

Eclipse Rich Client Platform (RCP)

Lars M. Kristensen
Michael Westergaard

Department of Computer Science
University of Aarhus

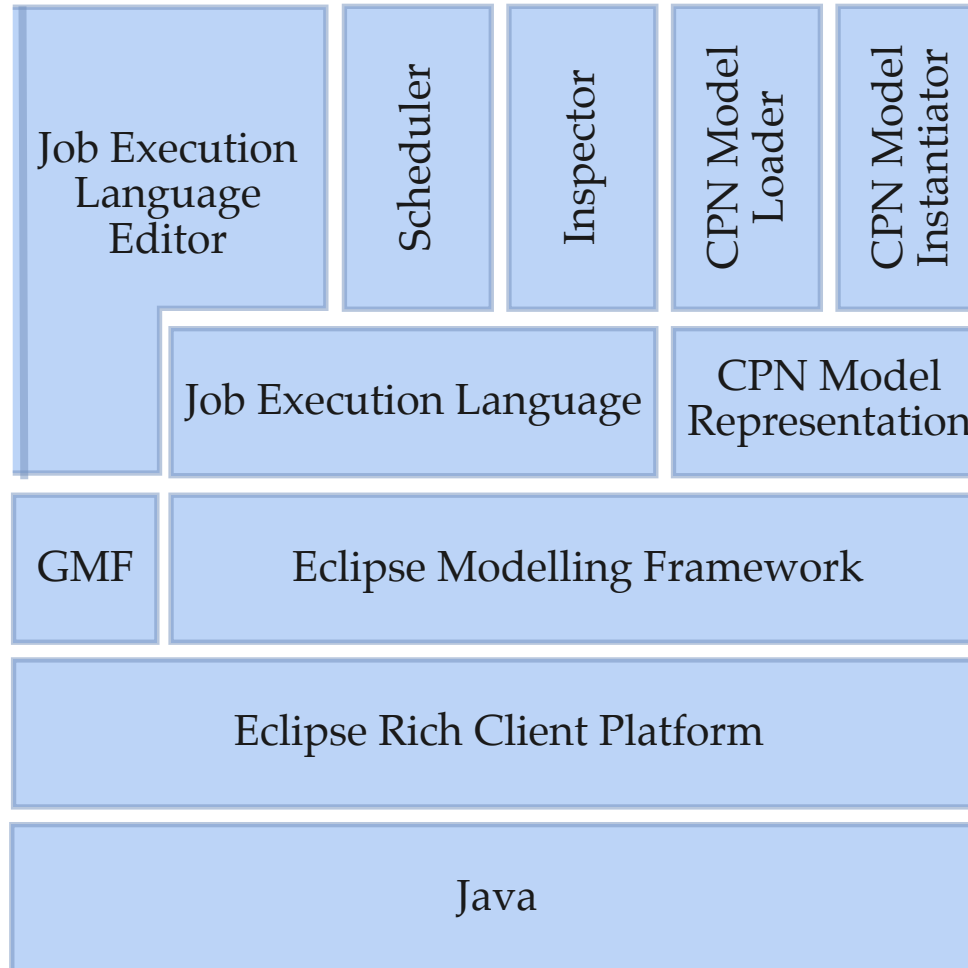


UNIVERSITY OF AARHUS

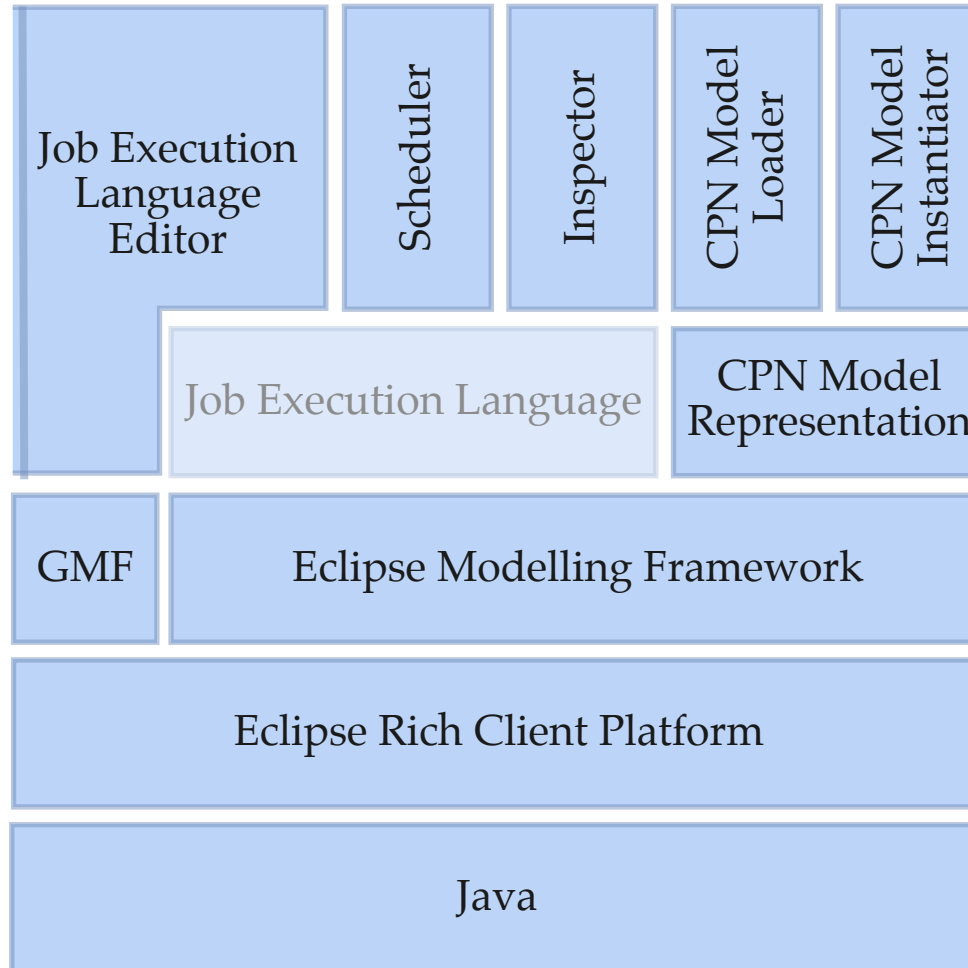
Department of Computer Science

Michael Westergaard

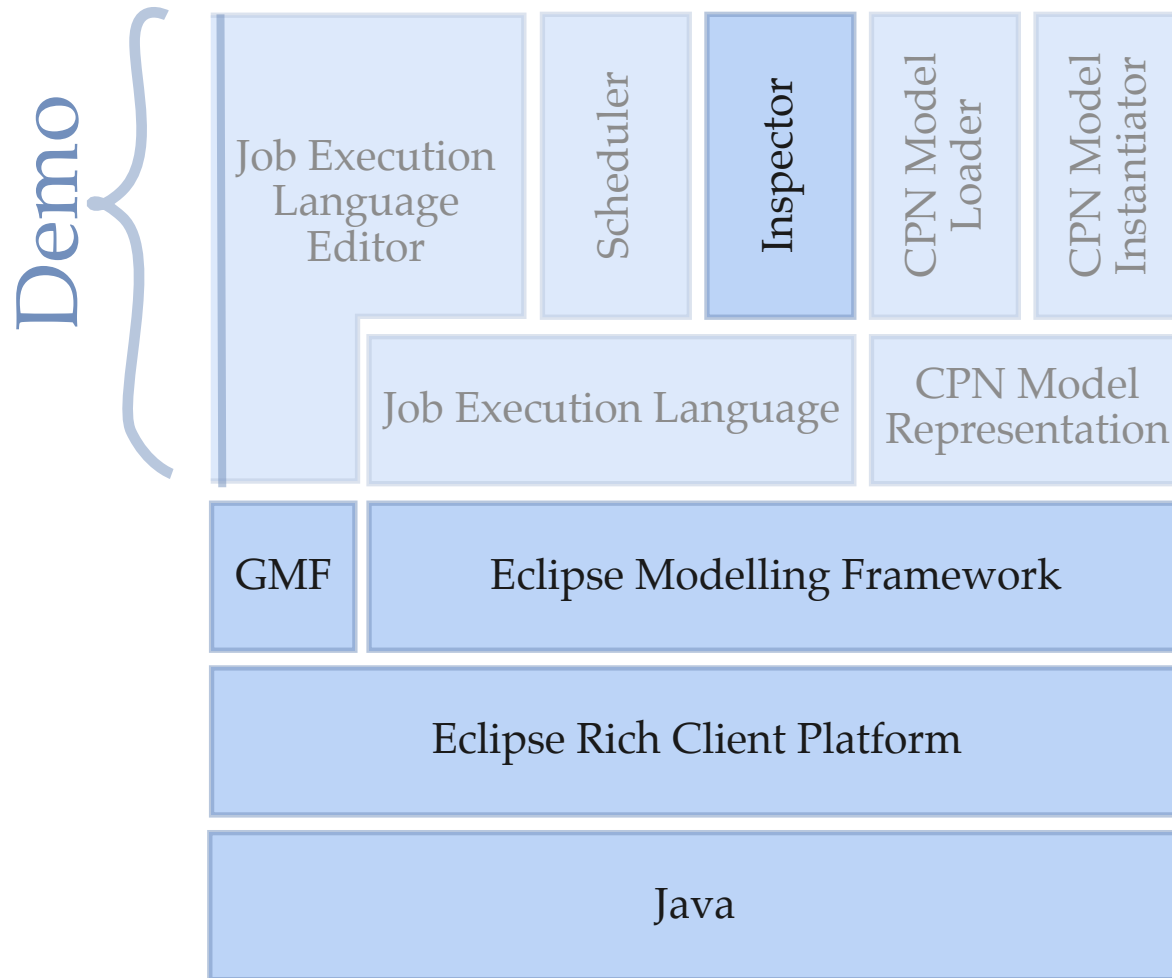
Architecture



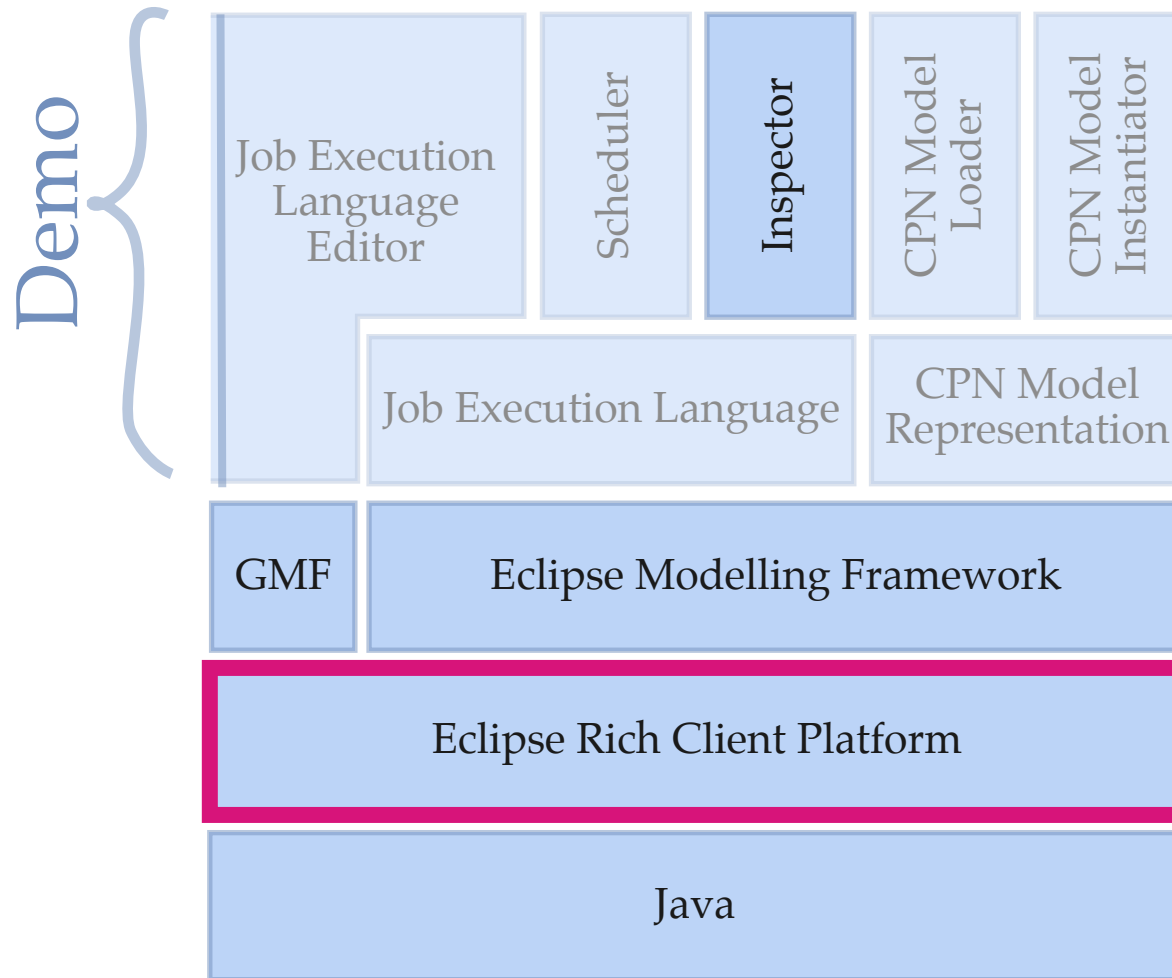
Architecture



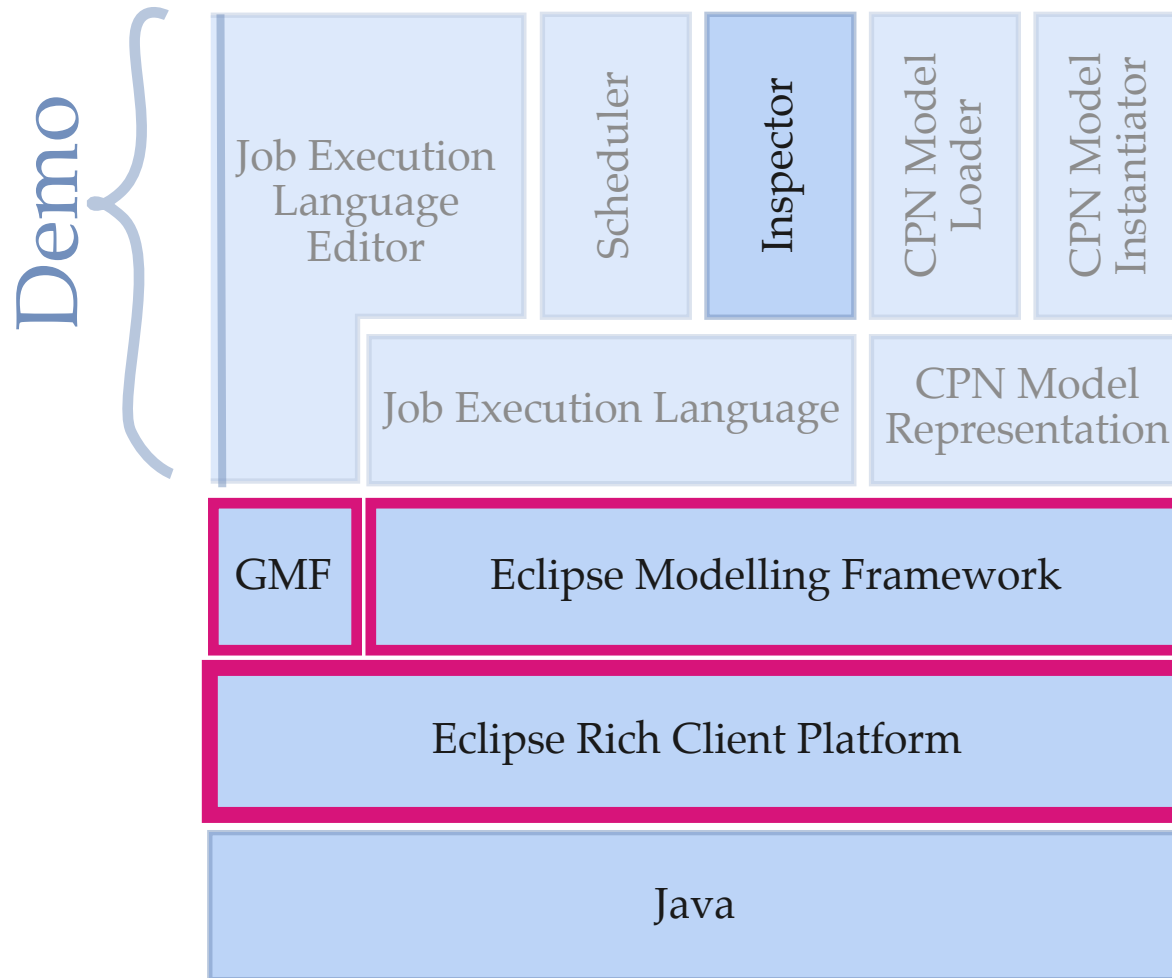
Architecture



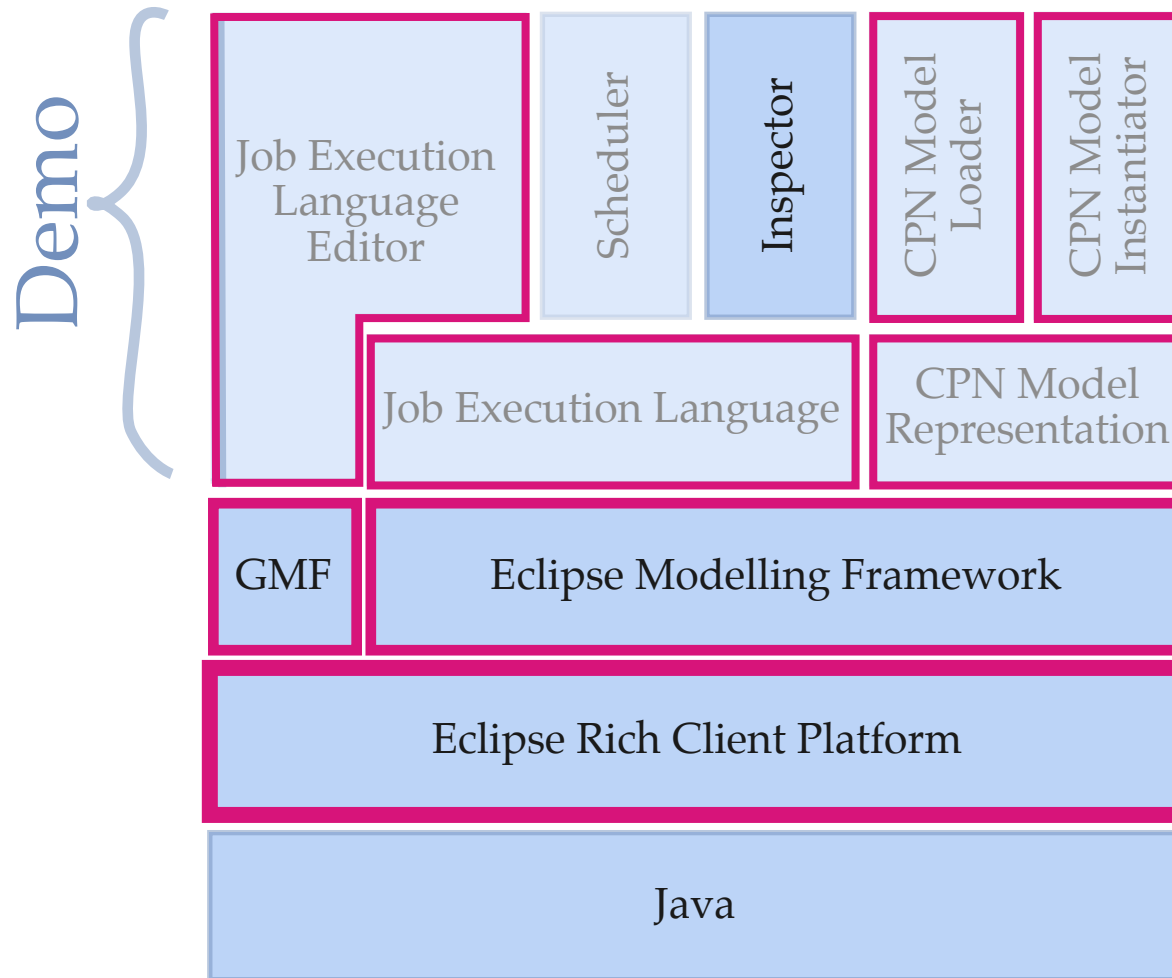
Architecture



Architecture



Architecture



Eclipse

- **Eclipse is well-known as a Java Integrated Development Environment (IDE)**
- **Eclipse is also the primary user of an underlying application platform**
- **The underlying platform can be used for other purposes**



Eclipse Applications

- **SWT applications**
 - Similar to standard Java (Swing) applications
 - Provides no integration with the Eclipse frameworks
- **Eclipse plug-in**
 - Integrated into the standard Eclipse IDE
 - Must be deployed within an Eclipse application
- **Eclipse Rich Client Platform application**
 - Bundles many plug-ins
 - Deployed as separate executable application



Built-in Frameworks

- **RCP features: update service, help, preferences, user interface, persistence, ...**
- **Eclipse provides a myriad of frameworks**
 - Eclipse Modelling Framework
 - Graphical Editing Framework
 - Reporting Framework
- **Most of the frameworks are quite well-thought-out, well-tested, and encourages best-practices**
- **We can focus on creating a great verification tool**



Eclipse Plug-ins

- **Plug-ins bundles well-defined functionality**
- **Plug-ins can optionally extend other plug-ins**
- **Plug-ins can optionally allow other plug-ins to extend their functionality**
- **RCP applications consist of a bundle of plug-ins and new plug-ins can be added**
- ***This is perfect for an extensible tool for formal verification***



Eclipse Modelling Framework (EMF)

- **Framework for easily generating implementations of object models**
- **Can read and generate Java interfaces, XML Schema, and UML**
- **Generated code is correct, which eliminates need for testing**
- **We get serialisation to XML, observer architecture, and several other niceties for free**



Graphical Modelling Framework (GMF)

- **Can generate graphical editors for EMF models**
- **The editor is detached from the domain model**
- **The domain model is automatically updated to reflect what is shown in the editor**
- **It is possible to generate a rough editor in hours, which can later be refined**



ASAP and RCP

The screenshot displays the ASAP and RCP software interface. The top window shows the 'Deadlock%20Freeness.execution_diagram' with a flowchart: 'Strina inout' -> 'Instantiate Model' -> 'Deadlock freeness' -> 'Safetv Checke' -> 'Generate report'. The middle window shows the 'Deadlock Freeness Top.execution_diagram' with a flowchart: 'Pipe' -> 'Oueue' -> 'Hash Storage' -> 'Waitina Set Exoloration' -> 'Simple Trace Exoloration' -> 'Safetv Checker'. The bottom window is the 'Console' showing simulator output:

```
Simulator Console
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
4
-
```

The interface includes a 'Projects' sidebar with a tree view of 'Demo' containing 'jobs' and 'models' folders. The 'models' folder is expanded to show 'erdp.model' and its sub-items: 'AssignNewPrefix', 'Config', 'EdgeRouter', 'ERDiscardPrefixes', 'ERDP', 'Gateway', 'GW_ER_Link', 'CWDiscardPrefixes', 'NoNewAssignPrefix', 'ProcessRS', 'ProcessUnsolicitedRA', 'ReceiveSolicitedRA', 'SendRouterSolicitation', and 'SendUnsolicitedRA'. A 'Palette' on the right lists 'Select', 'Zoom', 'Note', 'Connection', and 'Macro'. The 'Console' window has a 'Simulator Console' tab and a scrollable text area.



ASAP and RCP

The screenshot displays the Rich Client Platform (RCP) interface. On the left is a project tree with folders for 'Demo', 'jobs', and 'models'. The 'models' folder contains an 'erdp.model' sub-folder with various components like 'AssignNewPrefix', 'Config', 'EdgeRouter', etc. The main workspace is divided into two execution diagrams. The top diagram, titled 'Deadlock%20Freeness.execution_diagram', shows a flow from 'Strina inout' to 'Instantiate Model', which then branches to 'Deadlock freeness' and 'Safetv Checke', both leading to 'Generate report'. The bottom diagram, titled 'Deadlock Freeness Top.execution_diagram', shows a flow from 'Queue' to 'Hash Storage', 'Waitina Set Exoloration', and 'Simple Trace Exoloration', all of which lead to 'Safetv Checker'. A 'Pipe' component is also connected to 'Queue' and 'Safetv Checker'. A 'Palette' on the right side of the workspace contains options for 'Select', 'Zoom', 'Note', 'Connection', and 'Macro'. At the bottom, a 'Console' window shows the 'Simulator Console' output with three lines of log messages and a line number '4'.

Rich Client Platform

```
Simulator Console
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
4
-
```



ASAP and RCP

The screenshot displays the ASAP and RCP software interface. The top window shows the 'Deadlock%20Freeness.execution_diagram' with a flowchart: 'Strina inout' -> 'Instantiate Model' -> 'Deadlock freeness' -> 'Safetv Checke' -> 'Generate report'. The middle window shows the 'Deadlock Freeness Top.execution_diagram' with a flowchart: 'Pipe' -> 'Oueue' -> 'Hash Storage' -> 'Waitina Set Exoloration' -> 'Simple Trace Exoloration' -> 'Safetv Checker'. The bottom window is the 'Console' showing simulator output:

```
Simulator Console
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
4
-
```

The interface includes a 'Projects' sidebar with a tree view of 'Demo' containing 'jobs' and 'models' folders. The 'models' folder is expanded to show 'erdp.model' and its sub-items: 'AssignNewPrefix', 'Config', 'EdgeRouter', 'ERDiscardPrefixes', 'ERDP', 'Gateway', 'GW_ER_Link', 'CWDiscardPrefixes', 'NoNewAssignPrefix', 'ProcessRS', 'ProcessUnsolicitedRA', 'ReceiveSolicitedRA', 'SendRouterSolicitation', and 'SendUnsolicitedRA'. A 'Palette' on the right lists 'Select', 'Zoom', 'Note', 'Connection', and 'Macro'. The bottom right corner features a 'Palette' button and the name 'estergaard'.



ASAP and RCP

The screenshot displays the Eclipse Modelling Framework (EMF) interface. The top toolbar includes standard editing tools and a font size of 9. The left sidebar shows a project tree with a 'Demo' folder containing 'jobs' and 'models'. The 'models' folder is expanded to show 'erdp.model' with various components like 'AssignNewPrefix', 'Config', 'EdgeRouter', etc. The main workspace is divided into two execution diagram views. The top view, 'Deadlock%20Freeness.execution_diagram', shows a high-level flow with nodes like 'Deadlock freeness', 'Safety Checker', and 'generate report'. The bottom view, 'Deadlock Freeness Top.execution_diagram', provides a more detailed state transition diagram with nodes such as 'Queue', 'Hash Storage', 'Waiting Set Exploration', 'Simple Trace Exploration', 'Pipe', and 'Safety Checker'. The bottom panel is a 'Console' window titled 'Simulator Console' showing three lines of red text: `../compiler/TopLevel/interact/evalloop.sml:296.17-296.20`, `../compiler/TopLevel/interact/evalloop.sml:44.55`, and `../compiler/TopLevel/interact/evalloop.sml:66.19-66.27`. A blue box with the text 'Eclipse Modelling Framework' is overlaid on the top diagram.



ASAP and RCP

The screenshot displays the Eclipse Modelling Framework (EMF) interface. The top toolbar includes standard editing tools and a zoom level of 100%. The left sidebar shows a project tree with a 'Demo' folder containing 'Jobs' and 'models'. The 'models' folder is expanded to show a 'CPN' sub-folder with an 'erdp.model' file. The main workspace is divided into two views: the top view shows an 'execution diagram' for 'Deadlock freeness', and the bottom view shows a 'CPN Model Representation' with components like 'Pipe', 'Hash Storage', 'Queue', 'Waiting Set Exploration', 'Simple Trace Exploration', and 'Safety Checker'. A 'Console' window at the bottom displays the 'Simulator Console' output, showing three lines of red text: `../compiler/TopLevel/interact/evalloop.sml:296.17-296.20`, `../compiler/TopLevel/interact/evalloop.sml:44.55`, and `../compiler/TopLevel/interact/evalloop.sml:66.19-66.27`. A 'Palette' on the right side offers actions like 'Select', 'Zoom', 'Note', 'Connection', and 'Macro'.

Eclipse Modelling Framework

Job Execution Language & CPN Model Representation

```
Simulator Console
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
```



ASAP and RCP

The screenshot displays the ASAP and RCP software interface. The top window shows the 'Deadlock%20Freeness.execution_diagram' with a flowchart: 'Strina inout' -> 'Instantiate Model' -> 'Deadlock freeness' (top node) -> 'Safetv Checke' -> 'Generate report'. The middle window shows the 'Deadlock Freeness Top.execution_diagram' with a flowchart: 'Pipe' -> 'Oueue' -> 'Hash Storage' -> 'Waitina Set Exoloration' -> 'Simple Trace Exoloration' -> 'Safetv Checker'. The bottom window is the 'Console' showing simulator output:

```
Simulator Console
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
4
-
```

The interface includes a 'Projects' sidebar with a tree view of 'Demo' containing 'jobs' and 'models' folders. The 'models' folder contains 'erdp.model' and various sub-models like 'AssignNewPrefix', 'Config', 'EdgeRouter', etc. A 'Palette' on the right offers actions like 'Select', 'Zoom', 'Note', 'Connection', and 'Macro'. The top toolbar includes standard editing tools and a 'Tahoma' font setting.



ASAP and RCP

The screenshot displays the CPN Model Loader application interface. On the left, a 'Projects' pane shows a tree structure under 'Demo' with subfolders 'jobs' and 'models'. The 'models' folder is expanded, listing various CPN models such as 'erdp.model', 'AssignNewPrefix', 'Config', 'EdgeRouter', 'ERDiscardPrefixes', 'ERDP', 'Gateway', 'GW_ER_Link', 'CWDiscardPrefixes', 'NoNewAssignPrefix', 'ProcessRS', 'ProcessUnsolicitedRA', 'ReceiveSolicitedRA', 'SendRouterSolicitation', and 'SendUnsolicitedRA'. The main workspace is divided into two execution diagrams. The top diagram, titled 'Deadlock%20Freeness.execution_diagram', shows a flow from 'Strina inout' to 'Instantiate Model', which then branches to 'Deadlock freeness' and 'Safetv Checke', both leading to 'Generate report'. The bottom diagram, titled 'Deadlock Freeness Top.execution_diagram', shows a flow from 'Oueue' to 'Hash Storage', which then branches to 'Waiting Set Exploration' and 'Simple Trace Exploration', both leading to 'Safetv Checker'. A 'Pipe' node is also connected to 'Waiting Set Exploration'. A blue semi-transparent box with the text 'CPN Model Loader' is overlaid on the bottom diagram. At the bottom, a 'Console' window displays the following text: 'Simulator Console', './compiler/TopLevel/interact/evalloop.sml:296.17-296.20', './compiler/TopLevel/interact/evalloop.sml:44.55', and './compiler/TopLevel/interact/evalloop.sml:66.19-66.27'. The number '4' is visible on the left side of the console output.



ASAP and RCP

The screenshot displays a software development environment with the following components:

- Toolbar:** Includes standard editing tools (undo, redo, copy, paste) and a font size of 9.
- Projects Panel:** Shows a tree view with folders for 'Demo', 'jobs', and 'models'. Under 'models', the 'erdp.model' folder is expanded, listing various components like 'AssignNewPrefix', 'Config', 'EdgeRouter', etc.
- Diagram View (Top):** Titled 'Deadlock%20Freeness.execution_diagram', showing a flowchart with nodes: 'Strina inout', 'Instantiate Model', 'Deadlock freeness', 'Safetv Checke', and 'Generate report'.
- Diagram View (Bottom):** Titled 'Deadlock Freeness Top.execution_diagram', showing a more complex flowchart with nodes: 'Oueue', 'Hash Storage', 'Pipe', 'Waitina Set Exoloration', 'Simple Trace Exoloration', and 'Safetv Checker'.
- Console:** A 'Simulator Console' window showing the following output:

```
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
```
- Palettes:** A 'Palette' on the right side of the top diagram and another 'Palette' on the right side of the bottom diagram, both containing tools like 'Select', 'Zoom', 'Note', 'Connection', and 'Macro'.



ASAP and RCP

The screenshot displays the CPN Model Instantiator software interface. On the left, a 'Projects' pane shows a tree structure under 'Demo' with folders for 'jobs' and 'models'. The 'models' folder contains a sub-folder 'CPN' with a file 'erdp.model'. Below it are various model components like 'AssignNewPrefix', 'Config', 'EdgeRouter', etc. The main workspace is divided into two execution diagrams. The top diagram, titled 'Deadlock%20Freeness.execution_diagram', shows a flow from 'Strina inout' to 'Instantiate Model', which then branches to 'Deadlock freeness' and 'Safetv Checke', both leading to 'Generate report'. The bottom diagram, titled 'Deadlock Freeness Top.execution_diagram', shows a flow from 'Pipe' to 'Oueue', 'Hash Storage', and 'Waitina Set Exoloration'. 'Oueue' and 'Hash Storage' also lead to 'Waitina Set Exoloration', which then leads to 'Simple Trace Exoloration' and finally to 'Safetv Checker'. A 'Console' window at the bottom shows the 'Simulator Console' output with file paths and line numbers:

```
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
```

CPN Model Instantiator



ASAP and RCP

The screenshot displays a software development environment with the following components:

- Toolbar:** Includes standard editing tools and a font size of 9.
- Projects Panel:** Shows a tree structure under 'Demo' with folders for 'jobs' and 'models'. The 'models' folder contains 'erdp.model' and its sub-components.
- Execution Diagrams:**
 - Deadlock%20Freeness.execution_diagram:** A flowchart starting with 'Strina inout', leading to 'Instantiate Model', which then branches to 'Deadlock freeness' and 'Safetv Checke', both leading to 'Generate report'.
 - Deadlock Freeness Top.execution_diagram:** A flowchart starting with 'Oueue', leading to 'Hash Storage', which then leads to 'Waitina Set Exoloration', which leads to 'Simple Trace Exoloration', and finally to 'Safetv Checker'. A 'Pipe' component is also connected to 'Oueue' and 'Safetv Checker'.
- Console:** Shows the 'Simulator Console' with the following output:

```
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
```



ASAP and RCP

The screenshot displays a software interface for a Graphical Modelling Framework. The window title is "Tahoma" and the zoom level is 100%. A large text overlay reads "Graphical Modelling Framework".

Project Tree (Left):

- Demo
 - jobs
 - Deadlock%20Freeness.exe
 - Deadlock Freeness.execu
 - Deadlock Freeness Top.e
 - models
 - CPN erdp.model
 - AssignNewPrefix
 - Config
 - EdgeRouter
 - ERDiscardPrefixes
 - ERDP
 - Gateway
 - GW_ER_Link
 - CWDiscardPrefixes
 - NoNewAssignPrefix
 - ProcessRS
 - ProcessUnsolicitedRA
 - ReceiveSolicitedRA
 - SendRouterSolicitation
 - SendUnsolicitedRA
 - properties

Flowchart (Top): A flowchart titled "Deadlock freeness" showing a sequence of steps: "Strina inout" → "Instantiate Model" → "Safety Checke" → "Generate report".

Flowchart (Bottom): A flowchart titled "Deadlock Freeness Top.execution_diagram" showing a sequence of steps: "Pipe" → "Hash Storage" → "Waitina Set Exoloration" → "Simple Trace Exoloration" → "Safety Checker".

Console (Bottom):

```
Simulator Console
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
4
-
```



ASAP and RCP

Graphical Modelling Framework

Job Execution Language Editor

The screenshot displays the ASAP and RCP software interface. The top window, titled 'Tahoma', shows a graphical modelling framework with a flowchart. The flowchart starts with 'String input', followed by 'Instantiate Model', which branches into 'Deadlock freeness' and 'Safety Checker'. 'Safety Checker' leads to 'Generate report'. The bottom window, titled 'Deadlock Freeness Top.execution_diagram', shows a more complex flowchart. It starts with 'Pipe', which branches into 'Queue' and 'Hash Storage'. 'Queue' leads to 'Waitina Set Exoloration', which then leads to 'Simple Trace Exoloration'. 'Hash Storage' also leads to 'Simple Trace Exoloration'. 'Simple Trace Exoloration' leads to 'Safety Checker'. The interface includes a 'Palette' on the right with options like 'Select', 'Zoom', 'Note', 'Connection', and 'Macro'. A 'Console' window at the bottom shows the 'Simulator Console' output with the following text:

```
Simulator Console
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
```



ASAP and RCP

The screenshot displays the ASAP and RCP software interface. The top window shows the 'Deadlock%20Freeness.execution_diagram' with a flowchart: 'Strina inout' -> 'Instantiate Model' -> 'Deadlock freeness' (top node) -> 'Safetv Checke' -> 'Generate report'. The middle window shows the 'Deadlock Freeness Top.execution_diagram' with a flowchart: 'Pipe' -> 'Oueue' -> 'Hash Storage' -> 'Waitina Set Exoloration' -> 'Simple Trace Exoloration' -> 'Safetv Checker'. The bottom window is the 'Console' showing the 'Simulator Console' output:

```
Simulator Console
../compiler/TopLevel/interact/evalloop.sml:296.17-296.20
../compiler/TopLevel/interact/evalloop.sml:44.55
../compiler/TopLevel/interact/evalloop.sml:66.19-66.27
4
-
```

The interface includes a 'Projects' sidebar with a tree view of 'Demo' containing 'jobs' and 'models' folders. The 'models' folder contains 'erdp.model' and various sub-models like 'AssignNewPrefix', 'Config', 'EdgeRouter', etc. A 'Palette' on the right offers actions like 'Select', 'Zoom', 'Note', 'Connection', and 'Macro'. The top toolbar includes standard editing tools and a font size of 9.



Evaluation of RCP

■ Disadvantages

- Steeper learning curve – later but better results
- Sometimes it may be overkill – a lot of tweaking options are available
- Less flexibility

■ Advantages

- Can focus on task at hand (verification tool)
- Get a lot of functionality for free
- Can be integrated with Eclipse and can integrate Eclipse tools (say we want to incorporate versioning in ASAP)
- Less flexibility

