

On the shoulders of giants



Harnessing the Power of Eclipse
Enterprise Ecosystem

What is he going to talk about?

To show you how
Eclipse makes **you**
more **productive**

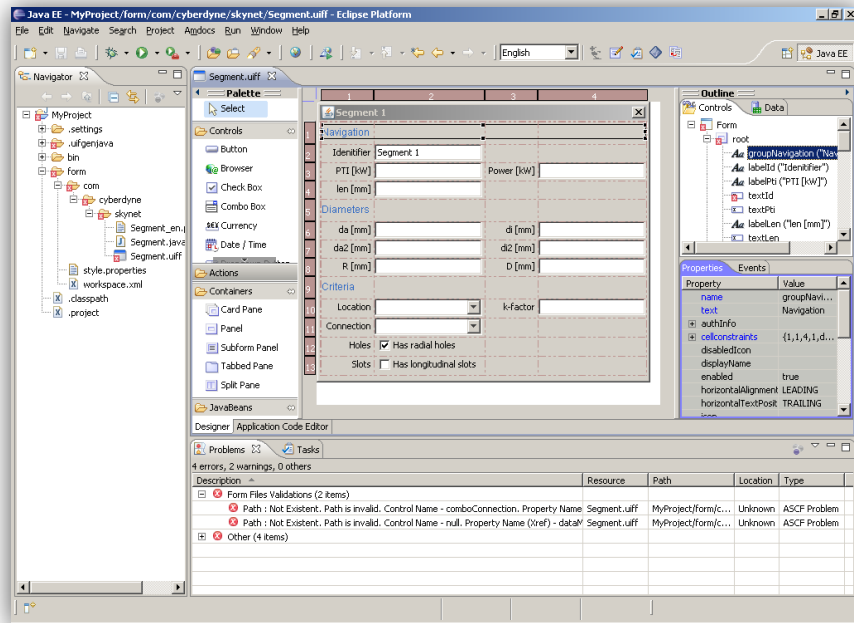
Why should I care?

Less code to write,
test and support

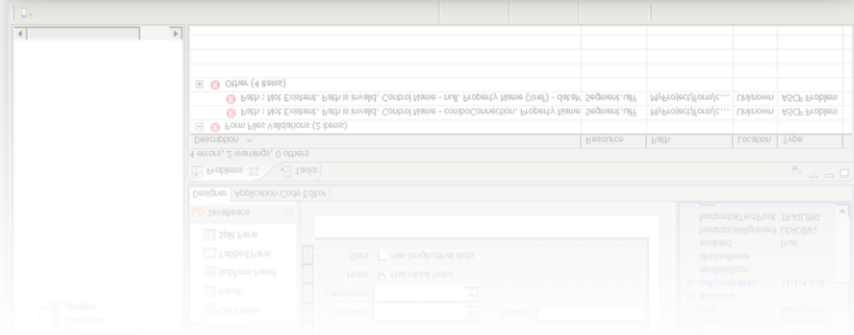
And then you can retire

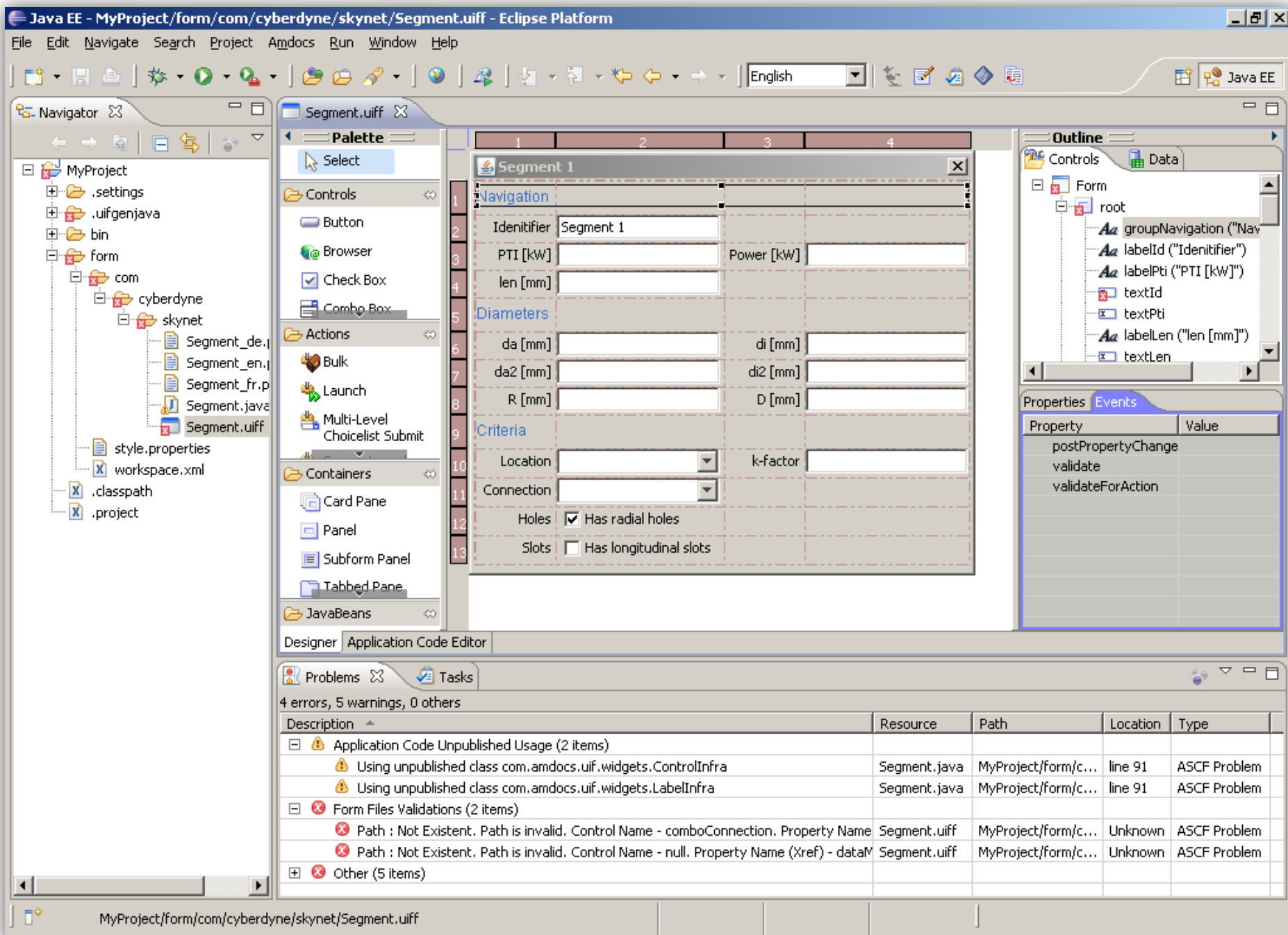


What is he going to talk about?



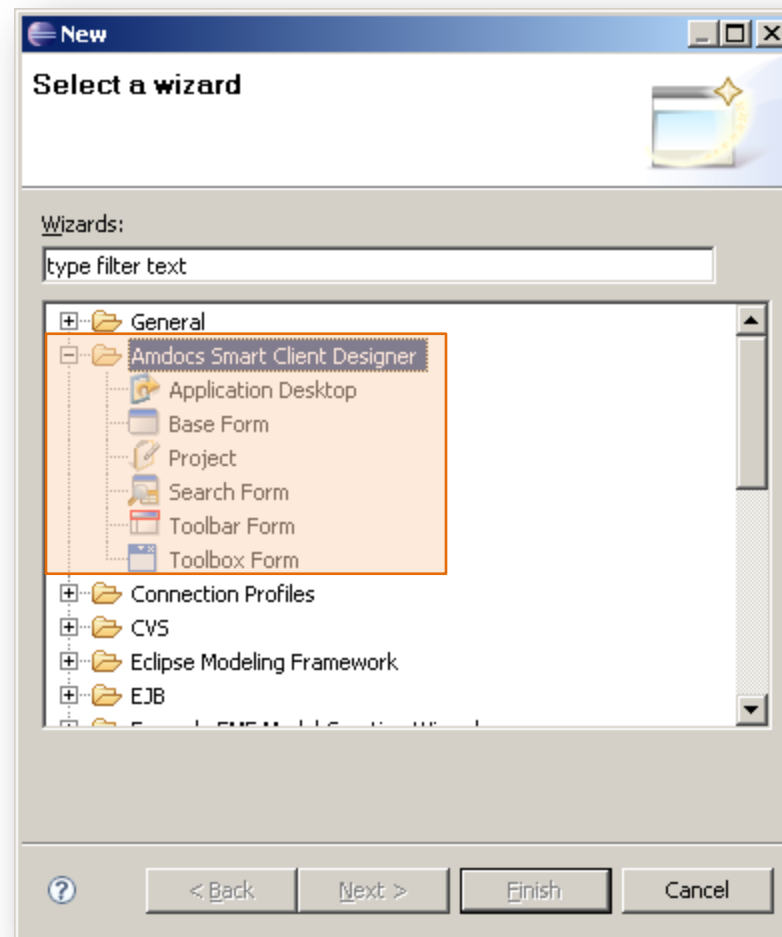
Amdocs Smart Client Designer



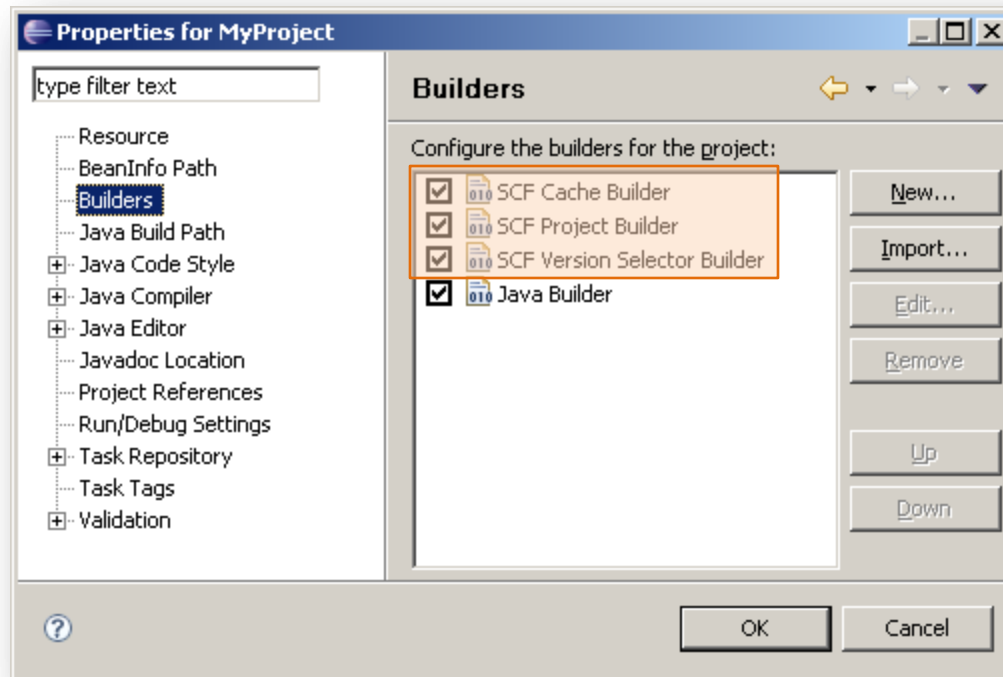


Provides familiar
experience

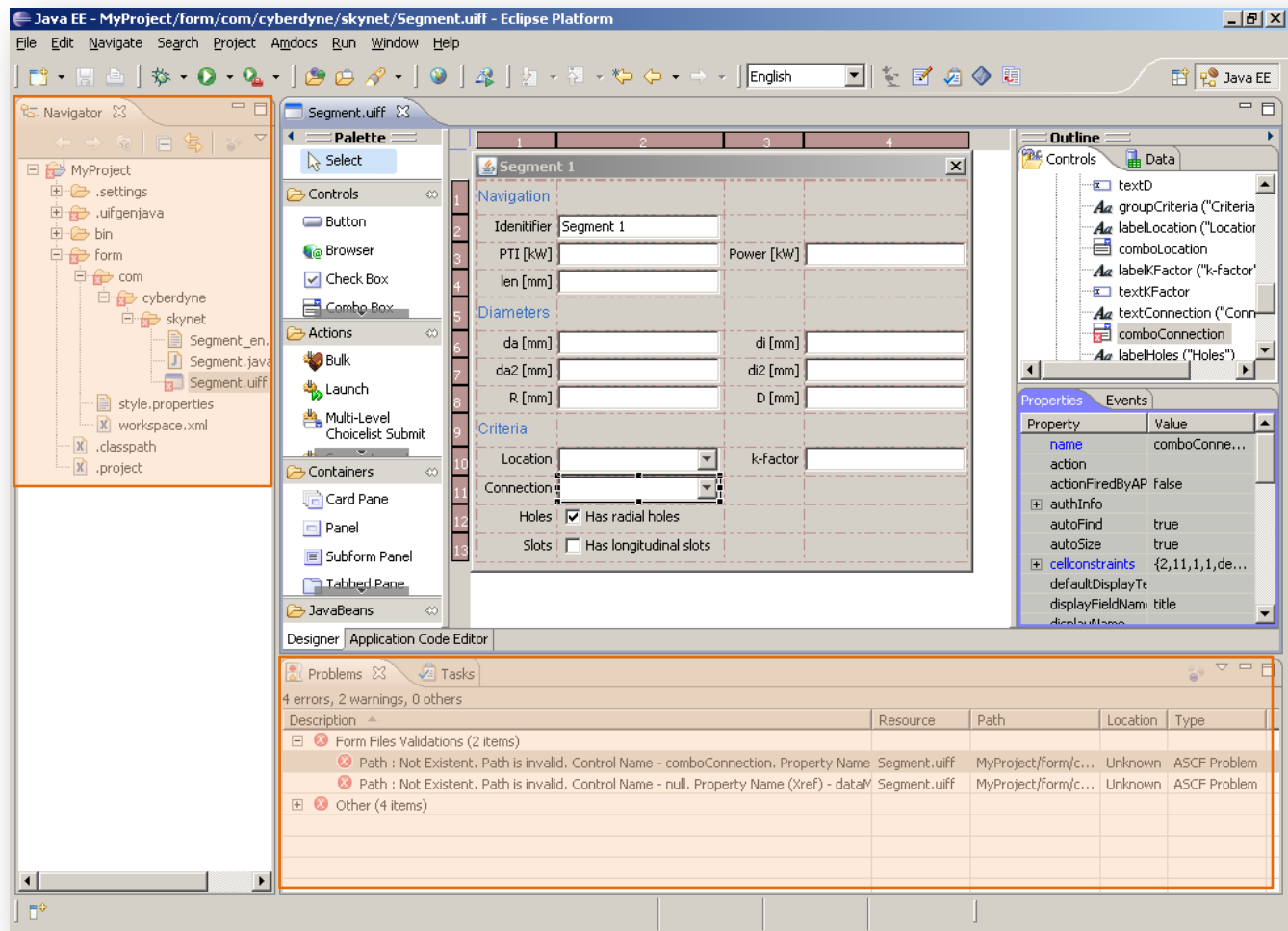
Immersed in Eclipse



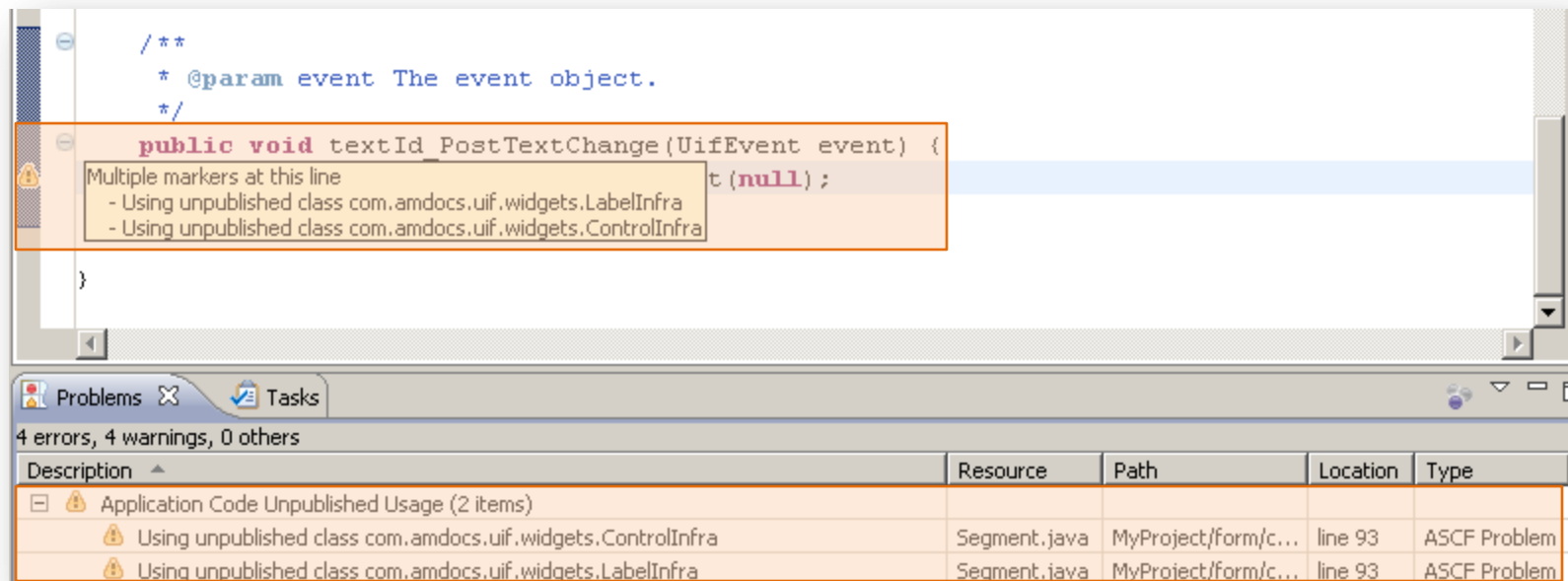
Immersed in Eclipse



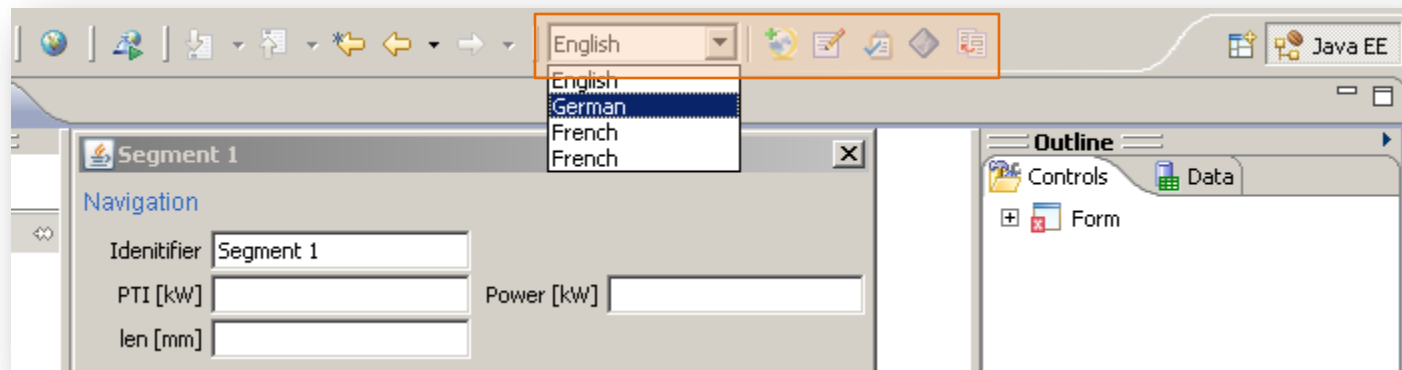
Immersed in Eclipse



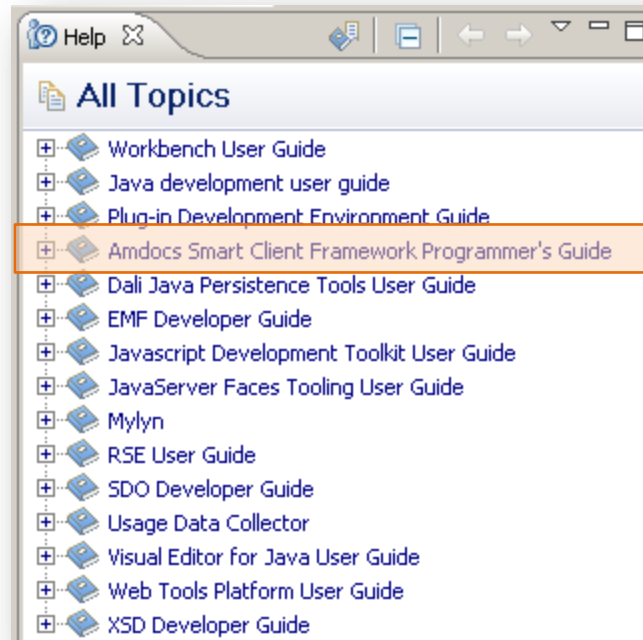
Immersed in Eclipse



Immersed in Eclipse



Immersed in Eclipse



For your users

Provides familiar
experience

For your developers

Less code to write,
test and support

Especially for
enterprise
solutions

Eclipse – rich
foundation

GMT

DTP

EMF

ECF

WTP

VE

GEF

RCP

JDT

AJDT

TPTP

JEM

RAP

WST

</acronyms>

Eclipse Projects									
Name & Project Information Page		Ph	Hm	Nw	Ma	Wi	IP		
Business Intelligence and Reporting Tools (BIRT)		✓	●	📖	📖	●	?		
Data Tools Platform		✓	●	📖	📖	●	?		
Connectivity		✓	●	📖	📖	●	?		
DTP Incubator		🥚	●	📖	📖	●	?		
Enablement		✓	●	📖	📖	●	?		
Model Base		✓	●	📖	📖	●	?		
SQL Dev Tools		✓	●	📖	📖	●	?		
Device Software Development Platform		Eclipse Model Framework Technology (EMFT)		🥚	●	📖	📖	●	?
Device Debugging		Model-to-Model Transformation (M2M)		🥚	●	📖	📖	●	?
ERCP - Embedded Rich Client Platform		Model To Text (M2T)		🥚	●	📖	📖	●	?
Mobile Tools for Java		Eclipse Modeling Framework (EMF)		✓	●	📖	📖	●	?
Native Application Builder		Textual Modeling Framework		🥚	●	📖	📖	●	?
Real-Time Software Components		Eclipse RT		✓	●	📖	📖	●	?
Target Management		Eclipse Communication Framework		✓	●	📖	📖	●	?
Tools for mobile Linux		Equinox		g-Eclipse Technology Project					
Eclipse Project		embedded Rich Client Platform		GEF3D					
e4 Project		Eclipse Persistence Services Project		Glimmer					
JDT - Java development tools		Rich Ajax Platform		Higgins					
PDE - Plugin Development Environment		Riena Platform Project		The IDE Meta-tooling Platform					
Eclipse Platform		SMILA		Eclipse IAM (Integration for Apache Maven)					
Eclipse Project Incubator		Swordfish		Eclipse IAM (Integration for Apache Maven)					
Galileo Simultaneous Release		SOA Tools		Java Workflow Tooling					
Eclipse Modeling Project		BPMN modeler		Maven Integration					
Modeling Amalgamation Project		SCA Tools		Maynstall					
Generative Modeling Technologies (GMT)		Eclipse Technology Project		Memory Analyzer					
Graphical Modeling Framework		Accessibility Tools Framework		The Eclipse on Linux Project					
Model Development Tools (MDT)		Albireo		Open Financial Market Platform					
		Aperi Storage Management Project		Open Healthcare Framework					
		Babel		Open Requirements Management Framework					
		BPEL Designer Project		Open System Engineering Environment					
		CloudFree Commerce Platform		Eclipse Packaging Project					
		Community-driven Systems Management in Open Source Project		Photran					
		Dash, Tools for Committers and members		Eclipse Process Framework Project					
		Dynamic Languages Toolkit		Phoenix Project					
		ECM Rich Client Platform		The Eclipse Spaces Project					
		The Eclipse Examples Project		Subversive - SVN Team Provider					
		Faceted Project Framework		Summer of Code					
				Supplemental Widgets for SWT (Nebula)					
				SWTBot					
				Tigerstripe					
				Tool Services Framework					
				Voice Tools Project					
				Tools Project					
				AJDT - AspectJ Development Tools Project					
				AspectJ					
				Buckminster Component Assembly					
				C/C++ Development Tooling (CDT)					
				COBOL IDE					
				GEF - Graphical Editor Framework					
				Hibachi					
				Myllyn					
				Eclipse Orbit Project					
				Parallel Tools Platform (PTP)					
				PHP Development Tools					
				VE - Visual Editor					
				Test and Performance Tools Platform Project					
				Monitoring Tools					
				Testing Tools					
				TPTP Platform					
				Tracing & Profiling Tools					
				Eclipse Web Tools Platform Project					
				AJAX Tools Framework					
				WTP Common Tools					
				Dali Java Persistence Tools					
				Datatools					
				WTP EJB Tools					
				WTP Java EE Tools					
				JavaServer Faces					
				Webtools Releng					
				Server Tools					
				Source Editing					
				Web Services Tools					
				Webtools Website					
				WTP Incubator					

Community health - example

Committers (14)



Victor Rodriguez
COO Committer
Open Canarias S.L.
Tenerife, Spain



Nick Doran
Release Engineer
Jboss, a Division of Red Hat
Toronto, Ontario, Canada



Christian Dumas
EMF-QTV & MDT-OCJ Lead
Zeligsoft Inc.
Gatineau, Québec, Canada



Arnaud Chenevaz
COO Committer
Puzzle ITC GmbH
Antibes, France



Stéphane Gauthier
Tango committer
Geosys
Boulogne-Billancourt, Hauts-de-Seine, France



Karen Hursey
MDT PWC
Embarcadero Technologies
Ottawa, Ontario, Canada



Simon Macdonald
COO Committer
Oxibus Inc.
Ottawa, Ontario, Canada



Ed MacIsaac
Working PWC Co-lead
EMF/EMF-T PWC, XSD Lead
MacIsaac Consulting and partner of
Janssen AG
Bellefleur, Ontario, Canada



Marlene Paternoster
EMF Core Team Lead
IBM Rational Software Canada
Toronto, Ontario, Canada



Shuaib Sheikh
Tango, Oak, & EclipseJunk
Committer
Ottawa, Canada
Toronto, Ontario, Canada



David Strassberg
EMF Core Team
IBM Rational Software Canada
Toronto, Ontario, Canada



Elwa Steppert
COO & N/A Lead
ES-Computersysteme
Berlin, Germany



Martin Tuel
Tango Lead, COO Committer
Elver.org
The Netherlands



Stefan Wenzel
COO Committer
FernUniversität in Hagen
Hagen, Germany

Contributors (1)



Neil Skrypnich
Infrastructure Developer
University of Brock
St. Catharines, Ontario,
Canada

<http://www.eclipse.org/modeling/emf/project-info/team.php>

- Open Canarias
- RedHat
- Zeligsoft
- Puzzle ITC
- Geensys
- Embarcadero
- Okidoo
- IBM
- Oracle
- ES-Computersysteme
- Elver
- FernUniversitaet
- University of Brock

GMT

DTP

EMF

ECF

WTP

VE

GEF

RCP

JDT

AJDT

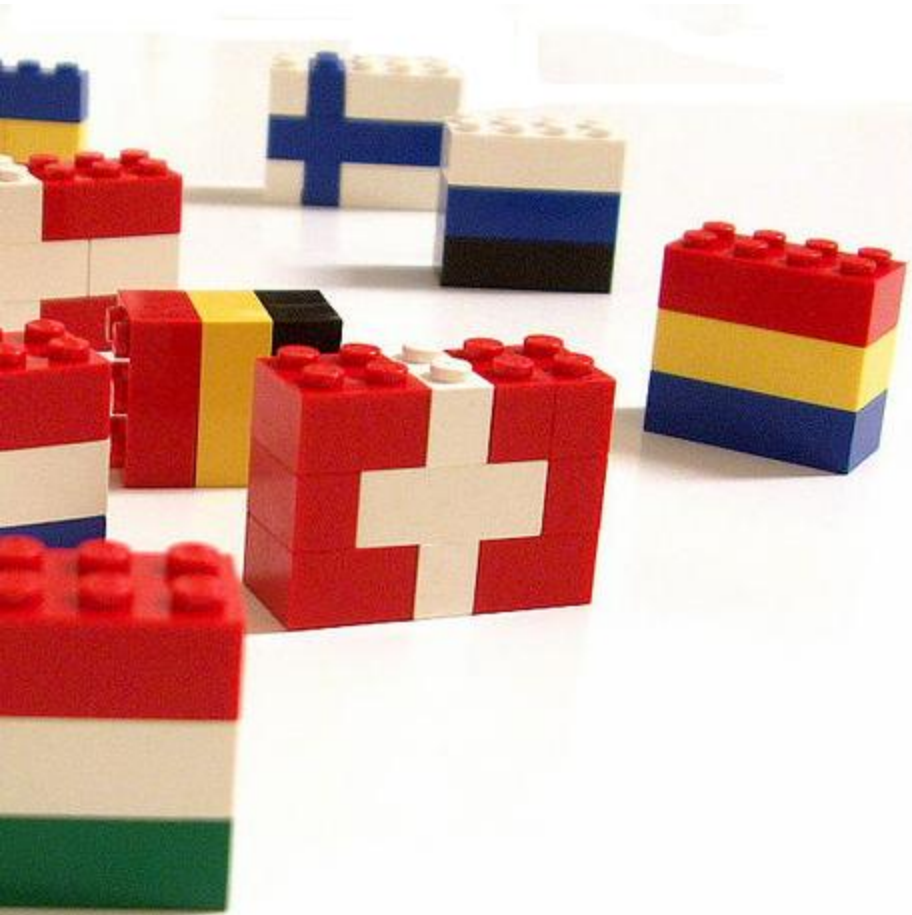
TPTP

JEM

RAP

WST

Our building blocks



- EMF – persistence
- JEM / VE – UI builder
- GEF – editors
- JDT – Java source trees

Putting the pixels on canvas

XML



Java model



Swing frame



Canvas

The screenshot shows a Swing window titled "Segment 1" with a close button in the top right corner. The window contains a form with a table-like structure with 13 rows and 4 columns. The columns are labeled 1, 2, 3, and 4 at the top. The rows are numbered 1 to 13 on the left. The form is organized into sections: "Navigation" (rows 1-4), "Diameters" (rows 5-8), and "Criteria" (rows 9-13). The "Navigation" section includes an "Identifier" field with the value "Segment 1", a "PTI [kW]" field, a "Power [kW]" field, and a "len [mm]" field. The "Diameters" section includes "da [mm]", "di [mm]", "da2 [mm]", "di2 [mm]", "R [mm]", and "D [mm]" fields. The "Criteria" section includes a "Location" dropdown, a "k-factor" field, a "Connection" dropdown, a "Holes" checkbox with the label "Has radial holes" (checked), and a "Slots" checkbox with the label "Has longitudinal sl..." (unchecked).

	1	2	3	4
1	Segment 1			
2	Navigation	Identifier	Segment 1	
3		PTI [kW]		Power [kW]
4		len [mm]		
5	Diameters			
6		da [mm]		di [mm]
7		da2 [mm]		di2 [mm]
8		R [mm]		D [mm]
9	Criteria			
10		Location		k-factor
11		Connection		
12		Holes	<input checked="" type="checkbox"/> Has radial holes	
13		Slots	<input type="checkbox"/> Has longitudinal sl...	

EMF as the persistence layer

The image shows a software development environment with two main panels. The left panel is a form editor titled "Segment 1" with a grid layout. The right panel shows the "Outline" and "Properties" windows.

Form Editor (Segment 1):

	1	2	3	4
1	Navigation			
2	Identifier	Segment 1		
3	PTI [kW]		Power [kW]	
4	len [mm]			
5	Diameters			
6	da [mm]		di [mm]	
7	da2 [mm]		di2 [mm]	
8	R [mm]		D [mm]	
9	Criteria			
10	Location		k-factor	
11	Connection			
12	Holes	<input checked="" type="checkbox"/> Has radial holes		
13	Slots	<input type="checkbox"/> Has longitudinal slots		

Outline:

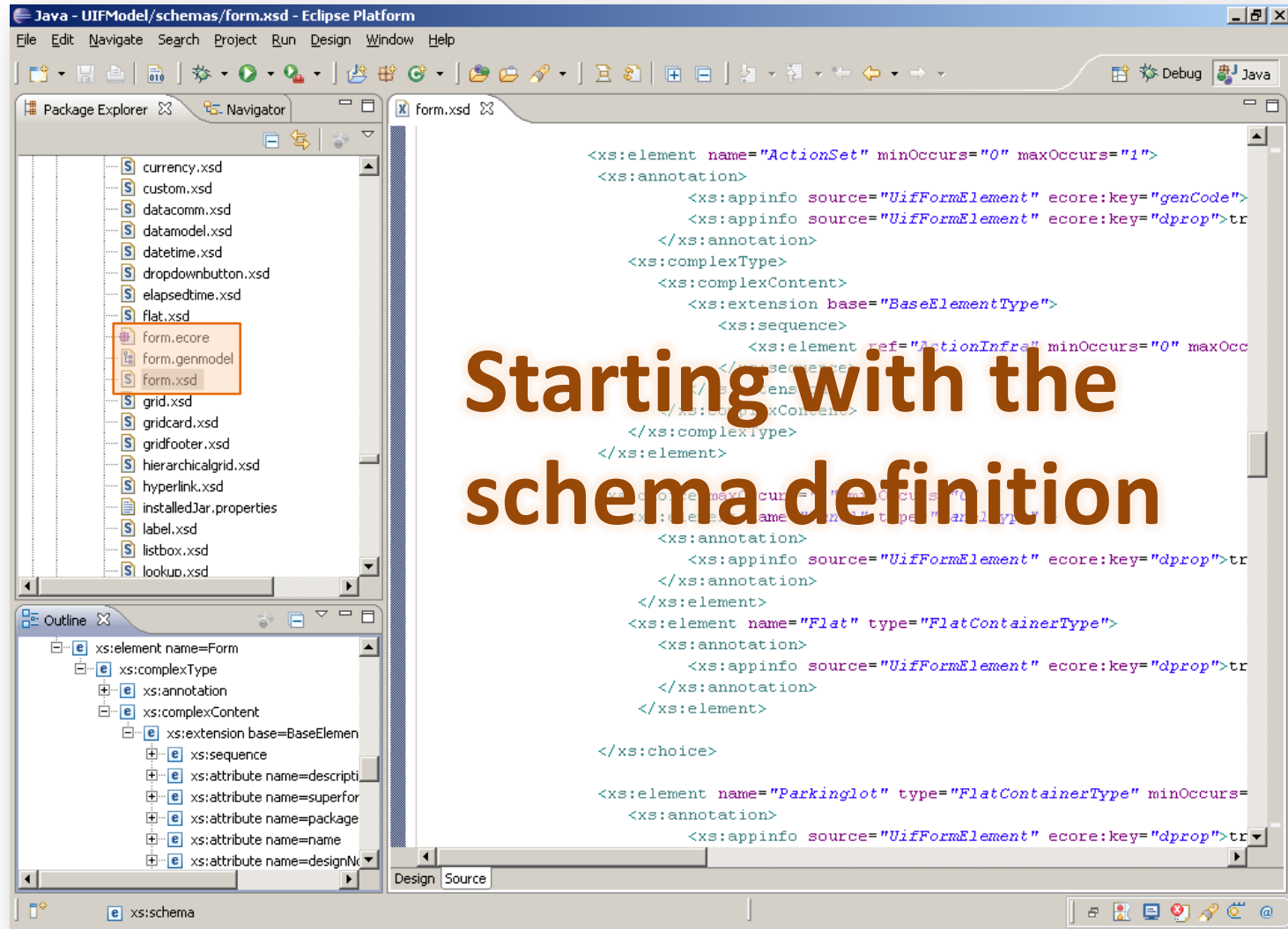
- Form
 - root
 - groupNavigation ("Nav")
 - labelId ("Identifier")
 - labelPti ("PTI [kW]")
 - textId
 - textPti
 - labelLen ("len [mm]")
 - textLen

Properties:

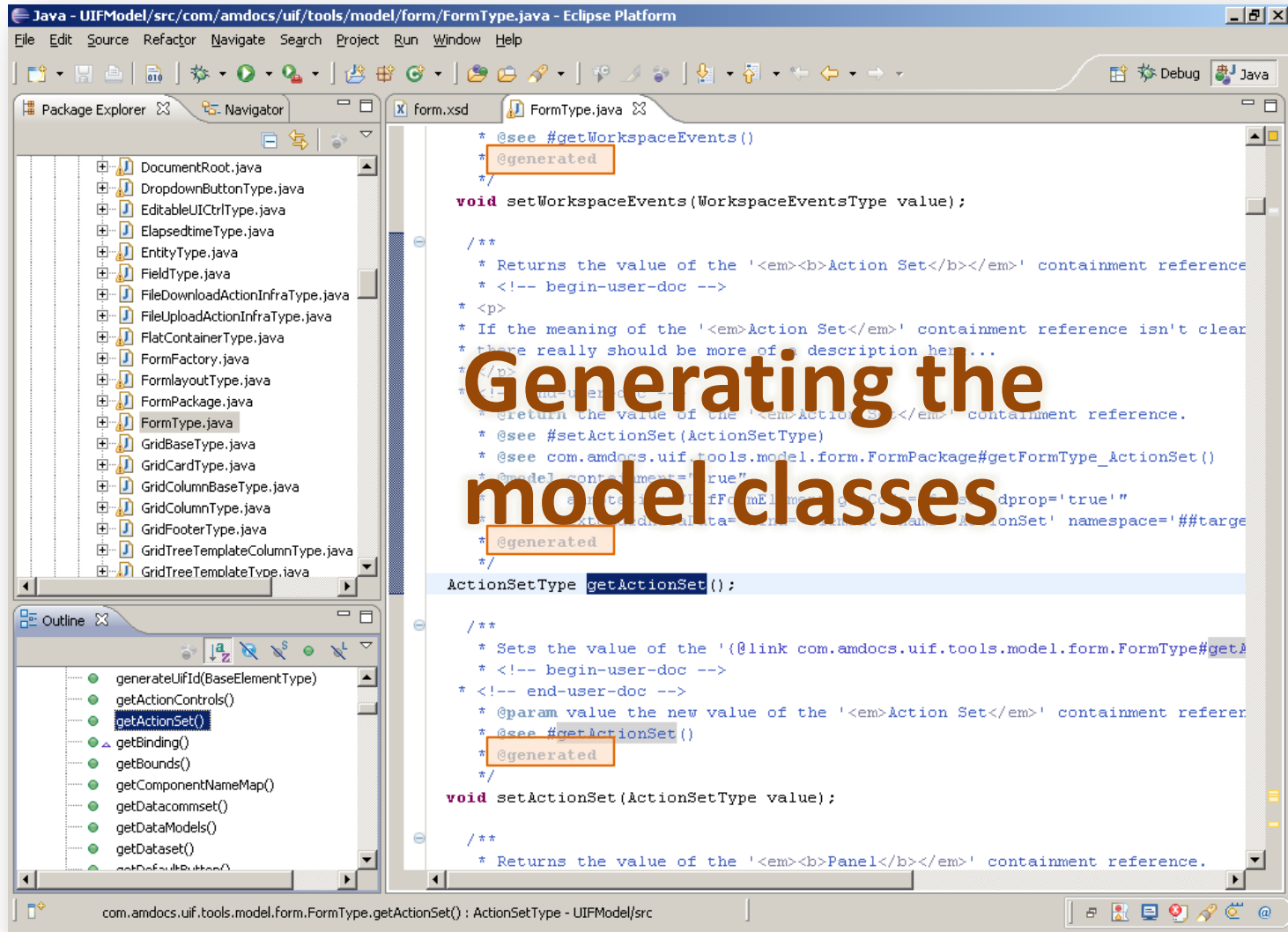
Property	Value
name	root
title	
authInfo	
displayName	
enabled	true
modificationTrac	true
preferredColumnr	-1
preferredRows	-1
required	false
scrollable	false



EMF as the persistence layer



EMF as the persistence layer



EMF as the persistence layer

Segment.uiff

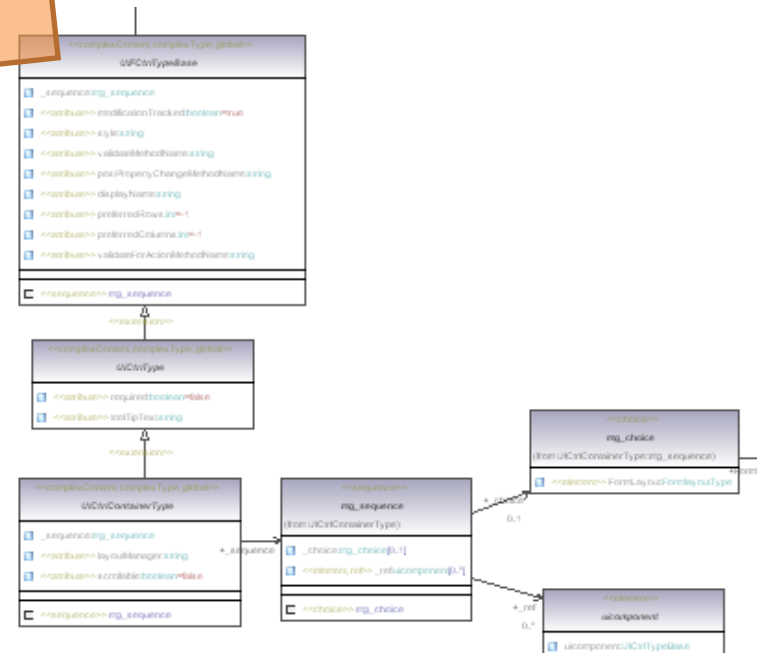
```
<?xml version="1.0" encoding="UTF-8"?>
<Form xmlns="http://" uifid="Form" bounds="0,
  <WorkspaceEvents uifid="uifId27123309482240
  <ActionSet uifid="uifId11233094090535"/>
  <Panel uifid="uifId21233094090535" name="ro
    <AuthInfo uifid="uifId81233094170908"/>
  <FormLayout uifid="uifId31233094090535" c
```

```
Resource formModelResource = null;
ResourceSet resourceSet = new ResourceSetImpl();
resourceSet.getResourceFactoryRegistry().getExtensionToFactoryMap()
    .put(Resource.Factory.Registry.DEFAULT_EXTENSION,
        new FormResourceFactoryImpl());

resourceSet.getPackageRegistry().put(FormPackage.eNS_URI,
    FormPackage.eINSTANCE);

try {
    URI inputURI = URI.createPlatformResourceURI(modelFile
        .getFullPath().toString(), false);
    formModelResource = resourceSet.createResource(inputURI);
    formModelResource.load(getLoadOptions());
} catch (Exception e) {
    e.printStackTrace();
}
return formModelResource;
```

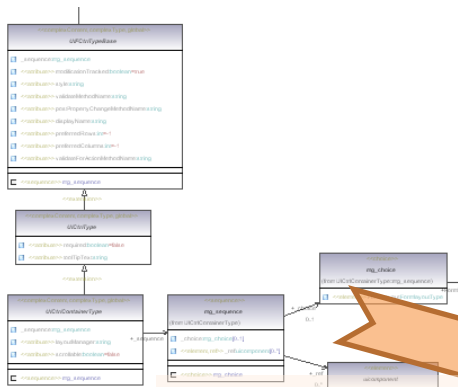
xml -> java model



xml \leftrightarrow java
notifications
validations

JEM proxies

Java model



Offscreen Swing JFrame

Creation and updates
via JEM proxies

Segment 1

Navigation

Identifier: Segment 1

PTI [kW]:

Power [kW]:

da [mm]:

di [mm]:

da2 [mm]:

di2 [mm]:

R [mm]:

D [mm]:

Criteria

Location: [dropdown]

Connection: [dropdown]

Holes: ☒ Has radial holes

Slots: ☐ Has longitudinal sl...

GEF graphical edit parts

image changes

Segment 1

Navigation

Identifier: Segment 1

PTI [kW]:

Power [kW]:

da [mm]:

di [mm]:

da2 [mm]:

di2 [mm]:

R [mm]:

D [mm]:

Criteria

Location: [dropdown]

Connection: [dropdown]

Holes: ☒ Has radial holes

Slots: ☐ Has longitudinal sl...

GEF - everything is an *EditPart*

The screenshot displays the Eclipse IDE interface with a GEF editor and two side panels. The editor, titled 'Segment 1', is divided into four columns and contains sections for Navigation, Diameters, and Criteria. The Navigation section includes fields for Identifier, PTI [kW], and len [mm]. The Diameters section includes fields for da [mm], da2 [mm], R [mm], and D [mm]. The Criteria section includes dropdowns for Location and Connection, and checkboxes for 'Has radial holes' and 'Has longitudinal slots'. The Outline view on the right shows a tree structure for the 'Form' control, with nodes for 'groupNavigation', 'labelPti', 'textId', 'textPti', 'labelLen', and 'textLen'. The Properties view at the bottom right shows a table of properties for the selected 'root' object.

**org.eclipse.gef.ui.parts
TreeViewer**

**org.eclipse.gef.editparts
AbstractGraphicalEditPart**

Property	Value
name	root
title	
authInfo	
displayName	
enabled	true
modificationTrac	true
preferredColumnr	-1
preferredRows	-1
required	false
scrollable	false

All together now

XML



Java model



Swing frame



Canvas

The screenshot shows a window titled "Segment 1" with a close button. The window is divided into several sections with a vertical index on the left (1-13). The sections are:

- Navigation** (row 1): Identifier: Segment 1 (row 2)
- Parameters** (rows 3-4): PTI [kW] (row 3), Power [kW] (row 3), len [mm] (row 4)
- Diameters** (row 5): da [mm] (row 6), di [mm] (row 6), da2 [mm] (row 7), di2 [mm] (row 7), R [mm] (row 8), D [mm] (row 8)
- Criteria** (row 9): Location (dropdown, row 10), k-factor (row 10), Connection (dropdown, row 11)
- Holes** (row 12): ☒ Has radial holes
- Slots** (row 13): ☐ Has longitudinal sl...

Java-based warnings

Java class change



Builder invocation



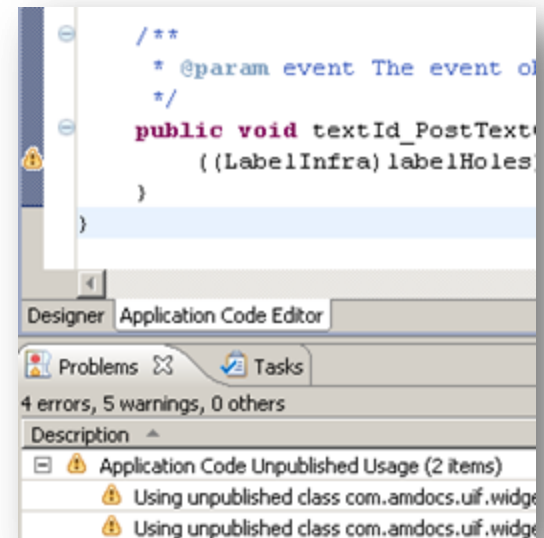
Java syntax trees



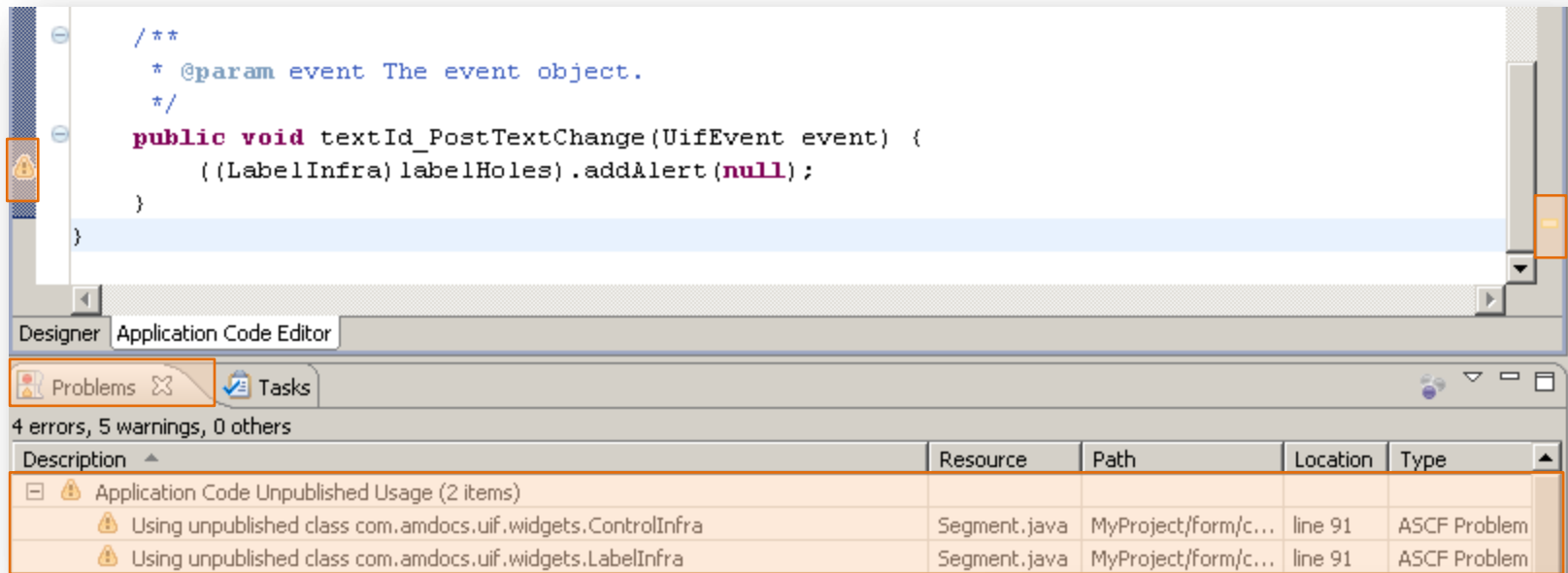
Marker creation



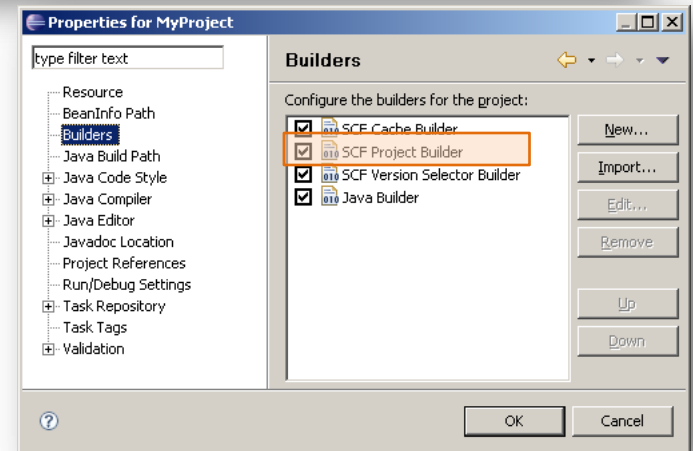
UI update



Flagging API usage violations



```
org.eclipse.jdt.core.dom.ASTParser  
  
public void createASTs(ICompilationUnit[] compilationUnits,  
    String[] bindingKeys,  
    ASTRequestor requestor,  
    IProgressMonitor monitor)
```



All together now

Java class change



Builder invocation



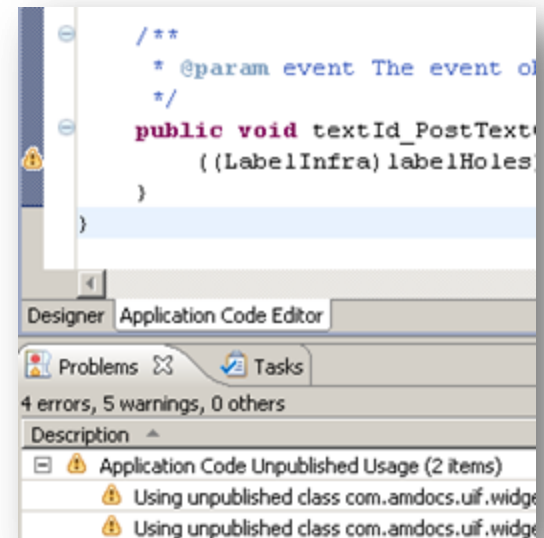
Java syntax trees



Marker creation



UI update



And those are just
the big pieces

markers

builders

actions

monitors

natures

commands

wizards

help

dialogs

validations

views

jobs

editors

perspectives

markers

builders

actions

monitors

natures

commands

wizards

help

dialogs

validations

views

jobs

editors

perspectives

Asynchronous

Background

Queuing

Cancellable

Progress

big pieces
+
small pieces

=

the Eclipse way

The **only** rule of
Eclipse way
don't fight it

==

provide familiar
experience to
your users

Our experience

- Reuse as much plumbing as possible
- Reuse as many UI parts as possible
- Use very few synchronous UI operations
- Reuse as many UI flows as possible
- Decompose into plugins
- Build for extensibility
- Don't depend on internal code

</dudeTalking>

Q & A

<http://www.eclipse.org/documentation/>

kirillg@amdocs.com

www.pushing-pixels.org

Thank you

Image credit: **cemre** at <http://flickr.com/photos/f/637025/> under NC-SA 2.0