EMF Parsley

From EMF to UIs: how to use EMF Parsley to get desktop, web and mobile UIs from the model

Vincenzo Caselli
Lorenzo Bettini

EclipseCon Europe 2015
Ludwigsburg - Nov 4, 2015

About us

Lorenzo Bettini
Dip. Informatica, Univ. Torino, Italy
bettini@dsi.unifi.it
@lorenzo_bettini
www.lorenzobettini.it

Vincenzo Caselli
RCP Vision
vincenzo.caselli@rcp-vision.com
@vcaselli
www.rcp-vision.com

Francesco Guidieri
RCP Vision
francesco.guidieri@rcp-vision.com
@fraguid
www.rcp-vision.com

Talk Goals

- A practical use case of Parsley usage: how to create and customize a Gmail webapp client with RAP and Parsley in minutes
  - ✓ RAP (Remote Application Platform) allows porting RCP applications on the web
- New Parsley feature for other non-desktop environments (web and mobile)

Intro a EMF Parsley

The goal was to provide an easy way to get a User Interface from an EMF model.

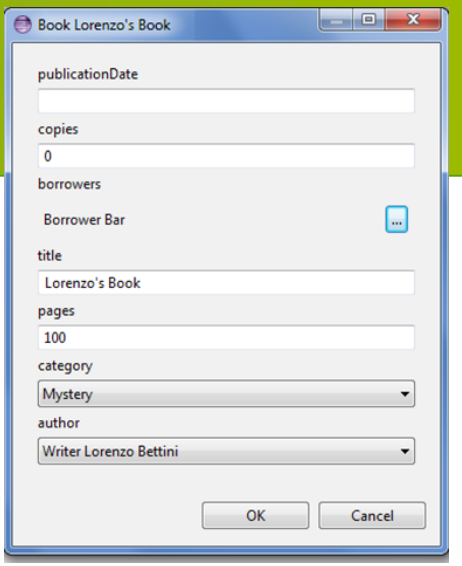
EMF Parsley provides:

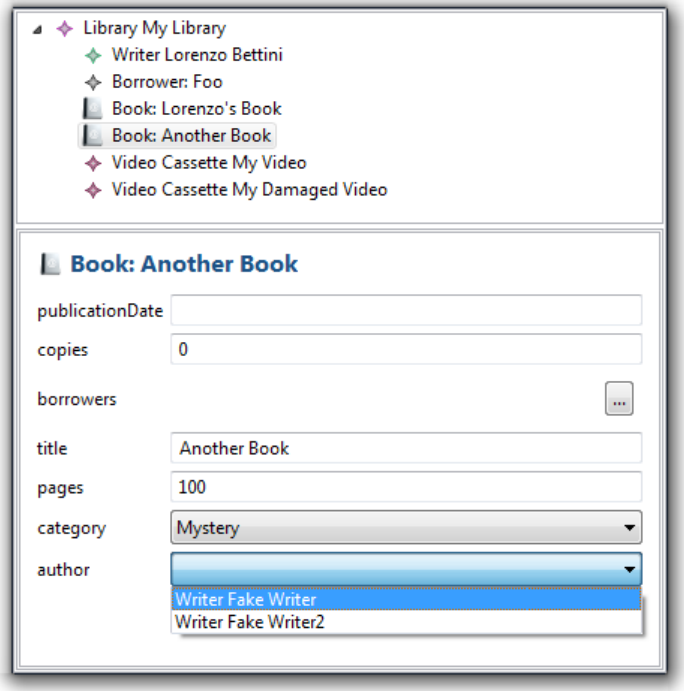
- simple UI components that can be used in existing applications
- easy-to-use customization mechanisms via Dependency Injection
- DSL to customize UI aspects

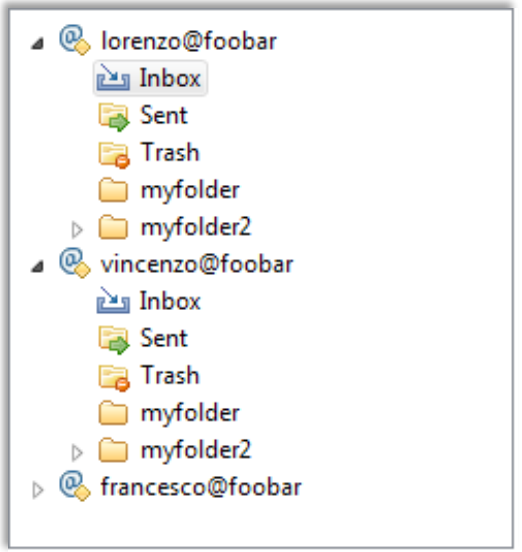
Parsley Components

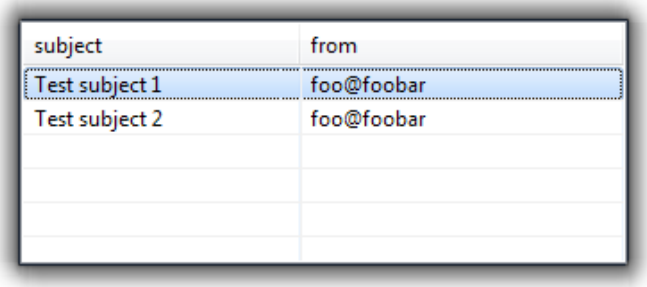
Basic components can be used out of the box:

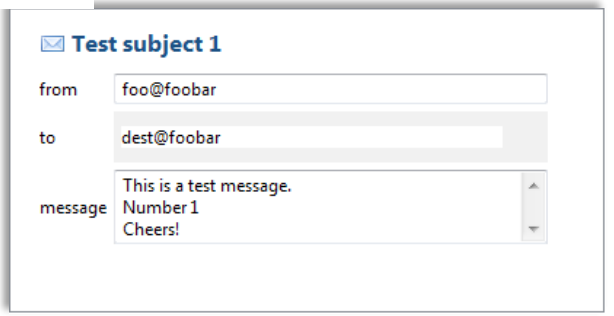
- Tree
- Table
- Form
- Tree Form
- Table Form
- Dialog











@EmfParsely

Reference implementations

- Views can also be considered as reference implementation of Parsley components usage.
  - Resource based
  - Selection based
- Project Wizards are provided with initial templates

How it works

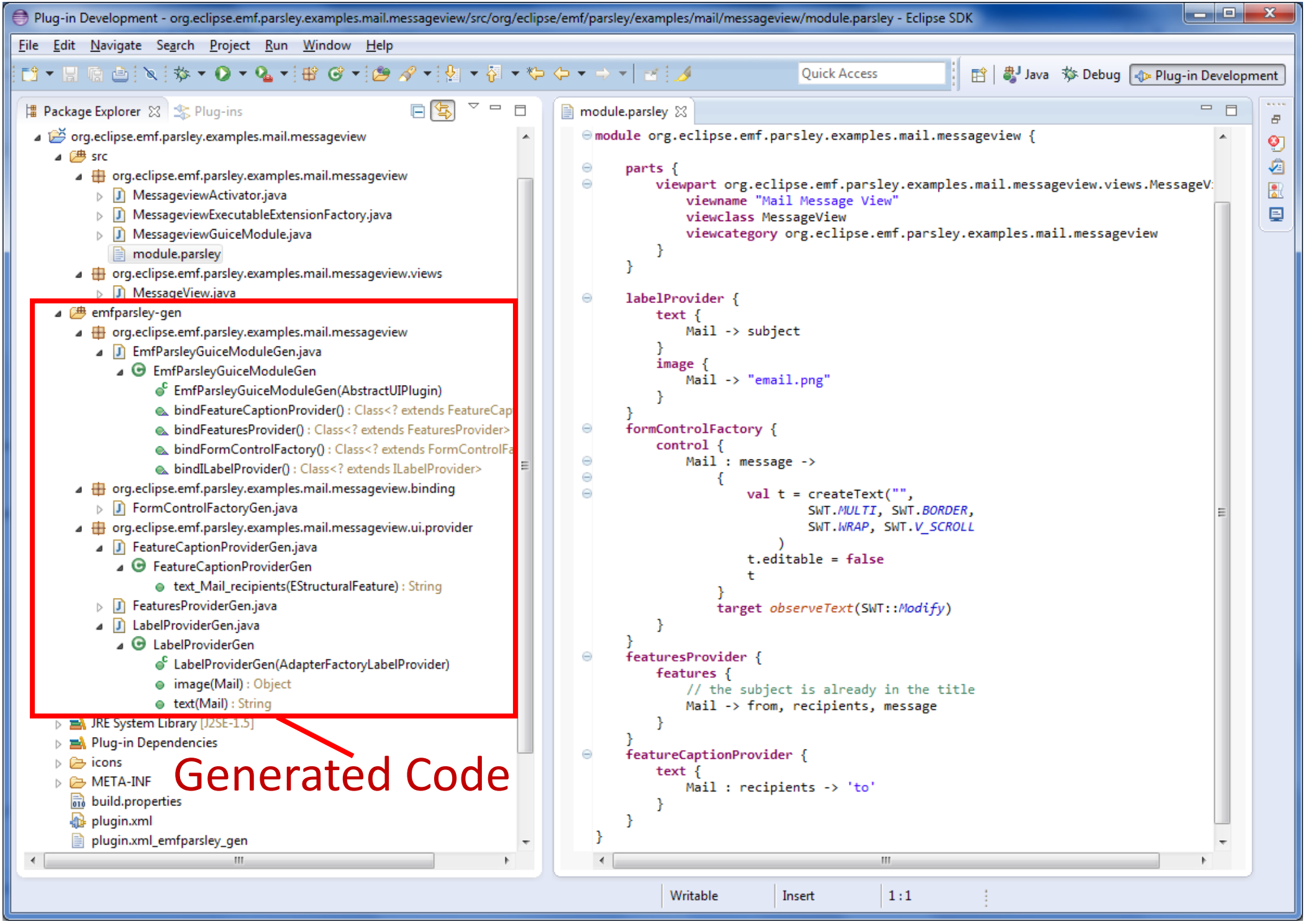
- Parsley UI is built on top of EMF.Edit reflective framework,
  - so the default behavior is delegated to EMF Edit
- Customizations are based on Dependency Injection:
  - no need to customize parts and composites
  - customize single behaviors
  - inject them in the framework

EMF Parsley DSL

- Implemented in Xtext, using Xbase
  - Interoperable with the Java type system
  - IDE tooling (including Debugging)
- Specify customizations in one single file in a compact form
  - Generates the corresponding Java code
  - Generates the Guice bindings
- You can use the DSL and manually written Java code together

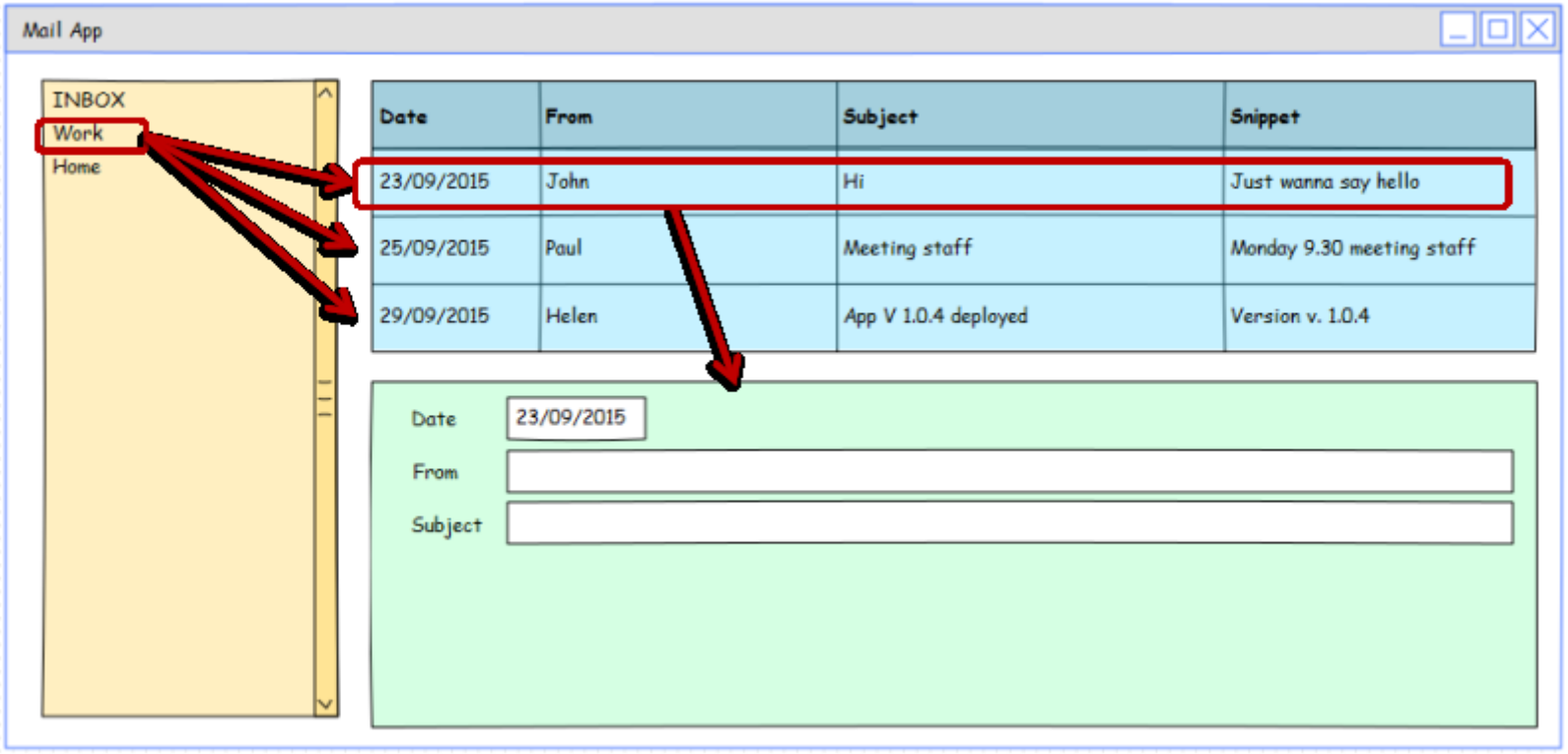
module.parsley

```
import org.eclipse.emf.parsley.examples.mail.Account
import org.eclipse.emf.parsley.examples.mail.Folder
import org.eclipse.emf.parsley.examples.mail.accountsview.views.AccountsView

/* org.eclipse.emf.parsley.examples.mail.accountsview Emf Parsley Dsl Module file */
module org.eclipse.emf.parsley.examples.mail.accountsview {

    parts {
        viewpart org.eclipse.emf.parsley.examples.mail.accountsview.views.AccountsView {
            viewname "Mail Accounts View"
            viewclass AccountsView
            viewcategory org.eclipse.emf.parsley.examples.mail.accountsview
        }
    }

    labelProvider {
        image {
            Account -> "account.gif"
            Folder -> {
                switch (name) {
                    case "Inbox" : "inbox.gif"
                    case "Sent" : "sent.gif"
                    case "Trash" : "trash.gif"
                    default: "folder.gif"
                }
            }
        }

        text {
            Account -> email
            Folder -> name
        }
    }

    viewerContentProvider {
        children {
            Folder -> subfolders // don't show emails
        }
    }
}
```


Plug-in Development - org.eclipse.emf.parsley.examples.mail.messageview/src/org/eclipse/emf/parsley/examples/mail/messageview/module.parsley - Eclipse SDK
File Edit Navigate Search Project Run Window Help
Quick Access
Java
Debug
Plug-in Development
Package Explorer
Plug-ins
org.eclipse.emf.parsley.examples.mail.messageview
src
org.eclipse.emf.parsley.examples.mail.messageview
MessageviewActivator.java
MessageviewExecutableExtensionFactory.java
MessageviewGuiceModule.java
module.parsley
org.eclipse.emf.parsley.examples.mail.messageview.views
MessageView.java
emfparsley-gen
org.eclipse.emf.parsley.examples.mail.messageview
EmfParsleyGuiceModuleGen.java
EmfParsleyGuiceModuleGen
EmfParsleyGuiceModuleGen(AbstractUIPlugin)
bindFeatureCaptionProvider() : Class<? extends FeatureCap
bindFeaturesProvider() : Class<? extends FeaturesProvider>
bindFormControlFactory() : Class<? extends FormControlFa
bindILabelProvider() : Class<? extends ILabelProvider>
org.eclipse.emf.parsley.examples.mail.messageview.binding
FormControlFactoryGen.java
org.eclipse.emf.parsley.examples.mail.messageview.ui.provider
FeatureCaptionProviderGen.java
FeatureCaptionProviderGen
text_Mail_recipients(EStructuralFeature) : String
FeaturesProviderGen.java
LabelProviderGen.java
LabelProviderGen
LabelProviderGen(AdapterFactoryLabelProvider)
image(Mail) : Object
text(Mail) : String
JRE System Library [J2SE-1.5]
Plug-in Dependencies
icons
META-INF
build.properties
plugin.xml
plugin.xml_emfparsley_gen
Generated Code
module.parsley
module org.eclipse.emf.parsley.examples.mail.messageview {
parts {
viewpart org.eclipse.emf.parsley.examples.mail.messageview.views.MessageV
viewname "Mail Message View"
viewclass MessageView
viewcategory org.eclipse.emf.parsley.examples.mail.messageview
}
}
labelProvider {
text {
Mail -> subject
}
image {
Mail -> "email.png"
}
}
formControlFactory {
control {
Mail : message ->
{
val t = createText("",
SWT.MULTI, SWT.BORDER,
SWT.WRAP, SWT.V_SCROLL
)
t.editable = false
t
}
target observeText(SWT::Modify)
}
}
featuresProvider {
features {
// the subject is already in the title
Mail -> from, recipients, message
}
}
featureCaptionProvider {
text {
Mail : recipients -> 'to'
}
}
}
Writable
Insert
1 : 1

Demo

A simple Gmail webapp client with EMF Parsley and RAP



Beyond RCP Concepts

- Ok, Parsley can be used in Web Applications with RAP
  (it is already in production)
- We wanted to go beyond the OSGi/RCP/RAP concepts
- Would it be possible to run Parsley on a pure Java Enterprise Environment (JEE) ?

Chopping Parsley

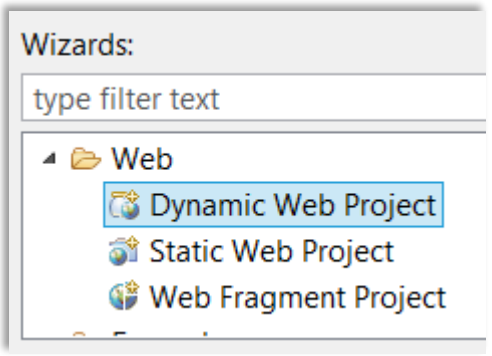
- Separated Parsley Core from (SWT) UI
- Surrounded Parsley Core with a JEE layer
- RCP Plugins dependencies are used as plain JARs in a pure Java environment
- Thanks to usage of Google Guice Injection Parsley Core keeps being fully operative

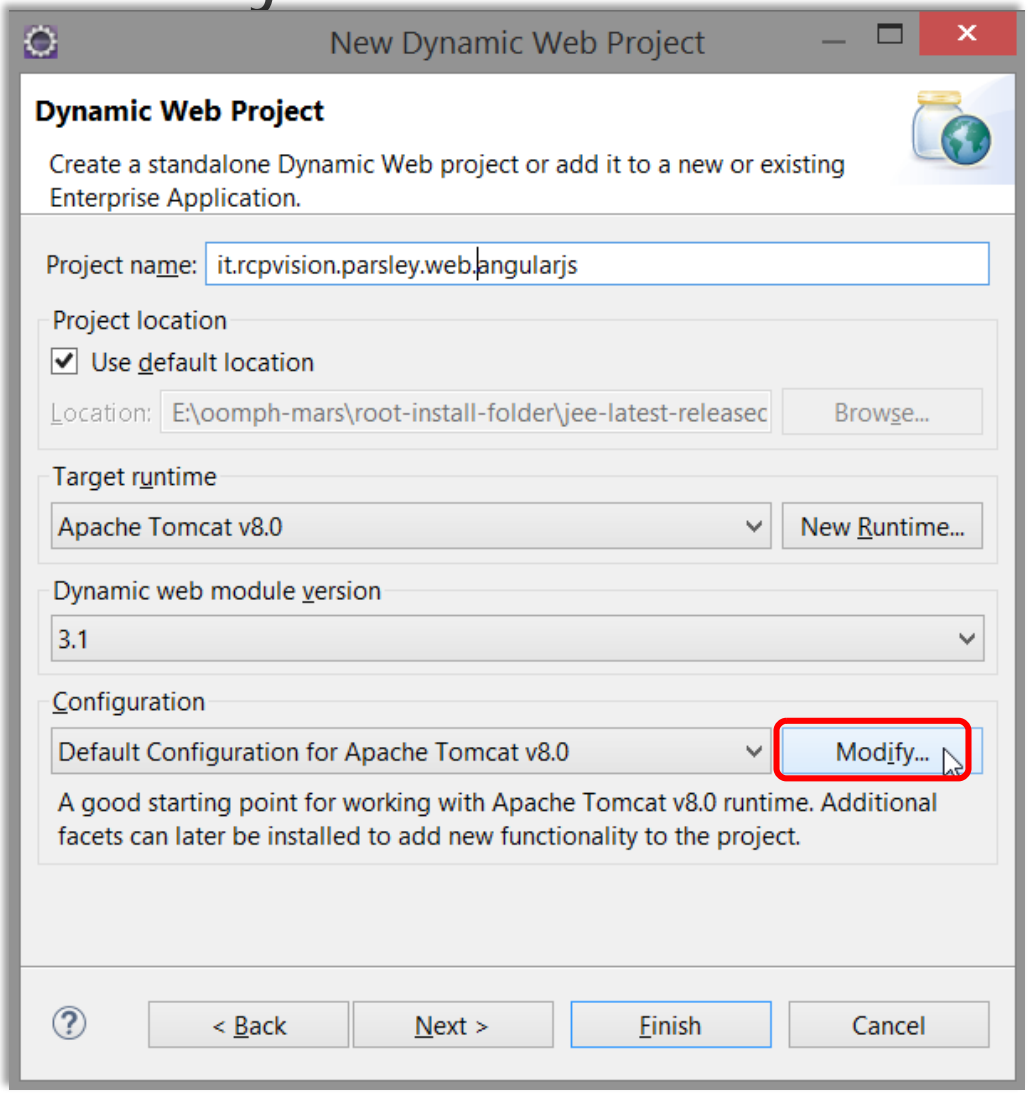
Dropping Parsley

- The JEE layer (servlets) access the Parsley Core and explores EMF Resources according with Parsley customizations
- JEE layer exposes Parsley via JSON APIs
- Also widgets customizations are available thanks to a lightweight headless porting of the SWT layer

Parsley JEE Wizard

Use classic Dynamic Web Project wizard
… but with the
new Parsley Facet

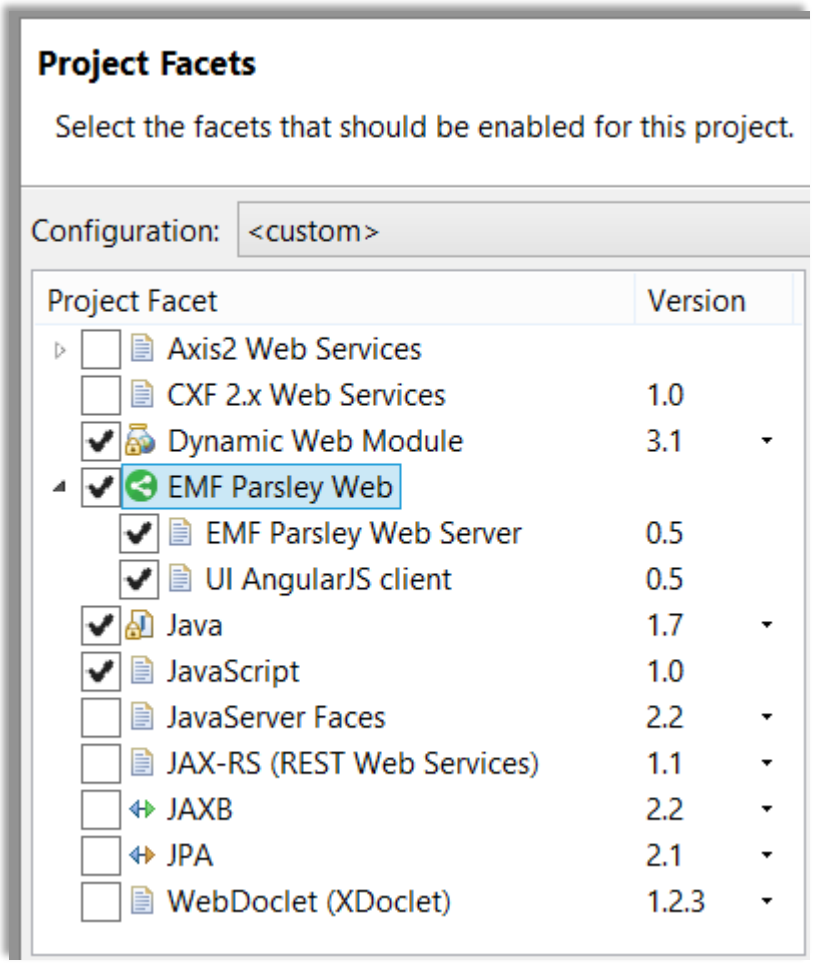




Parsley JEE Wizard

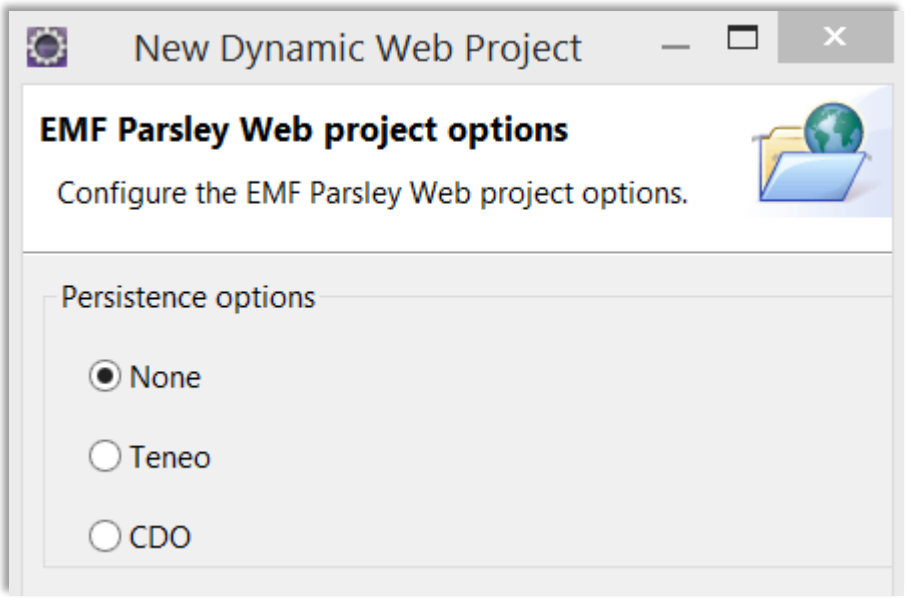
Two facets available:

- JSON API Parsley Server
- UI AngularJS

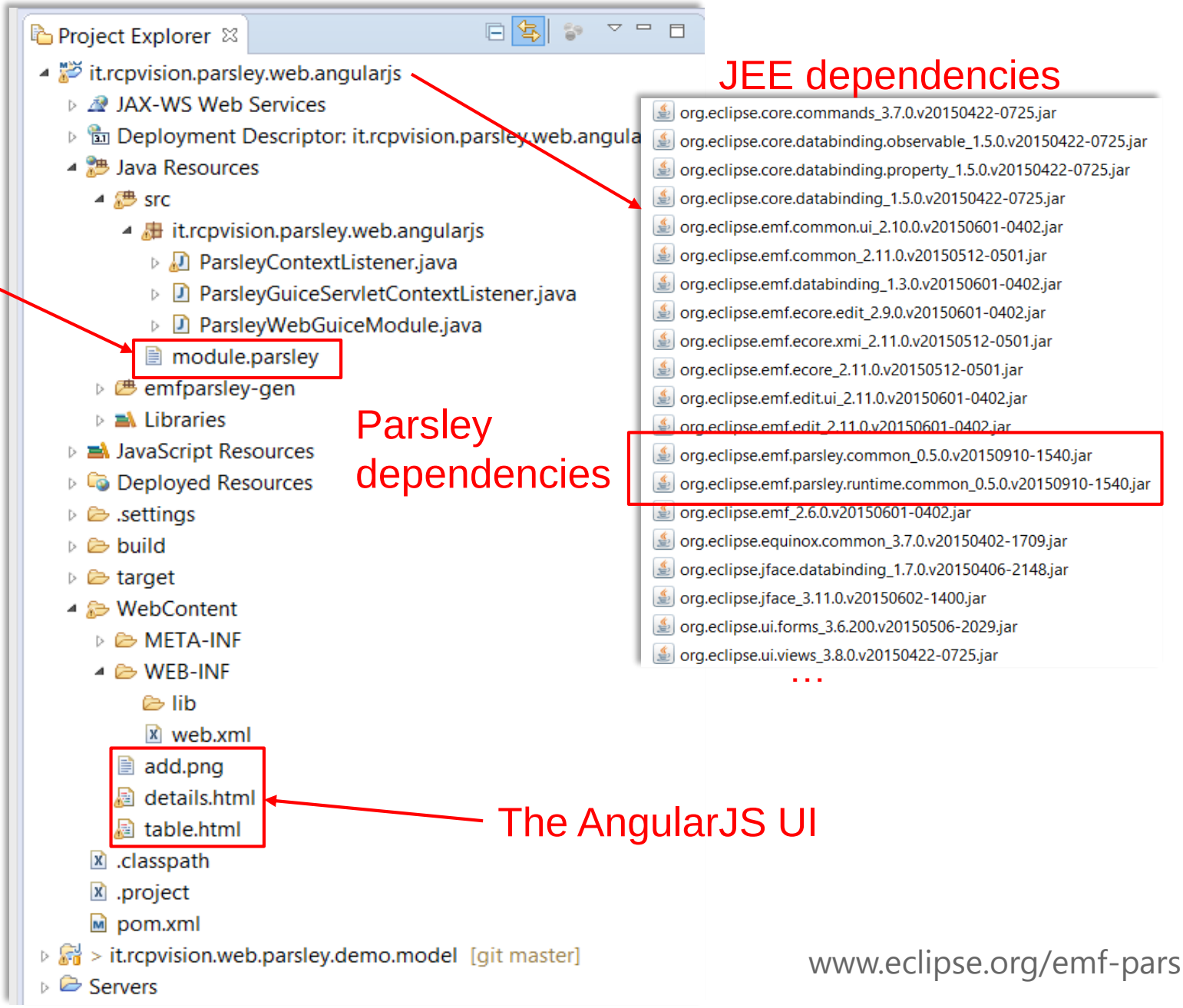


Parsley JEE Wizard

Persistence choice



The generated JEE project

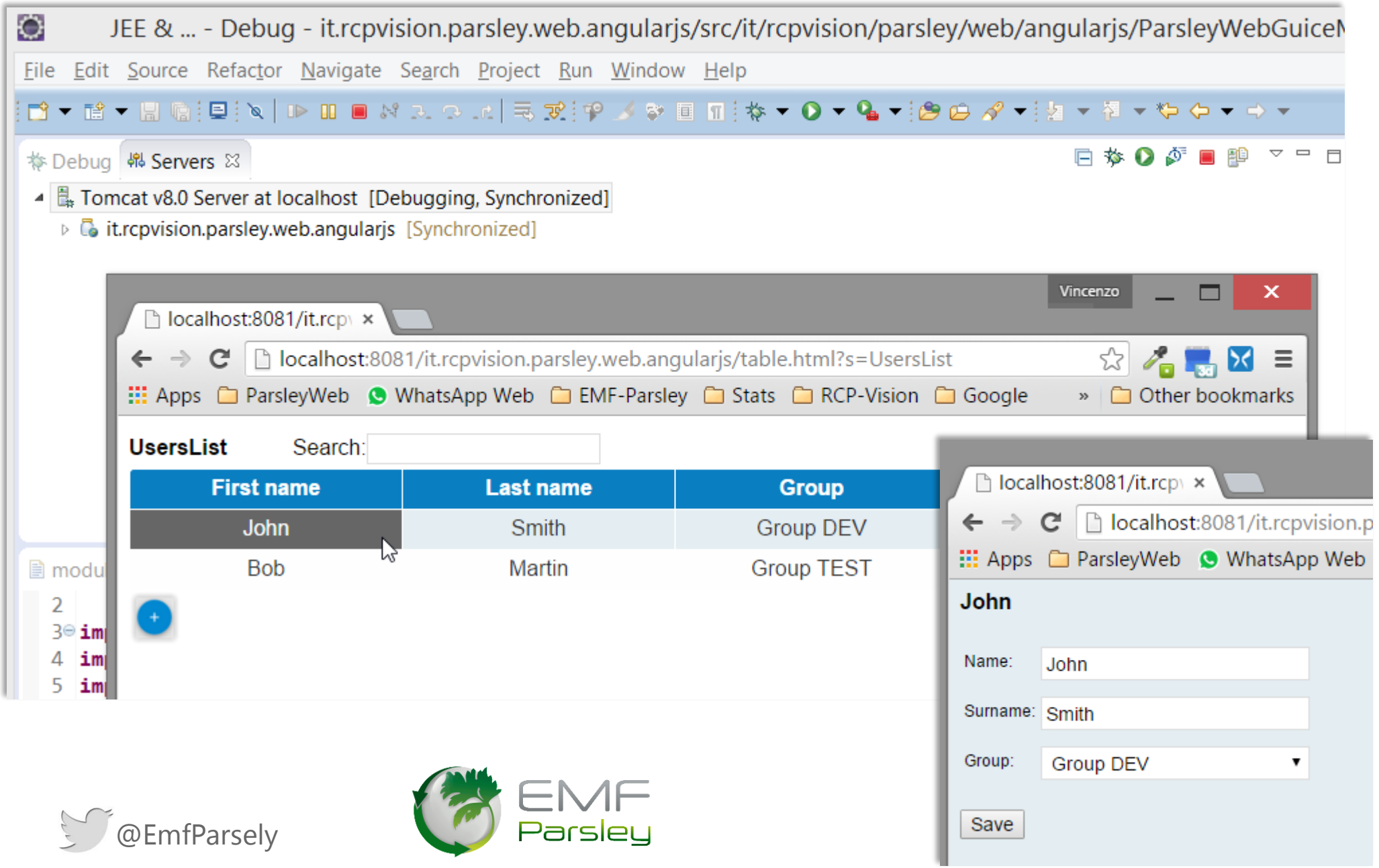


Parsley flavours

- JSON Parsley APIs allows building UIs with any technology

- Some initial implementations are available:
  - ✓ AngularJS and GWT for the web
  - ✓ Eclipse Andmore for mobile Android
  - ✓ Eclipse Thym for hybrid mobile

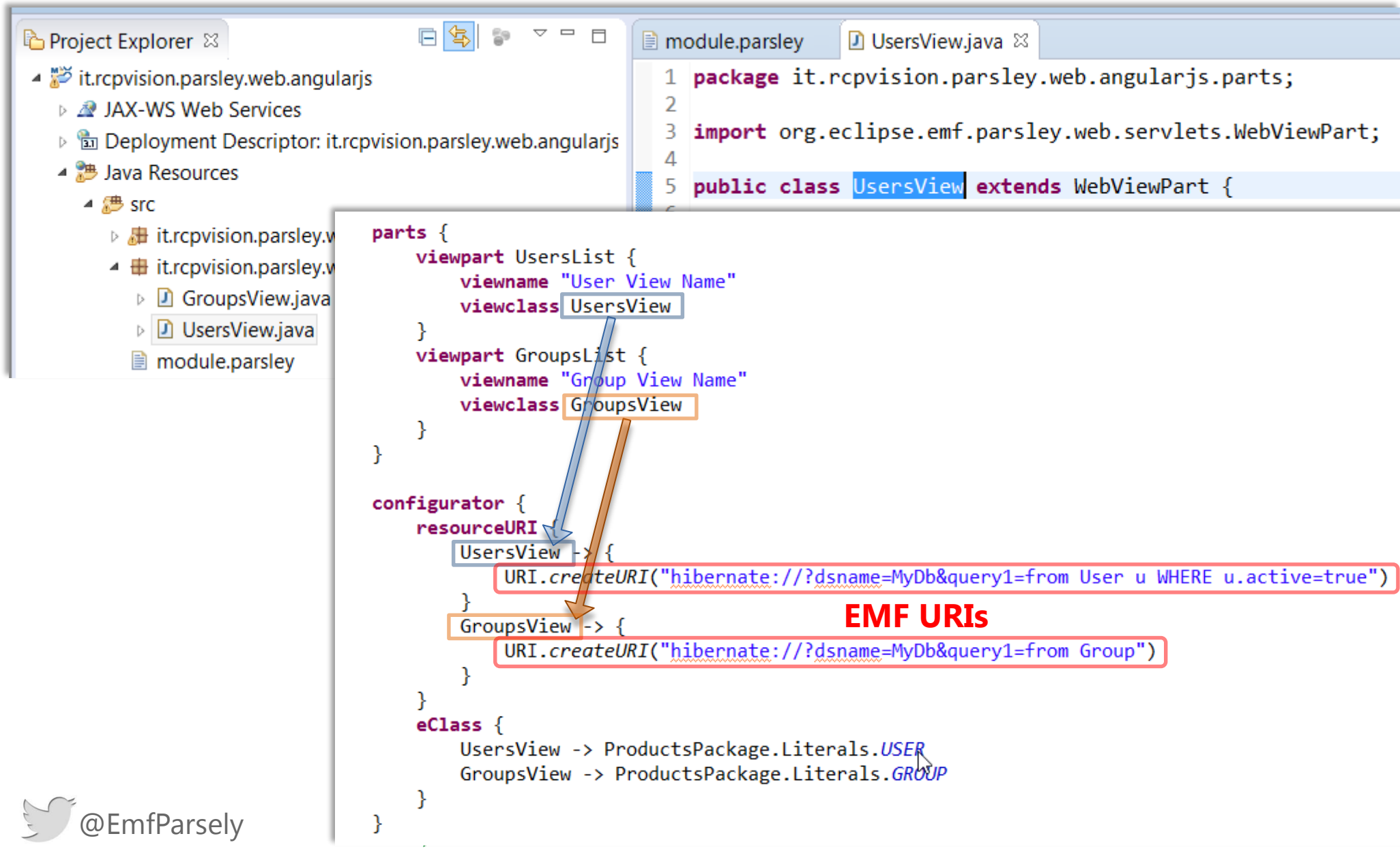
Parsley on AngularJS



@EmfParsely

EMF Parsley

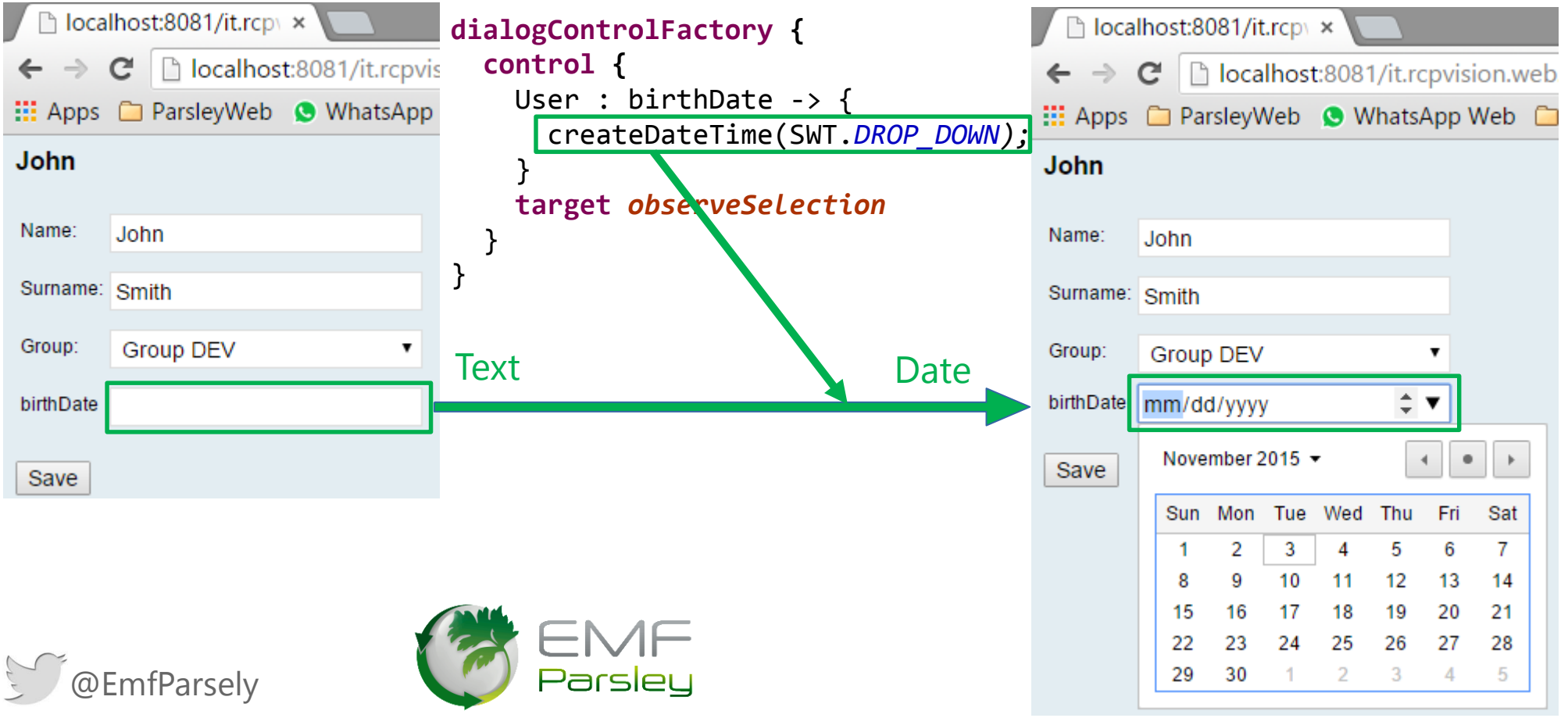
Keeping DSL & Parts customizations



@EmfParsely

Keeping SWT concepts in DSL

A thin and headless SWT layer allows to use SWT APIs in the DSL
UI customization via DSL

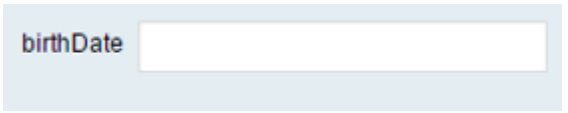


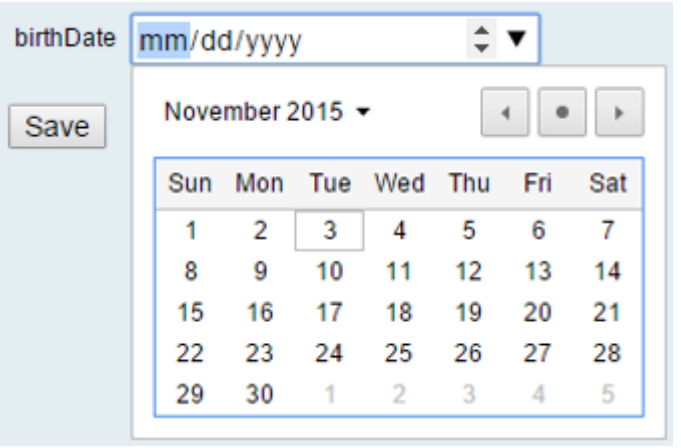
@EmfParsely

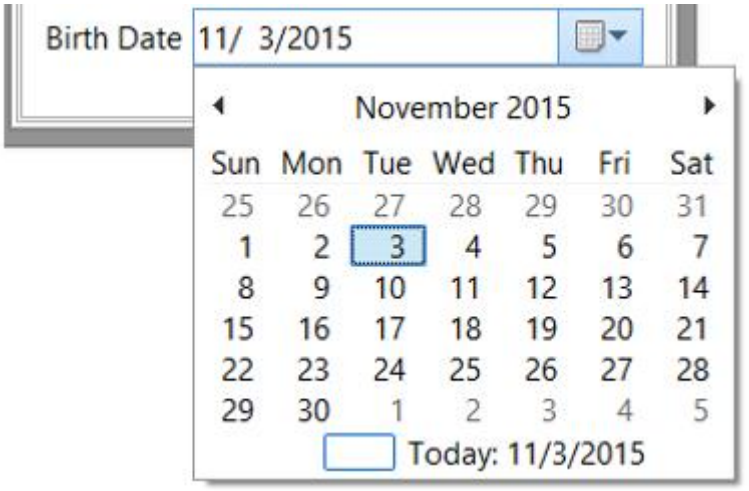
EMF Parsley

Keeping SWT concepts in DSL (cont'd: as in desktop/RCP, BTW)

Note that the DSL customization works in the same way for web and desktop implementation

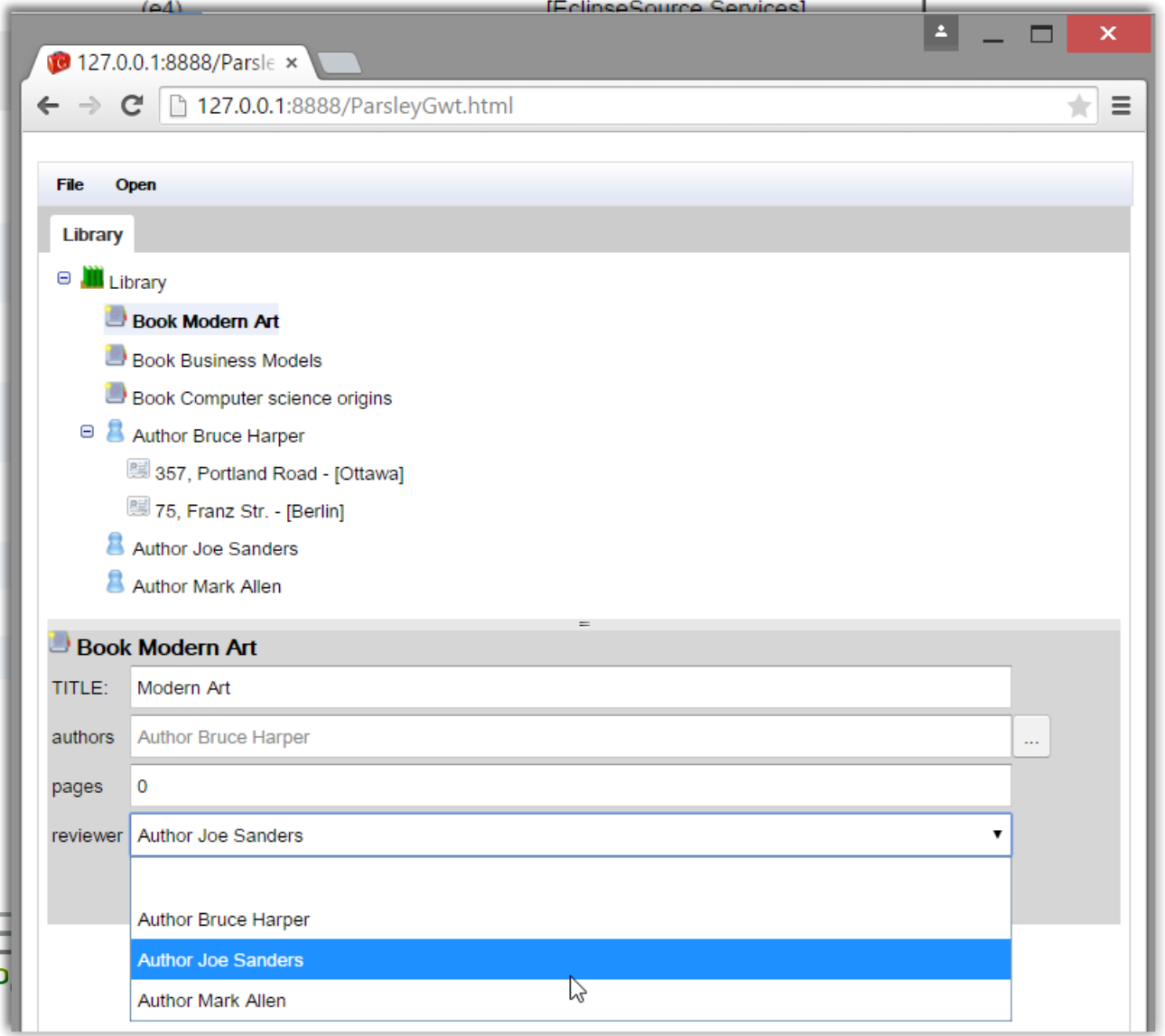






Parsley on GWT

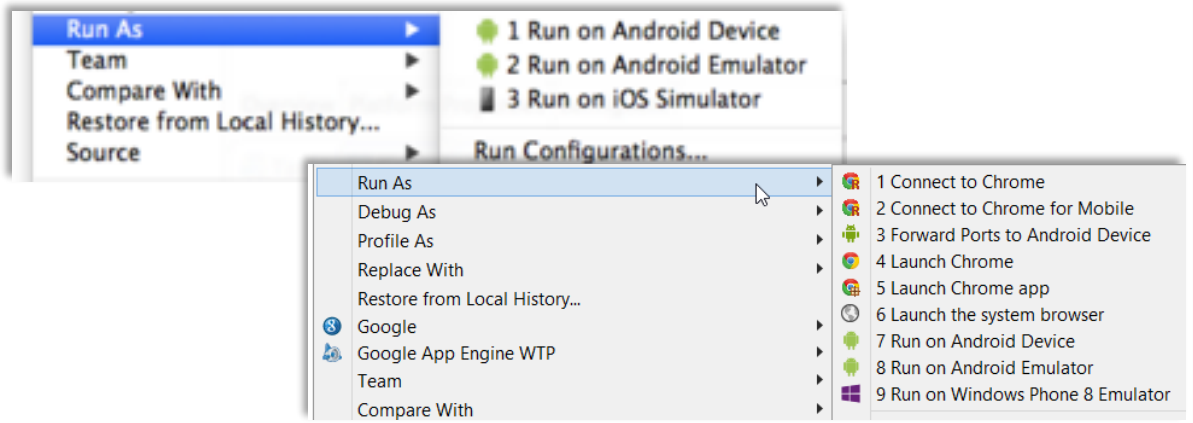
timeSlot	room	title	speakers
09:00 - 12:00	Theater Stage	What every Eclipse developer should know about Eclipse 4 (e4)	Jonas Helming [EclipseSource Munich], Philip Langer [EclipseSource Services]
09:00 - 12:00	Bürgersaal 2		
09:00 - 12:00	Silchersaal		
09:00 - 12:00	Schubartsaal		
09:00 - 12:00	Seminarraum 5		
09:00 - 12:00	Seminarräume 1-2		
09:00 - 12:00	Seminarräume 3-4		
09:00 - 12:00	FMZ Präsentationsraum		
12:00 - 12:30			
12:30 - 13:30			



@EmfParsely

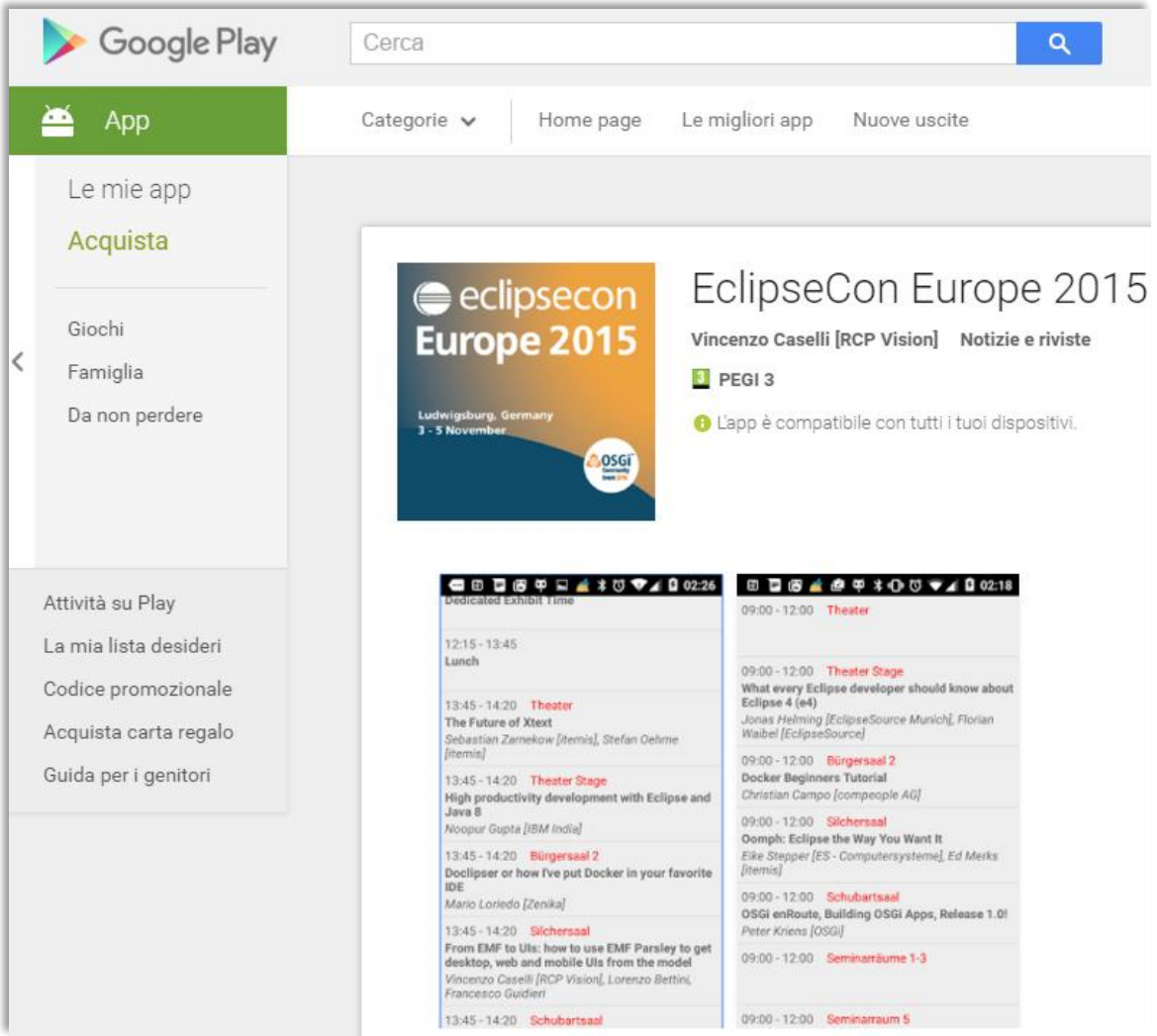
Parsley on Eclipse Thym

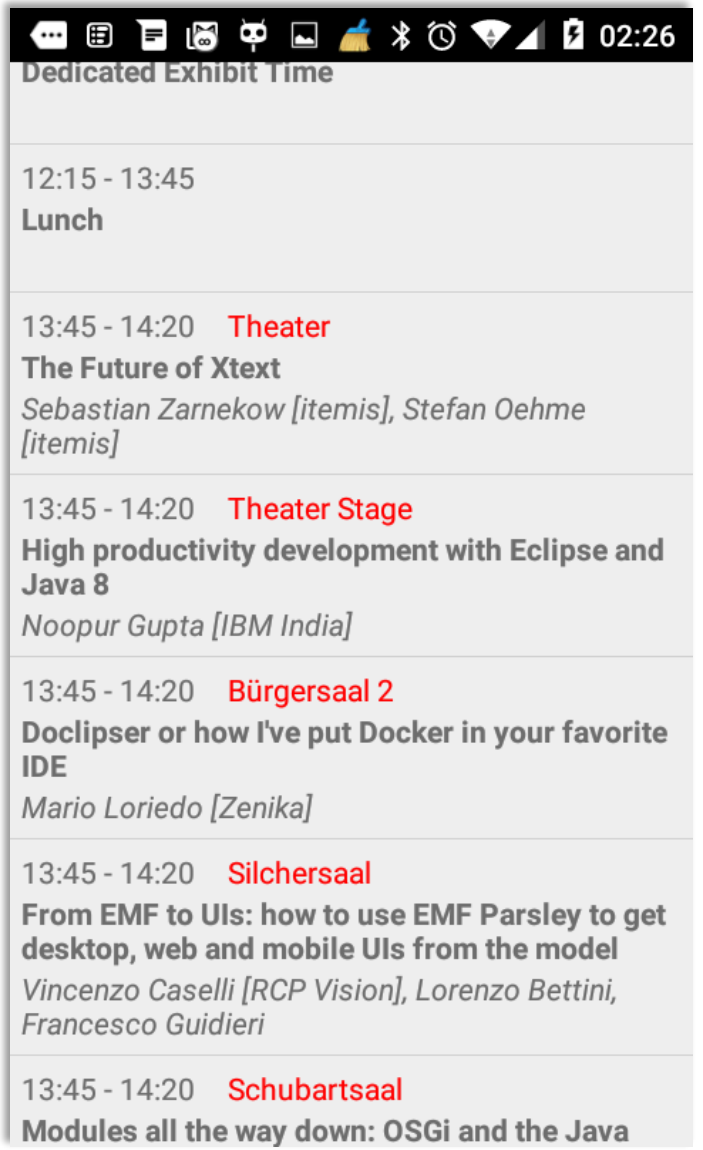
- www.eclipse.org/thym
- based on Apache Cordova
- allows cross-mobile dev
  - ✓ Android
  - ✓ iOS
  - ✓ Windows Phone



Parsley on Eclipse Andmore

- projects.eclipse.org/projects/tools.andmore
- fork of ADT





EMF Parsley resources

Homepage

- www.eclipse.org/emf-parsley

Documentation

- https://www.eclipse.org/emf-parsley/documentation.html

Forum

- https://www.eclipse.org/forums/index.php/f/263/

Bugzilla

- https://bugs.eclipse.org/bugs/buglist.cgi?product=EMF.Parsley


eclipsecon Europe
Ludwigsburg, Germany, 3 - 5 November 2015
Evaluate the sessions at www.eclipsecon.org
+1
0
-1