasonló Java projektekhez
  - Összetartozó kiterjesztések együttesen
- Terjesztés: jar archívumokban

# Plug-in projekt felépítése

- Deklaráció
  - manifest.mf
    - Név, verziószám, függőségek definíciója
  - plugin.xml
    - A plug-inek további adatai adhatók itt meg
    - Tipikus tartalma: kiterjesztések és kiterjesztési pontok
    - Hiányozhat
- Implementáció
  - Java kód
    - A plug-in megvalósítása
    - jar formátumban
- Erőforrások

# manifest.mf

- Open Services Gateway Initiative (OSGi) szabvány szerint
  - OSGi modulok neve: bundle
- Plug-in vs Bundle
  - Eclipse plug-in projekt egyben OSGi bundle is
- Eclipse 3.0 óta használható



# manifest.mf tartalma

- Plugin ID
  - A plug-in egyedi azonosítására szolgál - kötelező
- Plugin name
  - Név, megjelenik a felhasználói felületen - kötelező
- Verzió
  - Elvileg tetszőleges string formátum lehetséges
  - Kötelező, függőségek megadásához használható
- Provider
  - A plug-in készítőjének neve - opcionális
- Függőségek

# Verziószámok megadása

- Verziók összevetése: stringek összehasonlítása
- Ajánlott formátum: x.y.z.qualifier
  - x: főverzió, visszafelé kompatibilitás hiányát jelzi
  - y: mellékverzió, visszafelé kompatibilis frissítést jelez
  - z: alverzió, kívülről nem észrevehető módosítások (pl. bugfix)
  - qualifier: szokás szerint tartalmazza a készítés dátumát, valamint a kiadásra utaló jelzést (nightly, integration, milestone vagy release).
- Példa:
  - A legutolsó 3.4-es sorozatbeli kiadás verziószáma:

# plugin.xml

- Kiterjesztési pontok (extension point) definiálása
  - XSD séma, mely leírja a lehetséges kiterjesztési pontokat
    - Elemek lehetnek karakterláncok, forrásfájlok, erőforrások
    - Megadható, hogy egy elem kötelező vagy opcionális-e
  - Dokumentáció
- Kiterjesztések (extension) definiálása
  - Kiterjesztési pont hivatkozása
  - Sémának megfelelő XML leírás
  - Implementációhoz tartozó forrásfájlok hivatkozása
  - Erőforrásfájlok hivatkozása
- (Eclipse 3.0 előtt manifest.mf adatai is)

# plugin.xml példa

```
<?eclipse version="3.4"?>
<plugin>
  <extension
    point="org.eclipse.ui.commands">
    <command
      description="Shows a Helloworld message."
      id="com.optxware.commanddemo.helloworld"
      name="Display hello world!">
    </command>
  </extension>
  <extension
    point="org.eclipse.ui.handlers">
    <handler
      class="com.optxware.commanddemo.commands.HelloworldHandler"
      commandId="com.optxware.commanddemo.helloworld">
    </handler>
  </extension>
</plugin>
```

# plugin.xml példa

Kiegészítés megadása

```
<?eclipse version="3.4"?>
<plugin>
  <extension
    point="org.eclipse.ui.commands">
    <command
      description="Shows a Helloworld message."
      id="com.optxware.commanddemo.helloworld"
      name="Display hello world!">
    </command>
  </extension>
  <extension
    point="org.eclipse.ui.handlers">
    <handler
      class="com.optxware.commanddemo.commands.HelloworldHandler"
      commandId="com.optxware.commanddemo.helloworld">
    </handler>
  </extension>
</plugin>
```

# plugin.xml példa

Paraméterek megadása

```
<?eclipse version="3.4"?>
<plugin>
  <extension
    point="org.eclipse.ui.commands">
    <command
      description="Shows a Helloworld message."
      id="com.optxware.commanddemo.helloworld"
      name="Display hello world!">
    </command>
  </extension>
  <extension
    point="org.eclipse.ui.handlers">
    <handler
      class="com.optxware.commanddemo.commands.HelloworldHandler"
      commandId="com.optxware.commanddemo.helloworld">
    </handler>
  </extension>
</plugin>
```

# plugin.xml példa

```
<?eclipse version="3.4"?>
<plugin>
  <extension
    point="org.eclipse.ui.commands">
    <command
      description="Shows a Helloworld message."
      id="com.optxware.commanddemo.helloworld"
      name="Display hello world!">
    </command>
  </extension>
  <extension
    point="org.eclipse.ui.handlers">
    <handler
      class="com.optxware.commanddemo.commands.HelloworldHandler"
      commandId="com.optxware.commanddemo.helloworld">
    </handler>
  </extension>
</plugin>
```

Osztály megadása

# Plug-in metaadat szerkesztő

hu.bme.mit.commanddemo

## Extensions

All Extensions

Define extensions for this plug-in in the following section.

type filter text

- ▶ org.eclipse.ui.commands
- ▶ org.eclipse.ui.menus
- ▶ org.eclipse.ui.handlers
- ▶ org.eclipse.ui.commandImages

Add...

Remove

Up

Down

### Extension Details

Set the properties of the selected extension. Required fields are denoted by "\*".

ID:

Name:

- [Show extension point description](#)
- [Open extension point schema](#)
- [Find declaring extension point](#)

Overview Dependencies Runtime Extensions Extension Points Build MANIFEST.MF plugin.xml build.properties



# Kiterjesztési pontok

- Eclipse kiterjesztési pont
  - Bővítési lehetőség mindenki számára
- Hibakezelés
  - A kiterjesztések hibáit is kezelni kell!
  - Safe platform rule
    - „As the provider of an extension point, you must protect yourself against misbehavior on the part of extenders”
- Készítsünk bővítési lehetőségeket!
  - Invitation rule
    - „Whenever possible, let others contribute to your contributions.”
- Dokumentáció
  - A kiterjesztési ponthoz készítsünk dokumentációt

# Kiterjesztések

- Conformance rule
  - „Contributions must conform to expected interfaces”
  - A kiterjesztés feleljen meg a hívó elvárásainak
    - Kiterjesztési pont dokumentáció!

# Kiterjesztések

## ■ Sharing rule

- „Share, don't replace”
- A kiterjesztésünk más kiterjesztésekkel együtt is működjön
  - Ne próbáljuk kizárni a többi kiterjesztést!
  - Ne tételezzük fel, hogy nincs több kiterjesztés!
- Kivétel:
  - Ha a kiterjesztési pont egyszerre csak egy kiterjesztéssel működik
    - Pl. `org.eclipse.core.resources.teamHook`
  - Ilyenkor se a kiterjesztési pont zárja ki a többit!
  - Kiterjesztési pont dokumentáció!

# API megnyitása

## ■ API definiálása

- API csomagok kiajánlása (export)
  - Csak amit szükséges
- Nem kiajánlott: csak a plug-inen belül látszik
- Cél: információrejtés

## ■ Szokásos megoldás

- Grafikus felület: gui plugin
- Belső működés: core plugin
- Nagyobb projekteknél érdemes lehet még tovább bontani.
- Kikényszeríti API definiálását

# API tervezés

- Explicit API rule
  - „Separate the API from internals”
  - Pl. külön csomag (.internal végű)
- Stability rule:
  - „Once you invite others to contribute, don't change the rules”
  - API változik -> mindenki változik
- Defensive API rule:
  - „Reveal only the API in which you have confidence, but be prepared to reveal more API as clients ask for it.”

# Kiterjesztési pont - implementáció

## ■ Hogyan használhatjuk fel a kiterjesztéseket?

- `IExtensionRegistry`  
`Platform.getExtensionRegistry()`
  - Plug-in katalógus lekérése
- `IExtensionPoint`  
`registry.getExtensionPoint(String)`
  - Kiterjesztési pont lekérése
- `IExtension[] point.getExtensions()`
  - Kiterjesztések lekérése
- `IConfigurationElement`  
`extension.getConfigurationElements()`
  - Kiterjesztés definíciója (fastruktúra)

# IConfigurationElement

- Kiterjesztési pont definíciója
  - Kiterjesztés attribútumait tartalmazza
  - Fa struktúra (mint a plugin.xml leírásban)
- Validálható (kiterjesztési pont XSD sémával)
- `String element.getAttribute(String)`
  - Attribútum lekérése
  - Paraméter az attribútum neve
- `Object element.createExecutableExtension(String)`
  - Kiterjesztés osztályattribútumának példányosítása
    - Nem kell ezt az osztályt kiajánlani
  - Csak default (paraméter nélküli) konstruktorral!
  - Megfelelő típusra kell castolni

# Hibakezelés

- Interfészdefiníció: a kiterjesztési pontban
- Good fences rule:
  - „When passing control outside your code, protect yourself.”
- Kivételek
  - Kapjunk el minden dobott kivételt
  - Minden kiterjesztés hívása külön try-catch blokk
    - fault containment region
- Segítség implementációhoz:
  - ISafeRunnable megvalósítás
    - run() metódus: itt hívjuk meg a kiterjesztést
    - handleException(Throwable) metódus: itt kezelhetjük a hibákat



# Kiterjesztések

- Monkey see/monkey do rule
  - „Always copy the structure of a similar plug-in”
  - A már bevált receptet kell alkalmazni
    - Gyakori kiterjesztésekhez van template wizard
- Relevance rule
  - „Contribute only when you can successfully operate”
  - Csak akkor, ha használható
    - Különben: túlzásfolt felület
- Kövessük a platform konvencióit!
  - Használjuk a meglévő parancsokat, szolgáltatásokat
  - Másoljuk a meglévő GUI-kat is
  - Interface Guidelines van:
    - [http://wiki.eclipse.org/User\\_Interface\\_Guidelines](http://wiki.eclipse.org/User_Interface_Guidelines)

# Plug-inek futtatása

# Target Platform

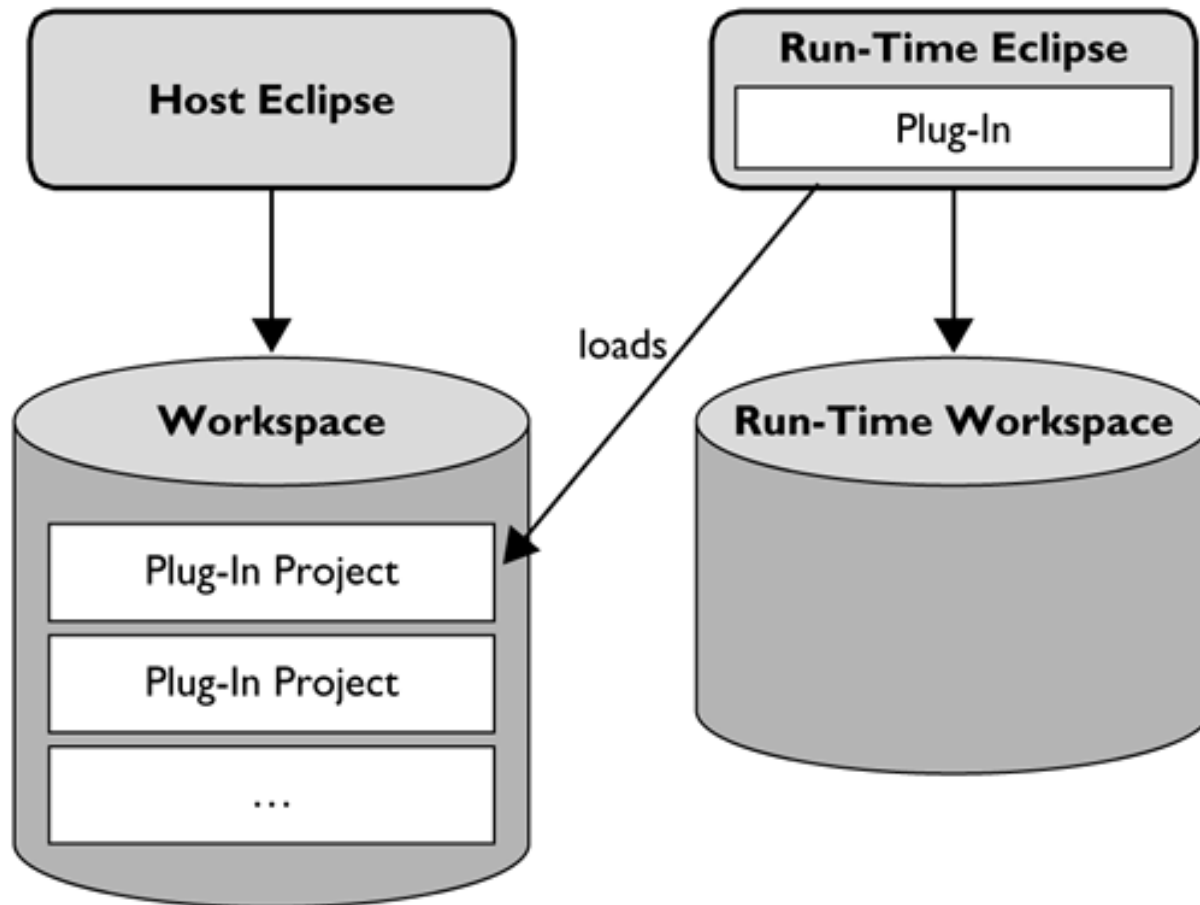
- Target Platform
  - Plug-in készlet
  - Fejlesztett plug-in függőségei!
    - Fordítás
    - Futtatás
  - Külön platform IDE, RCP és RAP alkalmazásokhoz
  - 3.5-ben jelentősen fejlődött a többféle platform kezelés

# Plug-in futtatás

## ■ Futtatás

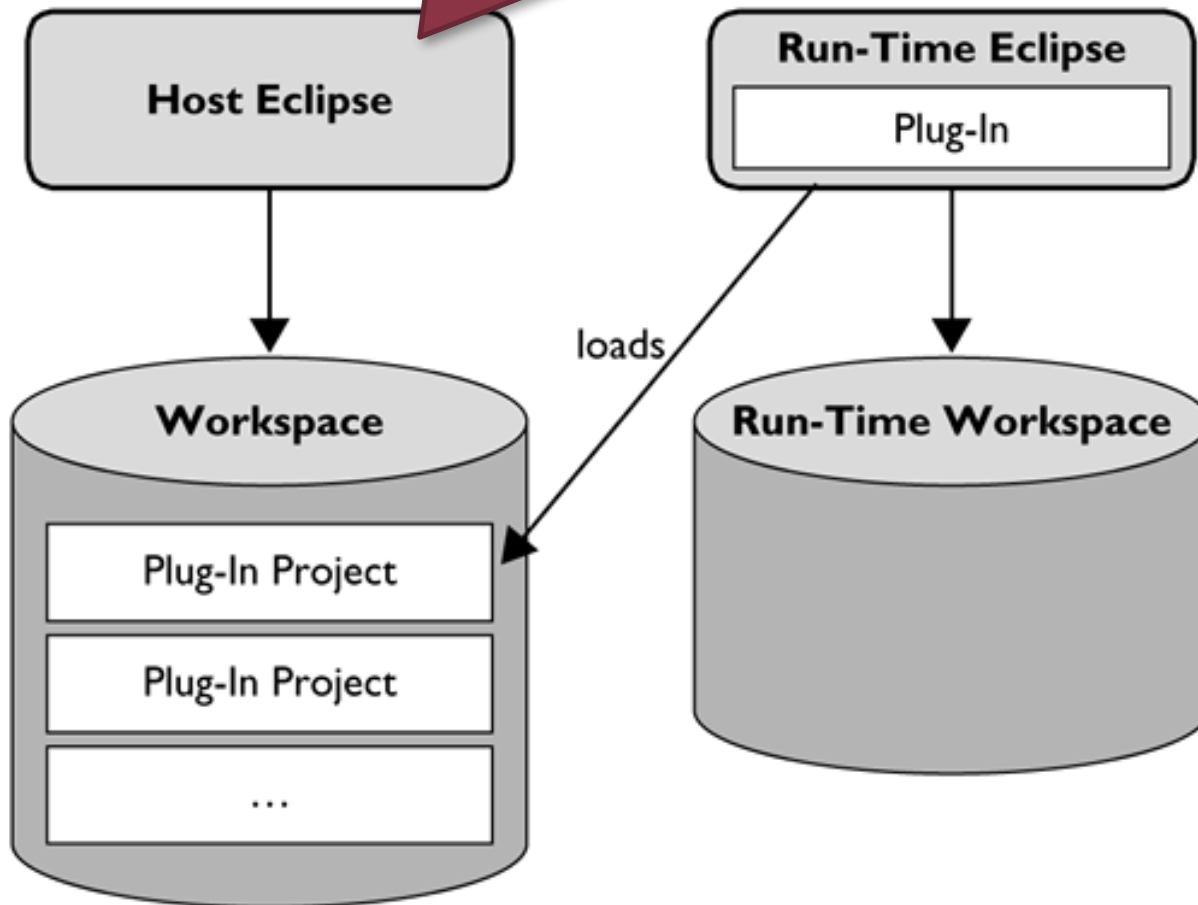
- Target Platform és saját plug-in alkotta rendszer fut
- IDE: teljes értékű Eclipse workbench indul
- RCP/RAP: a platformon futó alkalmazás indul el

# Futtatás

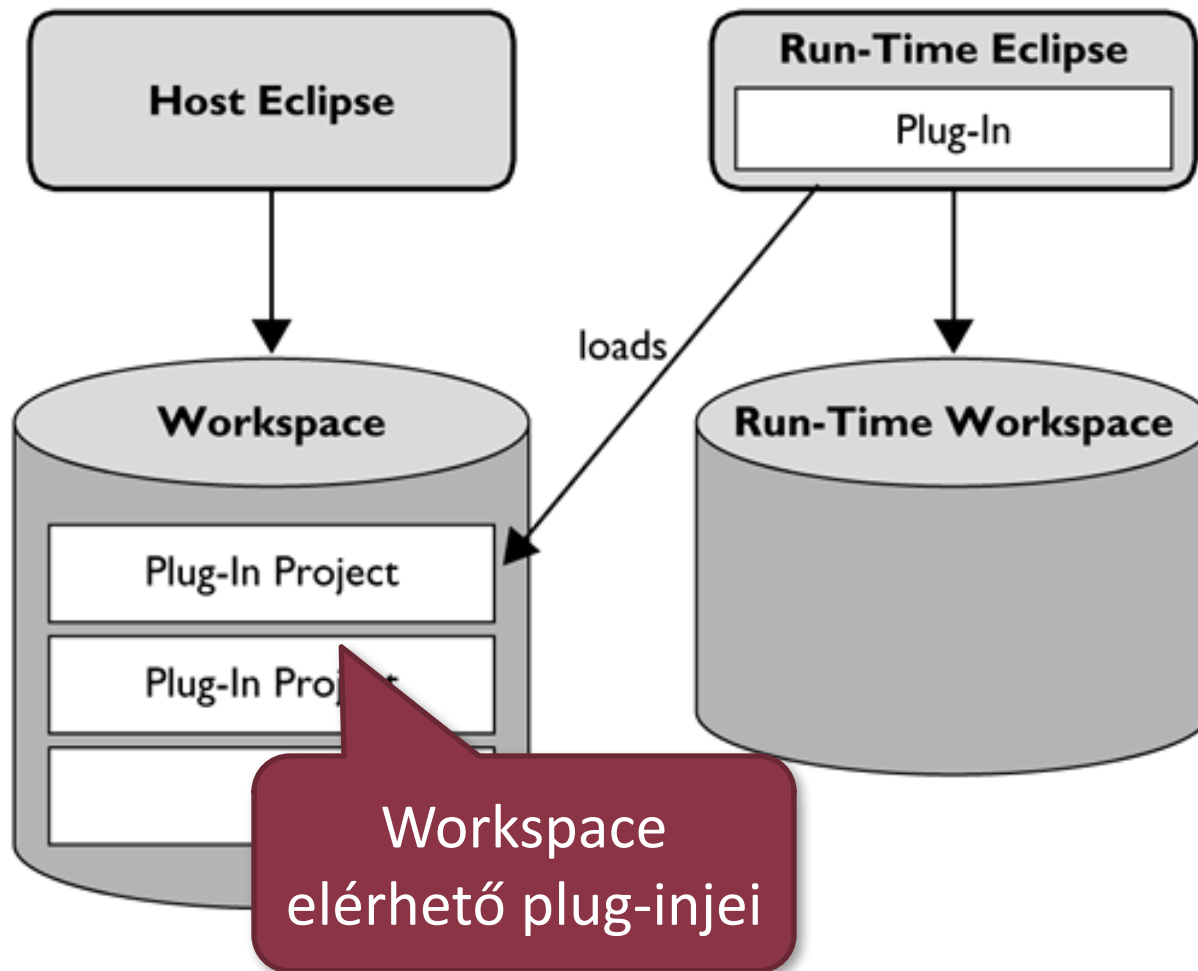


# Futtatás

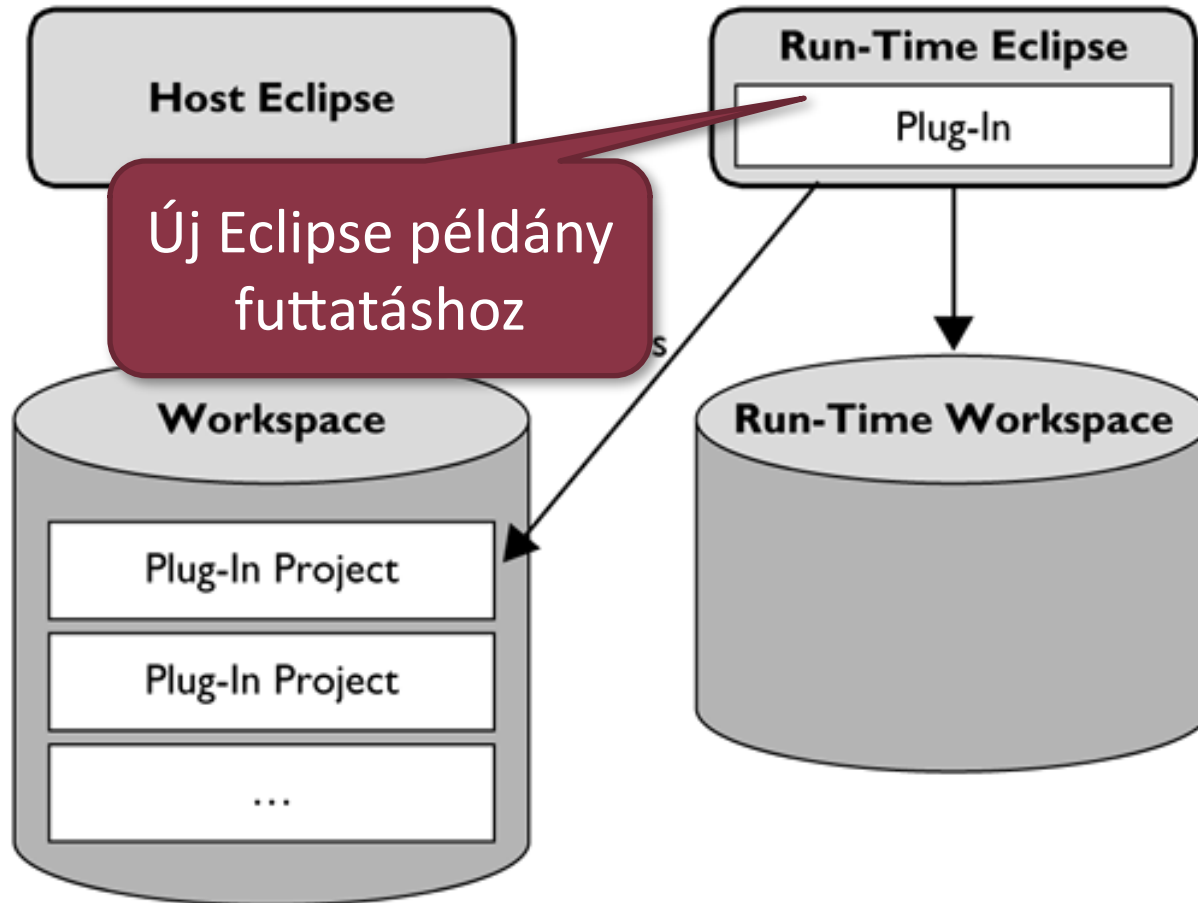
Alapvető munkakörnyezet



# Futtatás

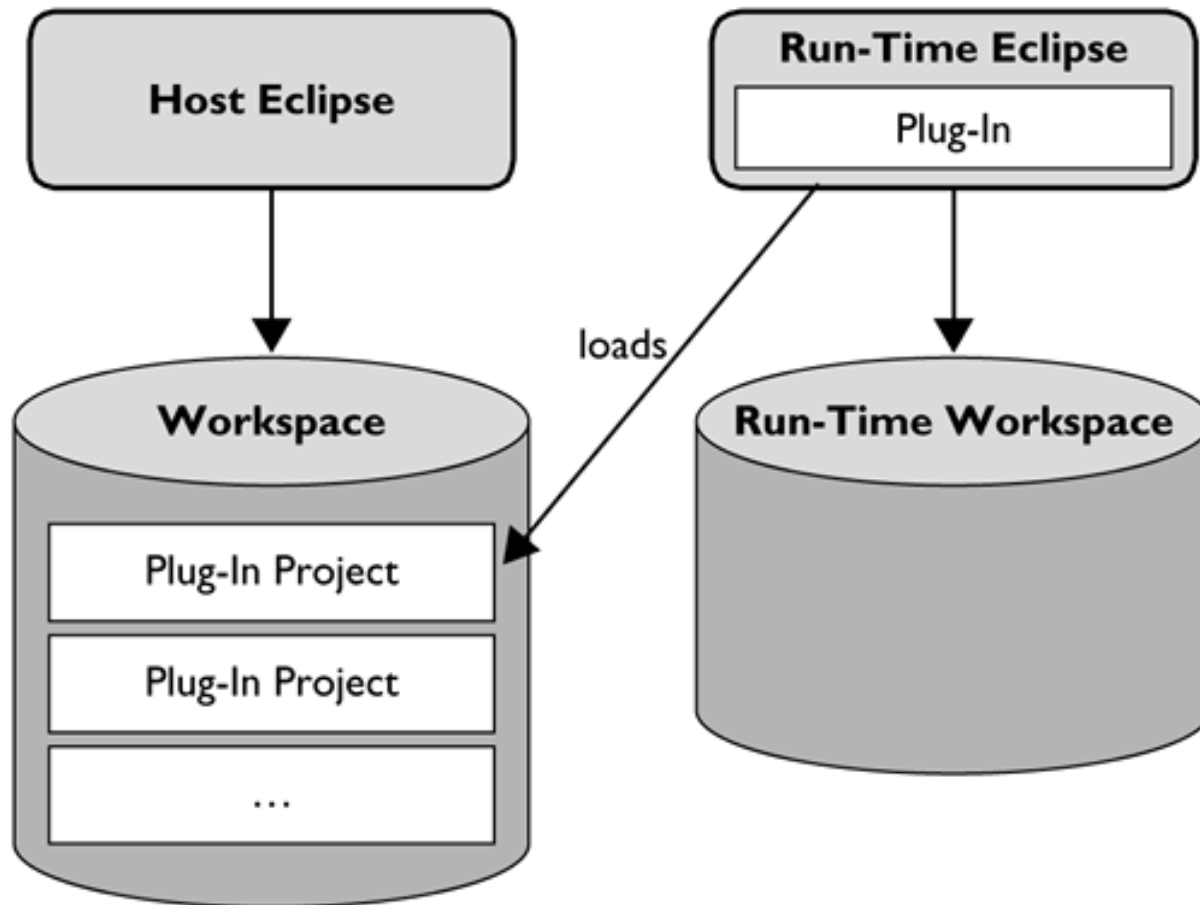


# Futtatás

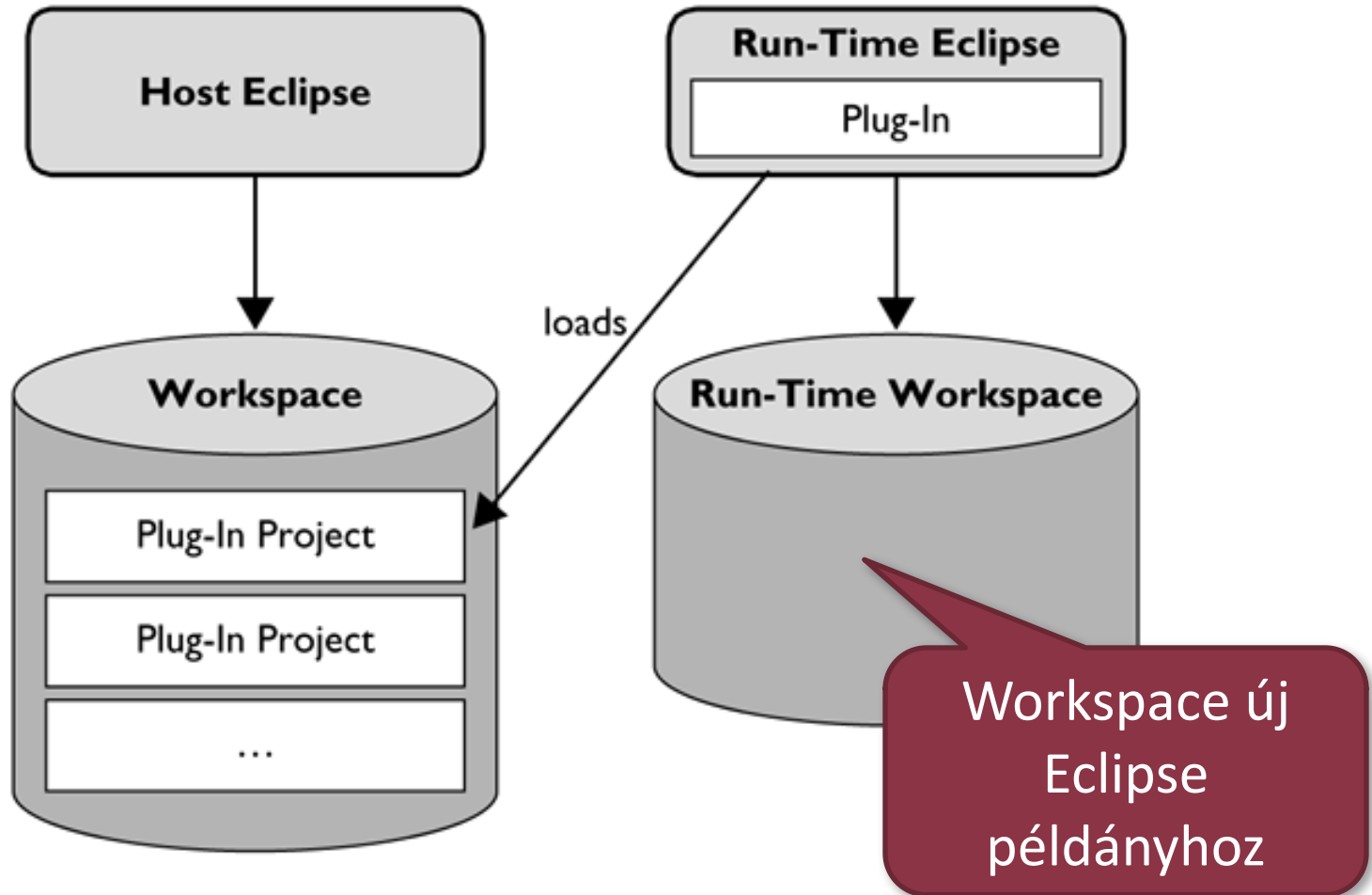




# Futtatás



# Futtatás

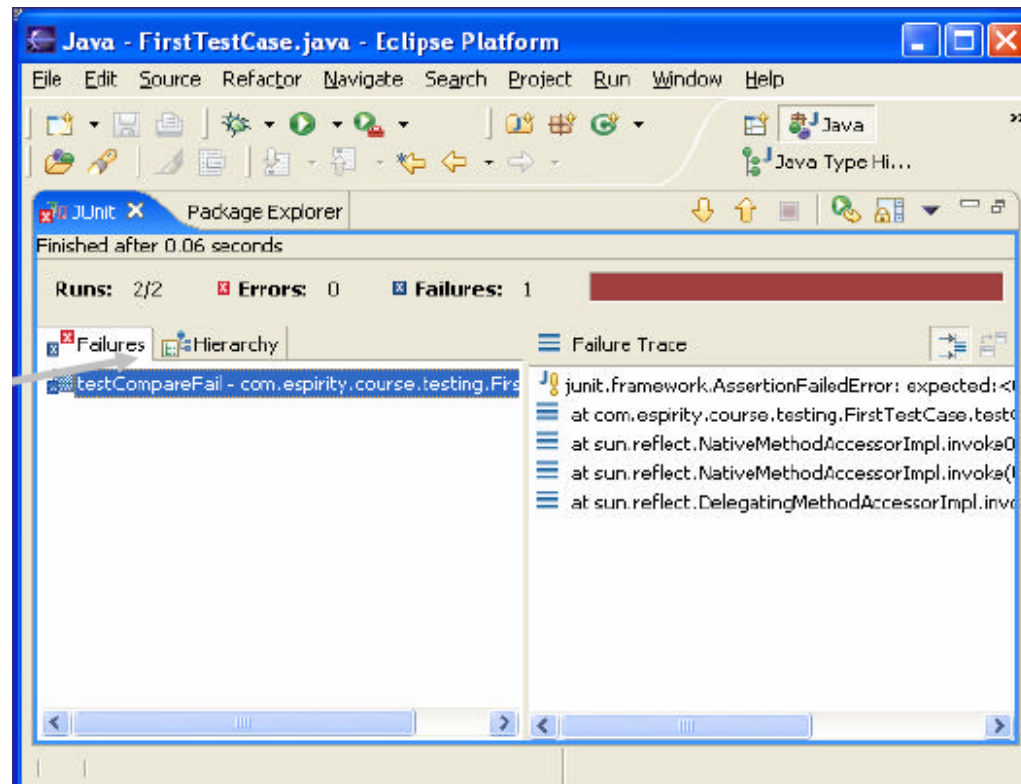


# Plug-in unit tesztelés

- Eclipse tartalmazza a JUnit keretrendszert
  - 3.8 ill. 4.5-ös verzió
- Tesztesetek futtatása:
  - *JUnit Plug-in Test* Run Configuration
  - Eclipse ablak nyílik meg, ill. záródik be futás után
  - A teszteredmények megjelennek az Eclipse grafikus felületén

# JUnit View

- Információk
  - Piros/zöld a teszt eredménye hiba/ok
  - Meghiúsult tesztek neve
  - Hiba trace
  - Lefutott tesztek száma
  - Hibák száma



# Plug-in unit tesztelés

## ■ Tesztek helye

### ○ Tesztelendő plug-in

- JUnit függőség!
- Kód és teszt keveredik!
- Teszt-esetekre nincs mindenkinek szüksége!

### ○ Külön plug-in

- Könnyen leválasztható
- A tesztelt plug-in független a JUnittól
- A tesztelő plug-in függjön a tesztelttől

# Gyakori kiterjesztések

# Nézet készítése

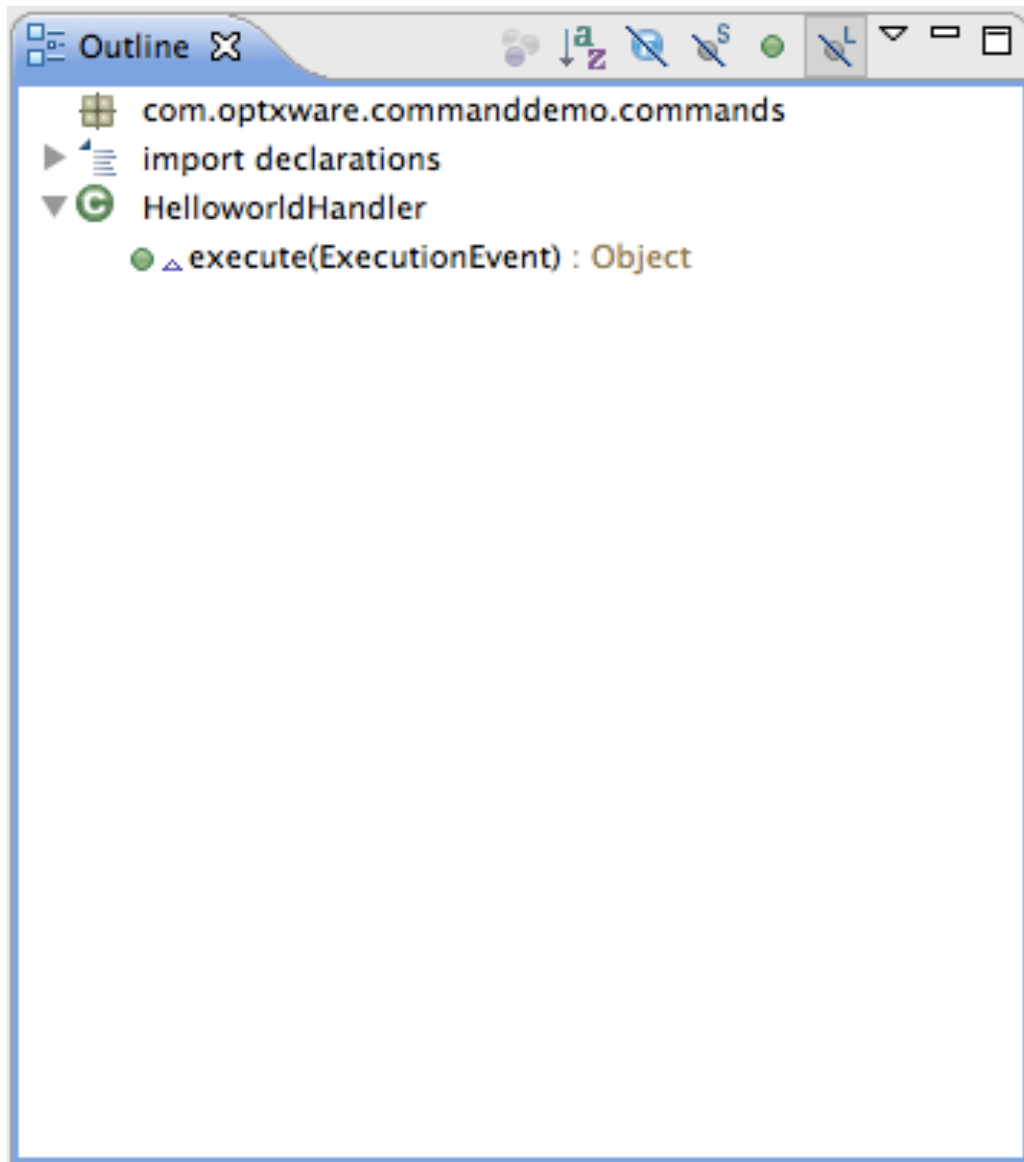
## ■ View

- Kiterjesztési pont: `org.eclipse.ui.views`
- Interfész: `IViewPart`

## ■ Cél:

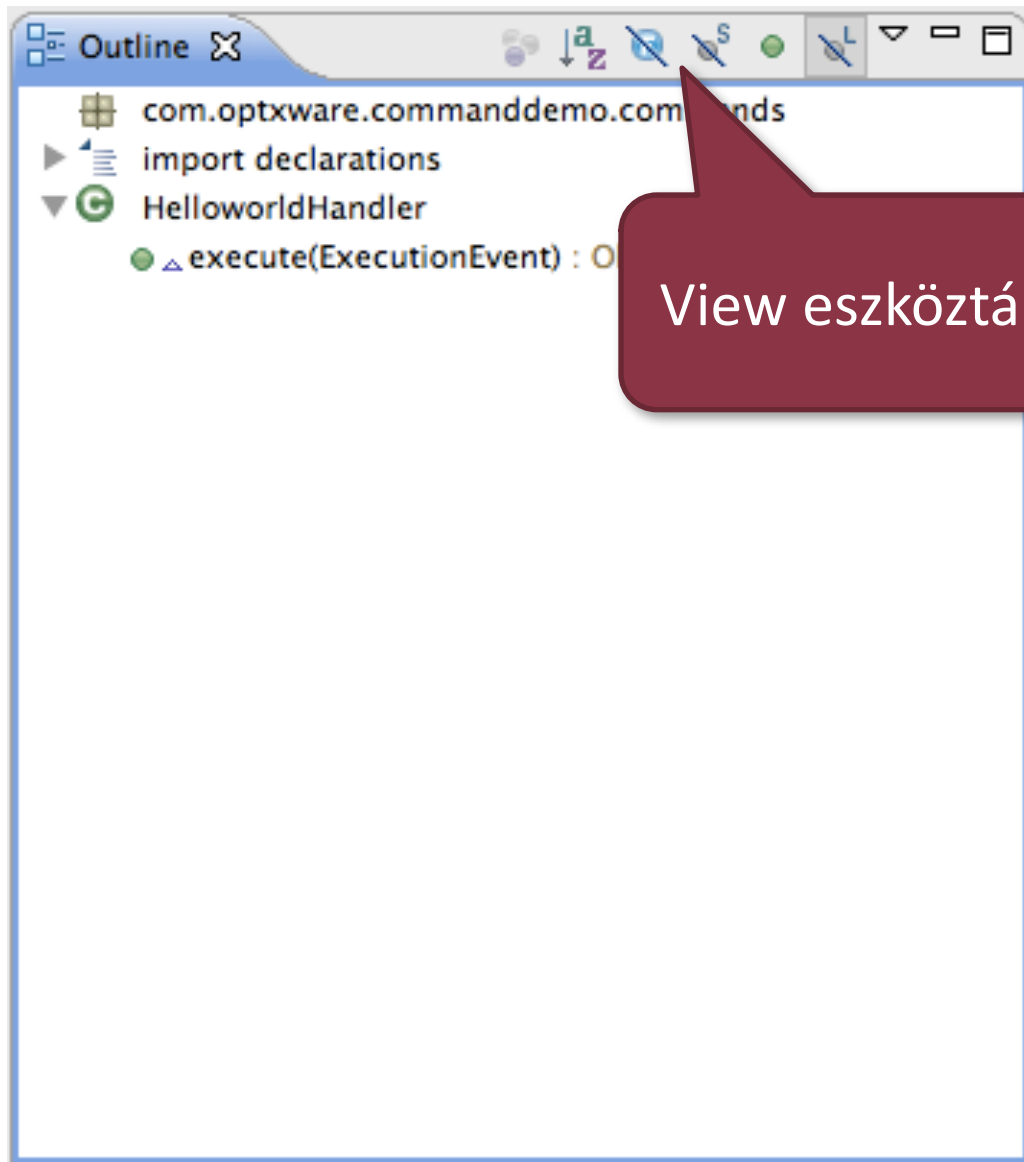
- Információk megjelenítése
- Információk szerkesztése
- Kontextusfüggő
  - Pl. kijelölésfüggő

# Nézet felépítése

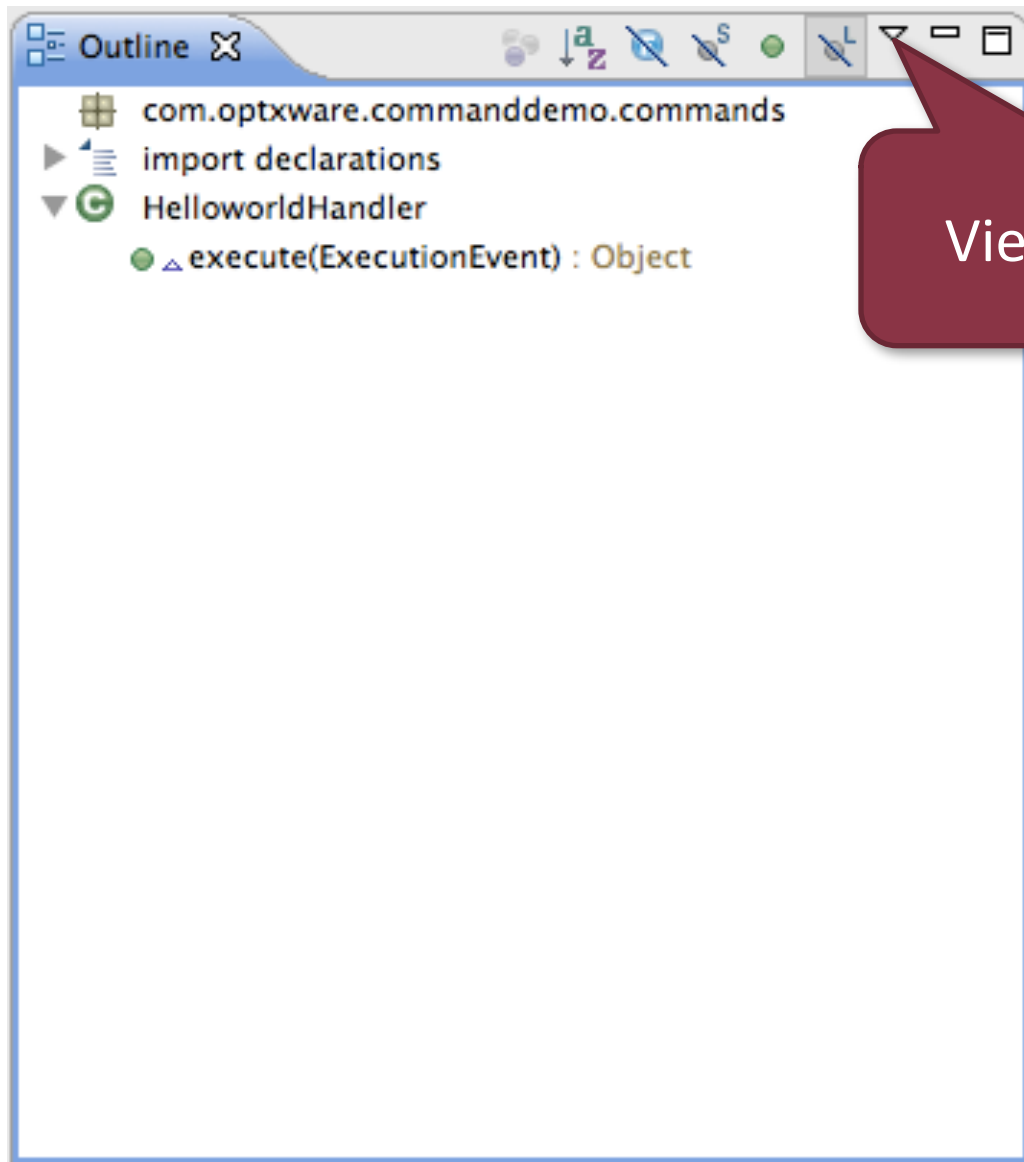




# Nézet felépítése

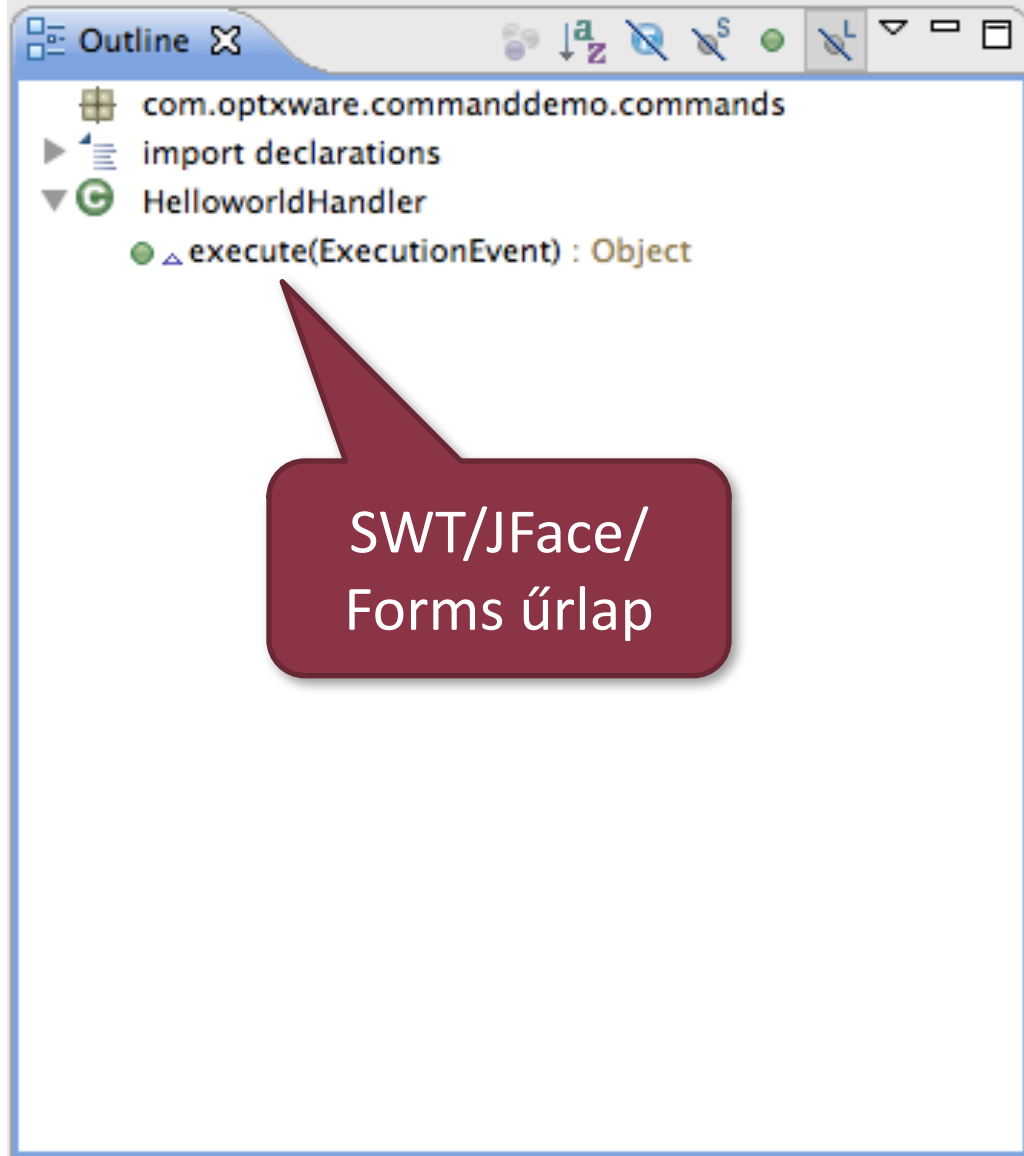


# Nézet felépítése



View menü

# Nézet felépítése



# Példa nézet

```
public class View extends ViewPart {  
    public static final String ID =  
        "com.optxware.app.view";  
  
    public void createPartControl  
        (Composite parent) {  
        //View controls initialization  
    }  
  
    public void setFocus() {  
        viewer.getControl().setFocus();  
    }  
}
```

# Perspektívák

- Perspective
  - Kiterjesztési pont: `org.eclipse.ui.perspectives`
  - Interfész: `IPerspectiveFactory`
- Megjelenítendő nézetek és editorok számára definiál megjelenést
  - Editor területhez képest lehet megadni a többi nézet helyét
- Bővítési lehetőségek
  - Hol jelenjen meg egy view, ha megnyitják?
  - Látható-e egy view kezdetben?

# Szerkesztők

- Editor
  - Kiterjesztési pont: org.eclipse.ui.editor
  - Interfész: IEditorPart
- Minimális funkciókészlet
  - Szerkesztő megjelenítés
  - Állapotjelzés
  - Adat mentés

# Szerkesztők

- `createPartControls(Composite parent)`
  - Űrlap megjelenítése
    - Szöveges editornál ez egy többsoros szövegmező
    - De tetszőleges űrlap építhető
- `init(IEditorSite site, IEditorInput input)`
  - Szerkesztő inicializálása
  - Adatmodell beolvasása
  - Adatmodell megjelenítése az elkészült widgeteken
  - Minden szerkesztőhöz egyedi megoldás

# Szerkesztők – Adatmodell megadása

- **IEditorInput: Adatmodell leíró (handle)**
  - A minimális adat, amely segítségével összegyűjthető a tartalom
  - Ne tartalmazza a teljes adatmodellt
    - A leíró a szerkesztő bezárása után is a memóriában marad!
- **Leíró típusok**
  - **FileEditorInput:** a bemenet egy fájl a workspace fájlrendszerben
    - A szerkesztő célja a fájl módosítása
    - Ez a leggyakrabban használt
  - **CompareEditorInput:** a bemenet egy fájl több változata
    - A szerkesztő célja változatok összehasonlítása és egyesítése
  - **MultiEditorInput:** több leíróból álló összetett leíró többlapos szerkesztőhöz



# Szerkesztők - Állapotjelzés

- `isDirty()`
  - Jelzi, hogy megváltozott-e az editor tartalom a mentetthez képest
  - Ha igaz, akkor engedélyezi a Workbench a Mentés parancsot
- Ha szerkesztés történik, értesítést kell küldeni
  - `firePropertyChange (PROP_DIRTY) ;`
  - Nem küldi át az új értéket
    - `isDirty()` hívás a környezettől

# Szerkesztők – Adatmodell mentése

- `isSaveAsEnabled()`
  - Megadja, hogy engedi-e a szerkesztő az adat mentését egy újabb fájlba
- `doSave(IProgressMonitor monitor)`
  - A rendszer Save parancsát kiválasztva hívódik meg
  - Egyedi megoldás kell a mentés elvégzésére
  - A paraméter segítségével állapotjelzést lehet adni a mentéshez
- `doSaveAs()`
  - A rendszer Save as parancsát kiválasztva hívódik meg
  - Egyedi megoldás kell az új hely kiválasztására
  - Egyedi megoldás a mentésre
  - Nincs hozzá monitor!

# Szerkesztők – Fejlettebb lehetőségek

- Egyszerű szöveges editor
  - AbstractTextEditor osztályból származtatni a szerkesztőt
  - Xtext és IMP projekt
    - Képes nyelvtan fájlokhoz szerkesztőt generálni
- Grafikus szerkesztő
  - Graphical Editing Framework (GEF) projekt
    - Grafikus szerkesztő készítése
  - Graphical Modeling Framework (GMF) projekt
    - GEF alapon EMF modellekhez grafikus szerkesztő készítése
- Többlapos szerkesztő
  - MultiPageEditor osztályból származtatni a szerkesztőt
  - A különböző lapokon lehet különböző típusú szerkesztő
    - Form alapú, szöveges vagy grafikus

# Eclipse súgó

- Help
  - Kiterjesztési pont: [org.eclipse.help.toc](http://org.eclipse.help.toc)
  - Interfész: nincs
- TOC (Table of Contents) elemekből áll
  - Fa struktúra
  - Egyéb TOC-eket és topicokat tartalmaz
  - Fő TOC: [toc.xml](#) (konvenció)
- Topic
  - HTML fájl - ez a tényleges tartalom
- Lehet külön plug-in is
- Fontos:
  - a HTML fájlok könyvtárát is csomagolni kell!

# Eclipse súgó

- Topic elemek
  - Címke
  - Fájl hivatkozás a tartalomra
  - Minden HTML tartalom használható (képek, linkek, ...)
- Létező elem bővítése új oldalakkal
  - TOC-ban Anchor helyezhető el (-> invitation rule)
  - TOC link\_to attribútum: a beállított Anchor lesz a TOC gyökere

# Eclipse Help

The screenshot shows the Eclipse Help window titled "Help - Eclipse SDK". The browser address bar displays "http://127.0.0.1:53980/help/index.jsp". The search bar contains "Search scope: All topics". The left sidebar, labeled "Contents", lists various guides and documents, including "Workbench User Guide", "Java development user guide", "Platform Plug-in Developer Guide", "Programmer's Guide", "Reference", "What's new", "3.5 Plug-in Migration Guide", "Examples Guide", "Questions Index", "Legal", "JDT Plug-in Developer Guide", "Plug-in Development Environment Guide", "Acceleo Model To Text Transformation", "BIRT Report Developer Guide", "C/C++ Development User Guide", "EclEmma Java Code Coverage", "Eclipse Articles", "Eclipse Metrics", and "EclipseCon 05 Slides". The main content area displays the page "Using the help system", which includes sections for "Using the help system", "Searching", "Synchronizing", and "Capabilities".

**Using the help system**

Browse topics in the **Contents** frame (📁) on the left. Click on a topic to have it displayed. Use the **Back** and **Forward** buttons to navigate within the history of viewed topics.

**Searching**

To quickly locate topics on a particular subject in the documentation, enter a query in the **Search** field. Use the **Search** frame (🔍) to display the Search view. You can narrow the **scope** of your search by selecting only the sections you are interested in.

**Synchronizing**

Clicking the **Show in Table of Contents** button (📁) will select that topic in the navigation tree. The **Link with Contents** button (🔗) keeps the navigation tree synchronized to the current topic.

**Capabilities**

To show documentation about capabilities that are disabled in the application, select the **Show All Topics** button (🔗). This affects the table of contents and search results.

# Eclipse Help

Kategóriák és témák

The screenshot shows the Eclipse Help window titled "Help - Eclipse SDK". The left pane, labeled "Contents", contains a tree view of help topics. The right pane displays the article "Using the help system".

**Contents**

- Workbench User Guide
- Java development user guide
- Platform Plug-in Developer Guide
  - Programmer's Guide
  - Reference
    - What's new
  - 3.5 Plug-in Migration Guide
  - Examples Guide
  - Questions Index
  - Legal
- JDT Plug-in Developer Guide
- Plug-in Development Environment Guide
- Acceleo Model To Text Transformation
- BIRT Report Developer Guide
- C/C++ Development User Guide
- EclEmma Java Code Coverage
- Eclipse Articles
- Eclipse Metrics
- EclipseCon 05 Slides

**Using the help system**

Browse topics in the **Contents** frame (📁) on the left. Click on a topic to have it displayed. Use the **Back** and **Forward** buttons to navigate within the history of viewed topics.

**Searching**

To quickly locate topics on a particular subject in the documentation, enter a query in the **Search** field. Use the **Search** frame (🔍) to display the Search view. You can narrow the **scope** of your search by selecting only the sections you are interested in.

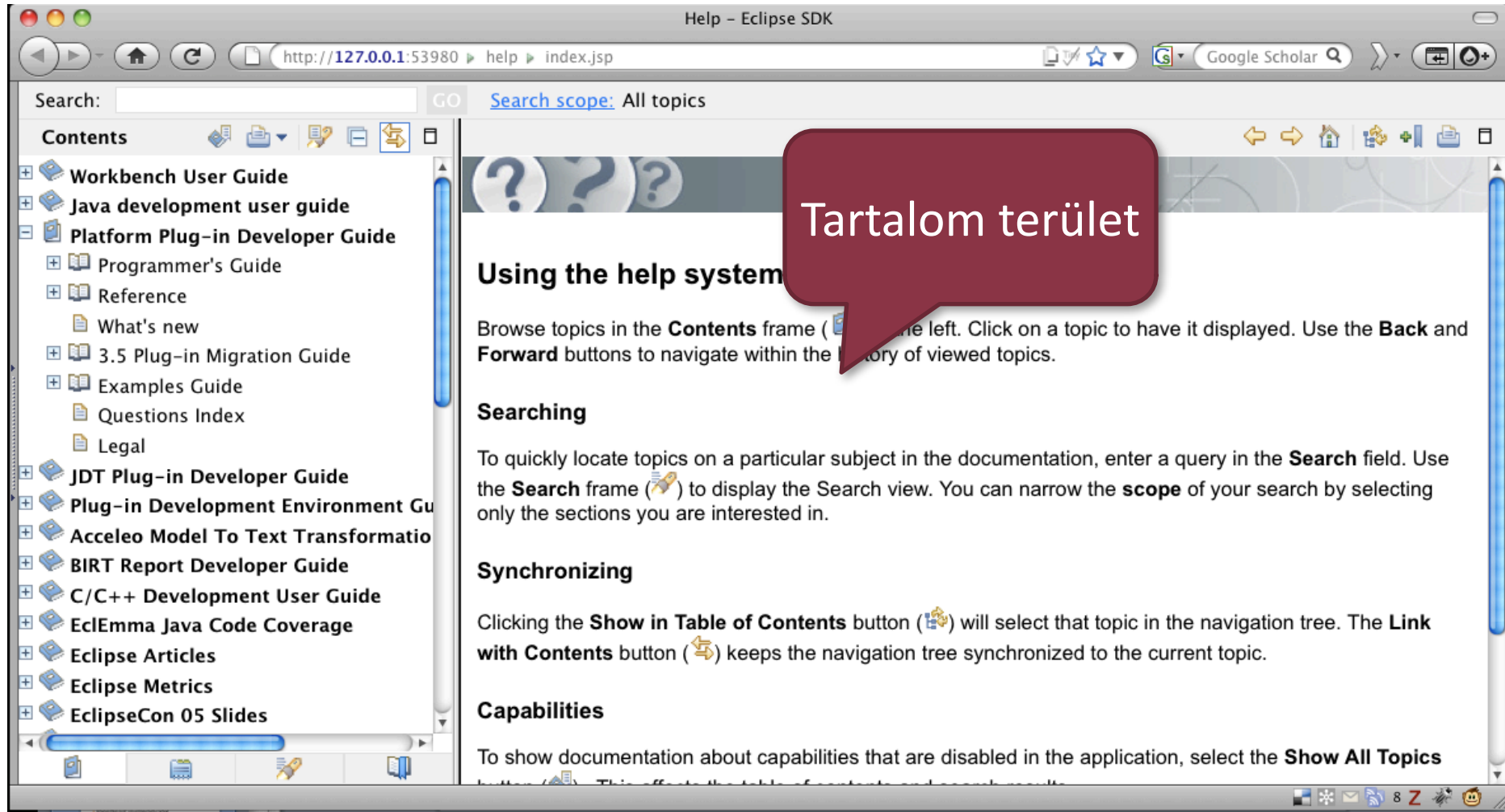
**Synchronizing**

Clicking the **Show in Table of Contents** button (📁) will select that topic in the navigation tree. The **Link with Contents** button (🔗) keeps the navigation tree synchronized to the current topic.

**Capabilities**

To show documentation about capabilities that are disabled in the application, select the **Show All Topics** button (📁). This affects the table of contents and search results.

# Eclipse Help



Help - Eclipse SDK

http://127.0.0.1:53980/help/index.jsp

Search:  GO Search scope: All topics

**Contents**

- Workbench User Guide
- Java development user guide
- Platform Plug-in Developer Guide
  - Programmer's Guide
  - Reference
    - What's new
  - 3.5 Plug-in Migration Guide
  - Examples Guide
  - Questions Index
  - Legal
- JDT Plug-in Developer Guide
- Plug-in Development Environment Guide
- Acceleo Model To Text Transformation
- BIRT Report Developer Guide
- C/C++ Development User Guide
- EclEmma Java Code Coverage
- Eclipse Articles
- Eclipse Metrics
- EclipseCon 05 Slides

**Using the help system**

Browse topics in the **Contents** frame (left) to find the topic you want. Click on a topic to have it displayed. Use the **Back** and **Forward** buttons to navigate within the history of viewed topics.

**Searching**

To quickly locate topics on a particular subject in the documentation, enter a query in the **Search** field. Use the **Search** frame (right) to display the Search view. You can narrow the **scope** of your search by selecting only the sections you are interested in.

**Synchronizing**

Clicking the **Show in Table of Contents** button (left) will select that topic in the navigation tree. The **Link with Contents** button (right) keeps the navigation tree synchronized to the current topic.

**Capabilities**

To show documentation about capabilities that are disabled in the application, select the **Show All Topics** button (right). This affects the table of contents and search results.

**Tartalom terület**



# Eclipse súgó

- **Környezetérzékeny Súgó**
  - Kiterjesztési pont: `org.eclipse.help.context`
  - Interfész: nincs
- **Help context extension**
  - XML fájl
  - Kontextusok leírása (Topic)
  - Általános leírás
- **Hozzárendelés**
  - `WorkbenchHelp.setHelp(Control, String)`
- **“F1” megnyomásával érhető el**

# Környezetérzékeny súgó

com.optxware.commanddemo

## Overview

### General Information

This section describes general information about this plug-in.

ID:

Version:

Name:

Provider:

Platform Filter:

Activator:

Activate this plug-in when one of its classes is loaded

This plug-in is a singleton

### Plug-in Content

The content of the plug-in is

- [Dependencies](#): lists all the plug-in's classpath to co
- [Runtime](#): lists the libraries in's runtime.

### Extension / Extension Point

This plug-in may define exte

- [Extensions](#): declares cont makes to the platform.
- [Extension Points](#): declare plug-in adds to the platf

### Testing

### Execution Environments

Specify the minimum execution environments required.

Overview Dependencies Runtime Extensions Extension Points Build MANIFEST.MF >> 2

## Help

### Related Topics

#### About Plug-in Manifest Editor

The Overview page shows you the plug-in at a glance.

See also:

- [Plug-in](#)
- [Plug-in Overview](#)

More results:

- [Search for "Plug-in Manifest Editor"](#)

Go To:

- [Contents](#)
- [Search](#)
- [Bookmarks](#)
- [Index](#)

# Többnyelvű alkalmazások

- String konstansok kiemelése
  - Cél: könnyű cseréhetőség
- plugin.xml egyes szövegei (pl. címkék) megjelennek a felületen
  - Honosítani kell
  - plugin.properties - név-érték párok
  - %név - az adott helyre behelyettesíti a talált értéket
- Java forráskód nem lefordítandó részeit jelölni kell:
  - `// $NON-NLS-1$` megjegyzésekkel
  - Nem annotáció - nem igényel Java 5-öt

# Forráskód többnyelvűsítése

- Resource bundle létrehozása a plug-inhoz
- Minden bundle-be kerüljön egy properties fájl
- Minden bundle-be egy helper osztály a properties betöltésére
- A kulcsok legyenek kvalifikáltak (ütközés elkerülés)
  
- Támogatás: `Source` / `Externalize strings`
  - Megkeresi a stringeket
  - Létrehozza a bundle-t
  - Létrehozza a beolvasó osztályt

# Többnyelvű terjesztés

- Fordítások megvárása -> idővesztés
- Új verzió minden fordításhoz -> bonyolult
- Plug-in fragmentek
  - Külön elemek, amelyek a plug-in egy részét tartalmazzák
  - Nyelv vagy platformspecifikus elemekhez találták ki
  - Invitation rule!

# Accessibility

- A plug-int mindenki tudja használni
  - Minden funkció legyen elérhető billentyűzetről
  - Az elemekhez rendeljünk címkéket
    - Felolvasó programok tudják használni
  - A színeket csak kiterjesztésként használjuk
  - Tegyük lehetővé erős kontrasztos üzemmód használatát
  - A hangjelzések és a képek legyenek kiválthatóak

# Háttérfolyamatok

Job API

# Háttérfolyamatok

- Hosszú műveleteket háttérszálon célszerű futtatni
  - Ekkor a felhasználói felület nem szaggat
- Eclipse alatt támogatás
  - Job: Egy háttérben elvégzendő feladat
  - Ütemezett végrehajtás
    - Időzítés
    - Ismétlés
  - Állapotjelzés a felhasználói felületen
    - Állapotjelzés
    - Szöveges állapot



# Job

- Plug-in függőség: `org.eclipse.core.runtime`
- Saját leszármazottat kell készíteni a Job osztályból
- Legfontosabb metódusok:
  - `run(IProgressMonitor monitor)`
    - A jobhoz tartozó végrehajtható kód – mindig felüldefiniálendő
  - `setPriority(int priority)`
    - Prioritás beállítása (`Job.LONG`, `Job.SHORT` konstansok)
  - `setName(String name)`
    - Név beállítása (megjelenik a felhasználói felületen)
  - `schedule(long delay)`
    - Késleltetett időzítés
  - `schedule()`
    - Azonnali időzítés (amilyen hamar csak lehet)

# Hello, world! Job

```
Job job = new Job("My First Job") {
    protected IStatus run
    (IProgressMonitor monitor) {
        System.out.println("Hello World
        (from a background job)");
        return Status.OK_STATUS;
    } };

job.setPriority(Job.SHORT);
job.schedule(); // start as soon as
possible
```

# Állapotjelzés

- run() paramétere: IProgressMonitor
  - Interfész a folyamatok nyomon követésére
  - A megvalósító objektum tárolja a folyamat állapotát
- Állapotváltozások
  - Végrehajtandó feladat frissíti
  - Tárolja a megszakítási kérést

# Állapotjelzés: IProgressMonitor

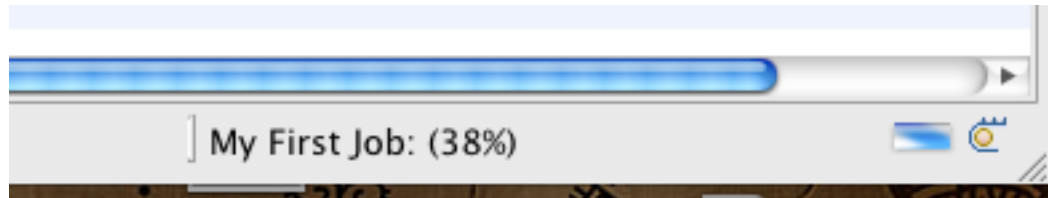
- Feladatok állapotának nyomonkövetése
  - `beginTask(String name, int totalWork)`
    - Részfolyamat kezdése névvel és elvégzendő lépésszámmal
  - `worked(int work)`
    - Az elvégzett lépésszám frissítése
  - `done()`
    - Jelzés, hogy a feladat befejeződött
- Hierarchikus feladatkezelés
  - `subTask(String)`
    - Jelzés, hogy egy alfolyamat megkezdődött

# Állapotjelzés

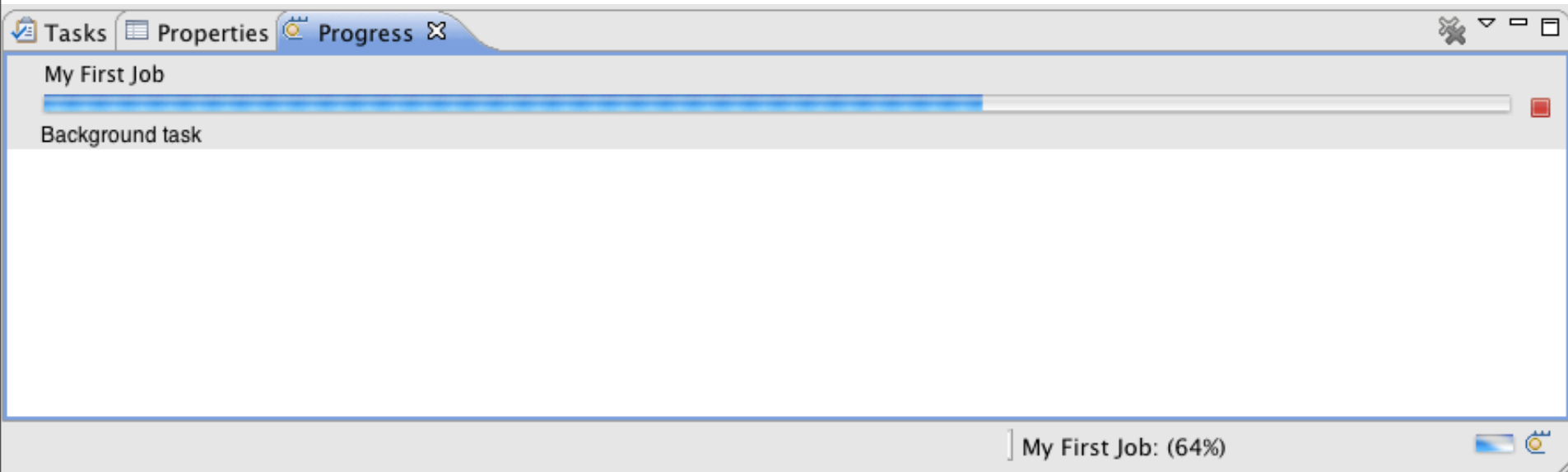
```
Job job = new Job("My First Job") {
    protected IStatus run(IProgressMonitor
monitor) {
        monitor.beginTask("Background task",
1000);
        for (int i=0; i<1000; i++) {
            calculationStep();
            monitor.worked(1);
        }
        monitor.done();
        return Status.OK_STATUS;
    }
};
```

# Állapotjelzés

- Háromféle megjelenés IDE alatt
  - Státuszszor



- Állapot nézet (Progress view)



# Állapotjelzés

## ■ Felhasználói folyamat

- `job.setUser(true);`
- Ajánlás: közvetlen felhasználó által kezdeményezett folyamat
  - Modális pop-up
  - Bezárható



# Folyamat megszakítása

- Megszakítás kezdeményezhető
  - Felhasználói felületen
    - Állapot nézetben nyomógombbal
    - Felhasználói folyamat ablakában
  - Kódban
    - A `Job cancel ()` metódusát kell meghívni
- Ez csak megszakítási kérés
  - Nem szakítja meg a végrehajtást
  - A kérés figyelése a végrehajtandó kód feladata



# Folyamat megszakítása

- Kód felkészítése megszakítás kezelésére:
  - `IProgressMonitor.isCanceled()` pollozása
  - Kilépés kétféleképpen
    - `return Status.CANCEL_STATUS;`
    - `OperationCanceledException` dobása
      - Ezt hívási lánc mélyén is lehet

# Folyamat megszakítása - példa

```
Job job = new Job("My First Job") {
    protected IStatus run(IProgressMonitor monitor) {
        monitor.beginTask("Background task", 1000);
        for (int i=0; i<1000; i++){
            calculationStep();
            monitor.worked(1);
            if (monitor.isCancelled())
                return STATUS.CANCEL_STATUS;
        }
        monitor.done();
        return Status.OK_STATUS;
    }
};
```

# Megjelenítő szál visszahívása

- A felhasználói felületet csak a megjelenítő szálon frissítsük
- Megoldás:
  - Display: `syncExec(Runnable r)`
    - A háttérszál megvárja, amíg a megjelenítő szál végez.
  - Display: `asyncExec(Runnable r)`
    - A háttérszál nem vár a futtatott parancs elvégzésére

# Feladatütemezés

- A Jobokat a platform ütemezi
  - Prioritás és késleltetés alapján
- Az időzítési kérelemhez késleltetés adható
  - Egy másodperces várakoztatás
    - `job.schedule(1000);`
  - Ezután a platform ütemezni fogja a végrehajtást
- Ismételt végrehajtás
  - Egy Job csak egyszer várakozhat az ütemezésre
  - Ismétlés: befejezéskor `schedule()` újrahívása
  - Így lehet periodikusan ismétlődő háttérfolyamatot definiálni

# Egyéb szolgáltatások

- Alfolyamatok létrehozása
- Ütemezi szabályok hozzáadása
- Erőforrásfoglalás
  - Holtpontdetekció

# Job API - Összefoglalás

- Magas szintű támogatás többszálú futáshoz
  - Állapotjelzés a felhasználói felületen
  - Ütemezés
- Kényelmes többszálú fejlesztés

# Parancsok hozzáadása

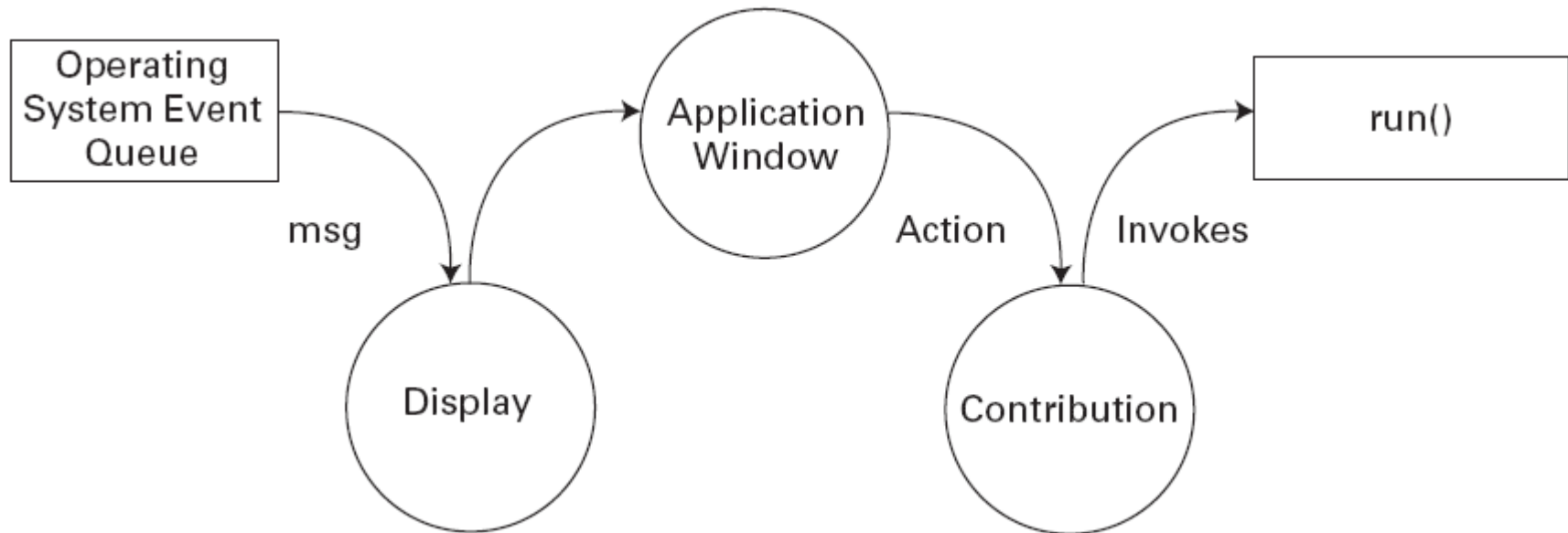
Command Framework

# Akciók és parancsok

- Új menüpontok és eszköztár gombok felvétele
  - Két megközelítés
    - JFace Action
    - Command Framework (Eclipse 3.3 óta)
  - Jelenleg mindkettő használható
  - Itt
    - Action megközelítés csak említés szintjén
    - Részletesebben Command Framework
  - Megjegyzés:
    - Action nem támogatottá válhat (deprecated)
      - Egyelőre terv



# Akciók



- AKCIÓK DEFINÍCIÓJA
  - Kontribúció rendelhető hozzá
- Alkalmazás ablak kezeli az eseményeket
  - Akcióvá alakítja
  - Kontribúció futtatása - `run()` metódus

# Akció definíció

- Többféle kiterjesztés (helytől függően)
  - org.eclipse.ui.editorActions
    - Editorok kontribúcióihoz
    - Interfész: IEditorActionDelegate
  - org.eclipse.ui.viewActions
    - nézetek eszköztára és menüje
    - Interfész: IViewActionDelegate
  - org.eclipse.ui.popupMenus
    - máshol definiált felbukkanó menük kiegészítésére
    - Interfész: IEditorActionDelegate v. IViewActionDelegate v. IObjectActionDelegate

# Akció kontribúció

- `org.eclipse.jface.action.Action` leszármazottai
  - `run()`: az akció futását vezérlő metódus
  - attribútumai
    - megjelenés: `text`, `toolTipText`, `image`, stb.
    - működés: `enabled`, `checked`
- Gyakran az `Action` egy leszármazottját kell beilleszteni
  - Kontribúció helyétől függően
  - Probléma: több helyre ugyanazt az `Action`ot?

# Problémák

- Sokféle kiterjesztési pont
- Különböző interfész a különböző beillesztési helyekre
- Nincs elkülönítve a megjelenítés és a viselkedés
- Probléma:
  - azonos Action több helyre?
    - hasonló XML különböző helyeken

# Command Framework

- Eclipse 3.3 óta
  - Action-ök lecserélésére
  - Folyamatosan bővül
- Alapfogalmak
  - Command (*Parancs*)
    - Egy azonosító, mely kijelöl egy funkcionalitást.
  - Handler (*Kezelő*)
    - Végrehajtható kód, amely megfelel egy parancs végrehajtásának.
  - Contribution (*Kontribúció*)

# Command

- Extension point: `org.eclipse.ui.commands`
- Kötelező paraméterek
  - Azonosító
  - Név (felhasználói felületre)
- Opcionális paraméterek
  - Kategória
  - Alapértelmezett kezelő

# Command példa

```
<extension point="org.eclipse.ui.commands">  
  <command  
    description="Shows a Hello, world message."  
    id="hu.mit.bme.commanddemo.helloworld"  
    name="Display hello world">  
  </command>  
</extension>
```

# Handler

- Kiterjesztési pont: `org.eclipse.ui.handlers`
- Kötelező paraméterek:
  - Parancsazonosító
  - `IHandler` vagy `AbstractHandler` leszármazott osztály
- Végrehajtandó kód:
  - Handler osztály `execute (ExecutionEvent event)` metódusa
  - `ExecutionEvent`
    - végrehajtás környezeti információi
    - `HandlerUtil` osztály statikus metódusaival kiértékelhető



# Handlerek választása

- Egy commandhoz több handler is tartozhat
- Választani kell
  - activeWhen: alkalmazási feltétel (Core Expression - később)
  - enabledWhen: engedélyezési feltétel (Core Expression)
- Melyik handlert futtassuk?
  - activeWhen igaz
  - több lehetséges command esetén legspecifikusabb feltétel (később)
  - ha több egyforma specifikus igaz feltétel van, akkor parancs letiltva!

# Core Expression

- Feltételek leírása a plugin.xml-ben
- A kiértékeléshez nem kell betölteni a plugin-t (lazy loading)
- Lehetséges feltételek
  - Típusegyezés
  - Leszámlálási feltételek
  - Összehasonlítások
  - Logikai műveletek (és, vagy, ...)
  - ...

# Mire írhatunk fel feltételeket?

- Leggyakoribb feltételek
  - Aktív kontextus (activeContexts)
  - Aktív akciókészlet (activeActionSets)
  - Aktív shell (activeShell)
  - Aktív ablak (activeWorkbenchWindow)
  - Aktív editor (activeEditorId)
  - Aktív elem (activePartId)
  - Aktuális kijelölés (selection)
- A lista elemei felülről lefelé egyre specifikusabbak

# Core Expression példa

- A kijelölés tartalmaz IFile elemet

```
<with variable = "selection">  
  <iterate operator ="or">  
    <instanceof  
      value =  
      "org.eclipse.core.resources.IFile">  
    </instanceof>  
  </iterate>  
</with>
```

# Core Expression példa

- A JDT Java szerkesztője aktív

```
<with variable = "activeEditorId">  
  <equals value =  
    "org.eclipse.jdt.ui.CompilationUnitEditor">  
  </equals>  
</with>
```

# Handler példa

- Kezelő, amely akkor aktív, ha a Java JDT editor aktív

```
<extension point="org.eclipse.ui.handlers">
  <handler
    class=
      "hu.bme.mit.commanddemo.handlers.helloworld"
    commandId="hu.bme.mit.commanddemo.helloworld">
    <activeWhen><with variable = "activeEditorId">
      <equals
        value="org.eclipse.jdt.ui.CompilationUnitEditor">
        </equals>
      </with></activeWhen>
    </handler>
  </extension>
```

# Contribution

- Kiterjesztési pont: `org.eclipse.ui.menus`
- Kontribúció helyének kijelölése:
  - URL megadása
  - "[Scheme]:[ID]?[Query]"
    - Scheme
      - kontribúciótípus kijelölése
      - menu, popup, toolbar
    - ID
      - editor, view vagy eszköztár azonosítója
    - Query
      - hely precízebb meghatározása
      - pl. ismert kontribúció előtt/után

# Contribution

- Hozzáadott parancs tulajdonságai
  - Kötelező
    - ParancsID
  - Opcionális
    - ID (Query-khez)
    - Címke
    - Ikonok (nem így célszerű!)
  - Láthatósági feltétel hozzáadható
    - Core Expression
    - visibleWhen gyerekelem



# Contribution példa

- Eszköztár gomb felvétele a fő eszköztárra

```
<extension point="org.eclipse.ui.menus">
  <menuContribution      locationURI =
    "toolbar:org.eclipse.ui.main.toolbar">
    <toolbar id="hu.bme.mit.commanddemo.toolbar1">
      <command commandId=
        "hu.bme.mit.commanddemo.helloworld"
        id="hu.bme.mit.commanddemo.helloworldInToolbar"
        label="Hello, world!"
        style="push"
        tooltip="Displays a hello, world
dialog.">
      </command>
    </toolbar>
  </menuContribution>
</extension>
```

# Contribution példa

- Eszköztár gomb felvétele a fő eszk

URL megadása

```
<extension point="org.eclipse.ui.actionURL">
  <menuContribution id="org.eclipse.ui.actionURL"
    toolbar:org.eclipse.ui.main.toolbar">
    <toolbar id="hu.bme.mit.commanddemo.toolbar1">
      <command commandId=
        "hu.bme.mit.commanddemo.helloworld"
        id="hu.bme.mit.commanddemo.helloworldInToolbar"
        label="Hello, world!"
        style="push"
        tooltip="Displays a hello, world
dialog.">
      </command>
    </toolbar>
  </menuContribution>
</extension>
```

# Contribution példa

- Eszköztár gomb felvétele a fő eszköztárra

```
<extension point="org.eclipse.ui.menus">  
  <menuContribution locationURI =  
    "toolbar:org.eclipse.ui.main.toolbar">  
    <toolbar id="hu.bme.mit.commanddemo.toolbar1">  
      <command commandId=  
"hu.bme.mit.commanddemo.hello" id="hu.bme.mit.commanddemo.hello"  
        label="Hello, world" style="push"  
        tooltip="Displays a hello, world  
dialog.">  
        </command>  
      </toolbar>  
    </menuContribution>  
</extension>
```

Saját toolbar  
felvétele (szükséges - toolbar"  
itt!)

# Contribution példa

- Eszköztár gomb felvétele a fő eszköztárra

Parancs  
megadása

```
<extension point base="org.eclipse.ui.menus">
  <menuContribution locationURI =
    "toolbarContribution.se.ui.main.toolbar">
    <toolbar id="hu.bme.mit.commanddemo.toolbar1">
      <command commandId=
        "hu.bme.mit.commanddemo.helloworld"
        id="hu.bme.mit.commanddemo.helloworldInToolbar"
        label="Hello, world!"
        style="push"
        tooltip="Displays a hello, world
    dialog.">
      </command>
    </toolbar>
  </menuContribution>
</extension>
```

# Ikon parancsokhoz

- Kiterjesztési pont: `org.eclipse.ui.commandImages`
- Kötelező paraméterek
  - `commandID`
  - `icon`
- Cél:
  - Ikon hozzárendelése a parancshoz
- Kép ne kontribúcióhoz tartozzon!
  - Egy parancshoz lehet több kontribúció is!
  - Esetleg kontribúció már nem is éri el a képet (más plug-in)

# commandImages példa

```
<image
```

```
  commandId="hu.bme.mit.commanddemo.helloworld"  
  icon="icons/sample.gif">
```

```
</image>
```

# Gyorsbillentyű hozzávétele

- Kiterjesztési pont: `org.eclipse.ui.bindings`
- Cél: gyorsbillentyűk hozzárendelése a parancsokhoz
- Kötelező paraméterek:
  - Sequence: a tényleges billentyűszekvencia
  - SchemeID: billentyűkiosztás sablon választása
  - ContextID: hol érvényes a gyorsbillentyű
  - CommandID: aktiválandó parancs
- Opcionális paraméter:
  - Platform: milyen rendszeren működjön a gyorsbillentyű
  - Locale: milyen nyelven működjön

# Gyorsbillentyű példa

```
<key
sequence =
  "M2+F5"
commandId =
  "hu.optxware.rcpdemo.list"
schemeId =
  "org.eclipse.ui.defaultAcceleratorCon
figuration"
contextId =
  "org.eclipse.ui.contexts.dialog" />
```

M2: módosító gomb;  
pontos értéke  
platformfüggő



# Command Framework

- Parancsok hozzáadása a GUI-hoz
- Kontribúció és parancs szigorú elkülönítése
  - négyféle kiterjesztés
- Core Expression feltételek a kapcsoláshoz
  - plugin.xml -> lazy loading támogatás