Advanced Rendering

Exploiting Eclipse 4 advanced rendering

Sopot Çela

- Application R&D at Petrotechnical Data Systems

- Committer
  - Eclipse e4
  - Vaaclipse

- Contributor
  - Platform UI

Agenda

- Customize default SWT renderer
  - Minor customization
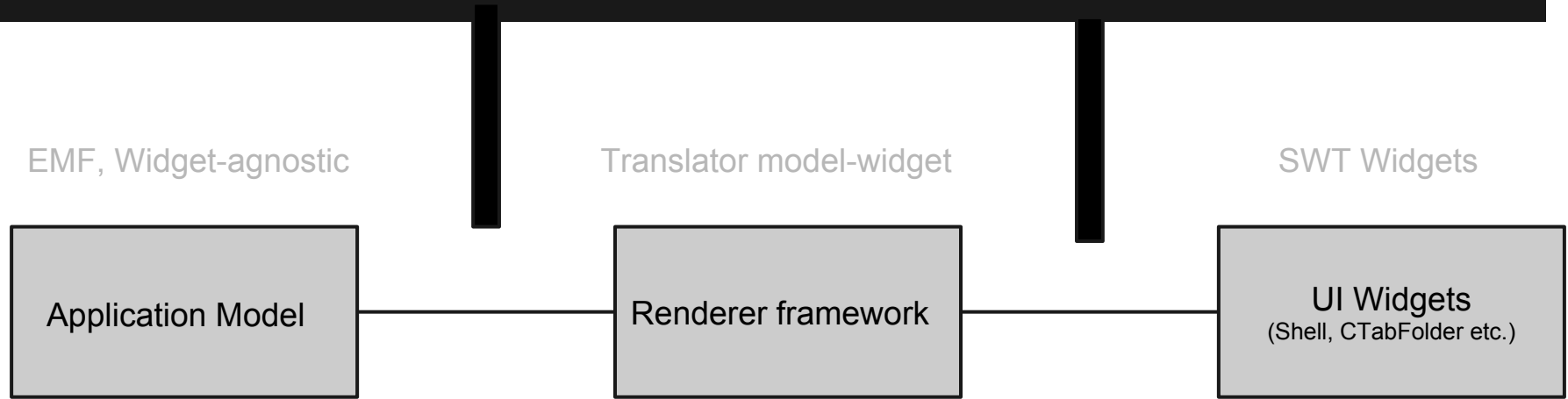 - Major customization

Agenda

- Major overhaul of the renderers
  - Vaaclipse renderer
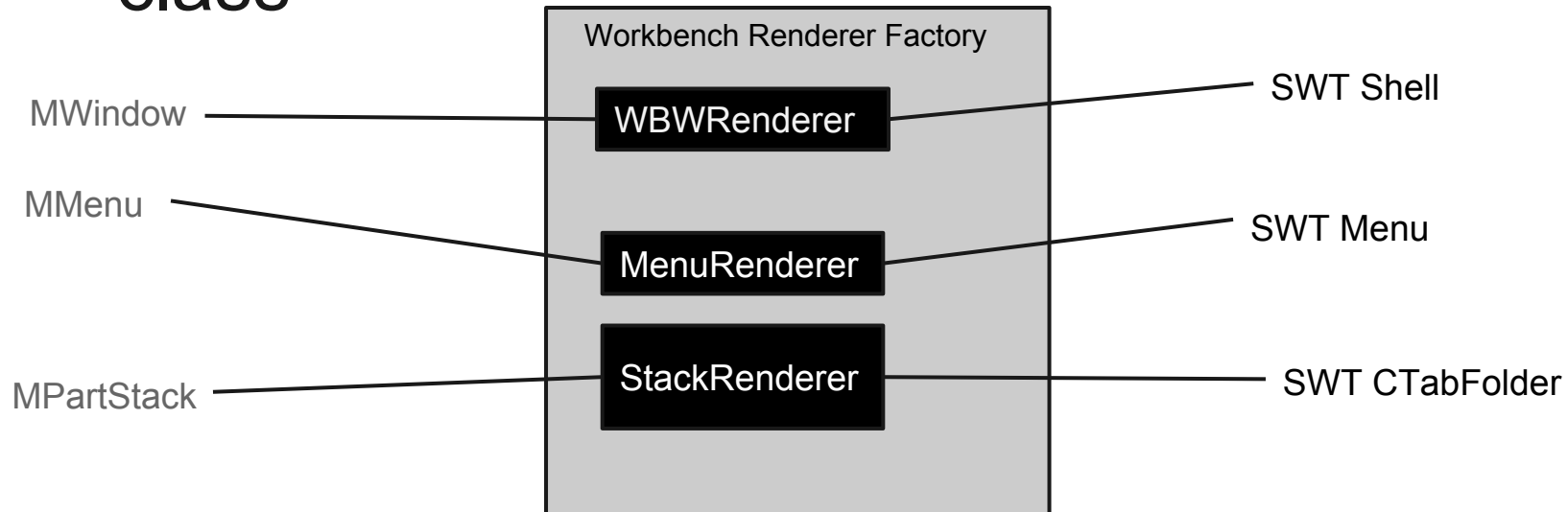 - E(fx)clipse renderer

Agenda

- Uber overhaul of the renderers
  - Bling and GPU-powered renderers (demo)

Renderer framework

EMF, Widget-agnostic

Translator model-widget

SWT Widgets

| Application Model | Renderer framework | UI Widgets (Shell, CTabFolder etc.) |

Renderer Framework

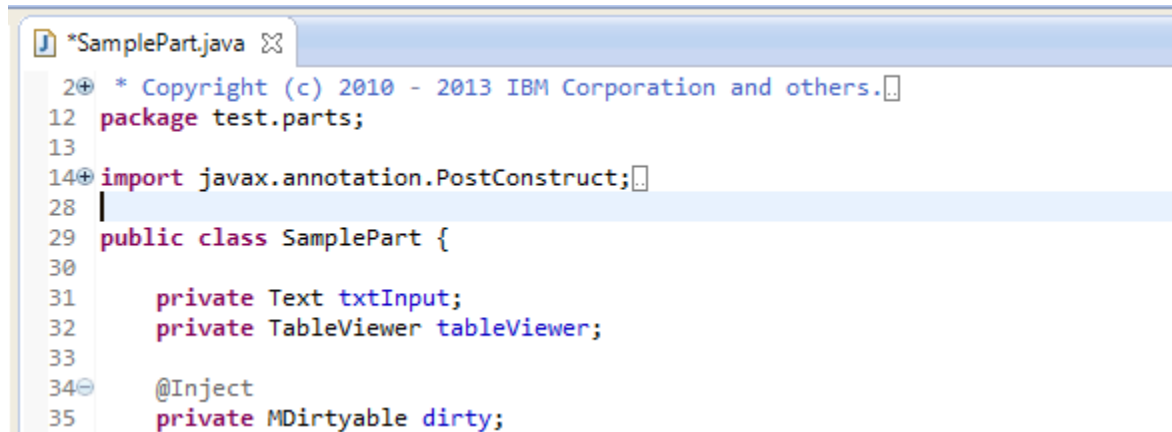- Every UI model element has a renderer class

The "Dispatcher"

```java
public AbstractPartRenderer getRenderer(MUIElement uiElement, Object parent) {
    if (uiElement instanceof MArea) {
        if (areaRenderer == null) {
            areaRenderer = new AreaRenderer();
            initRenderer(areaRenderer);
        }
        return areaRenderer;
    } else if (uiElement instanceof MPart) {
        if (contributedPartRenderer == null) {
            contributedPartRenderer = new ContributedPartRenderer();
            initRenderer(contributedPartRenderer);
        }
        return contributedPartRenderer;
    } else if (uiElement instanceof MMenu) {
        if (menuRenderer == null) {
            menuRenderer = new MenuManagerRenderer();
            initRenderer(menuRenderer);
        }
        return menuRenderer;
    } else if (uiElement instanceof MToolBar) {
        if (toolbarRenderer == null) {
            toolbarRenderer = new ToolBarManagerRenderer();
            initRenderer(toolbarRenderer);
        }
        return toolbarRenderer;
    } else if (uiElement instanceof MMenuSeparator
            || uiElement instanceof MToolBarSeparator) {
        if (separatorRenderer == null) {
            separatorRenderer = new SeparatorRenderer();
            initRenderer(separatorRenderer);
        }
        return separatorRenderer;
    } else if (uiElement instanceof MPlaceholder) {
```

Customized SWT Renderer

- Let's change what is commonly defined as a 'dirty' editor


```java
 2⊕  * Copyright (c) 2010 - 2013 IBM Corporation and others.
12  package test.parts;
13
14⊕ import javax.annotation.PostConstruct;
28
29  public class SamplePart {
30
31      private Text txtInput;
32      private TableViewer tableViewer;
33
34⊖     @Inject
35      private MDirtyable dirty;
```

Hands On

Exploiting Eclipse 4 advanced rendering

Hands On, Demo

Vaaclipse, e(fx)clipse, Bling Engine