



EMF Forms ~~2.0~~ => 1.4 - 1.6

Jonas Helming
jhelming@eclipsesource.com



Manual coding of these UIs is a huge effort

```

Shell shell = new Shell (display);
GridLayout GridLayout = new GridLayout (5, false);
shell.setLayout (GridLayout);

Label Vorname = new Label (shell, SWT.NONE);
Vorname.setText ("Vorname");

Text VornameFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
VornameFeld.setLayoutData (data);

Label Trennlinie = new Label (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
Trennlinie.setLayoutData (data);

Label Titel = new Label (shell, SWT.BORDER);
Titel.setText ("Titel");
data = new GridLayout ();
data.horizontalIndent = 10;
Titel.setLayoutData (data);

Text TitelFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
TitelFeld.setLayoutData (data);

Label Nachname = new Label (shell, SWT.BORDER);
Nachname.setText ("Nachname");

Text NachnameFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
NachnameFeld.setLayoutData (data);

Label Geschlecht = new Label (shell, SWT.BORDER);
Geschlecht.setText ("Geschlecht");
data = new GridLayout ();
data.horizontalIndent = 10;
Geschlecht.setLayoutData (data);

Text GeschlechtFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
GeschlechtFeld.setLayoutData (data);

shell.pack ();

Shell shell = new Shell (display);
GridLayout GridLayout = new GridLayout (5, false);
shell.setLayout (GridLayout);

Label Vorname = new Label (shell, SWT.NONE);
Vorname.setText ("Vorname");

Text VornameFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
VornameFeld.setLayoutData (data);

Label Trennlinie = new Label (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
Trennlinie.setLayoutData (data);

Label Titel = new Label (shell, SWT.BORDER);
Titel.setText ("Titel");
data = new GridLayout ();
data.horizontalIndent = 10;
Titel.setLayoutData (data);

Text TitelFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
TitelFeld.setLayoutData (data);

Label Nachname = new Label (shell, SWT.BORDER);
Nachname.setText ("Nachname");

Text NachnameFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
NachnameFeld.setLayoutData (data);

Label Geschlecht = new Label (shell, SWT.BORDER);
Geschlecht.setText ("Geschlecht");
data = new GridLayout ();
data.horizontalIndent = 10;
Geschlecht.setLayoutData (data);

Text GeschlechtFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
GeschlechtFeld.setLayoutData (data);

shell.pack ();

Shell shell = new Shell (display);
GridLayout GridLayout = new GridLayout (5, false);
shell.setLayout (GridLayout);

Label Vorname = new Label (shell, SWT.NONE);
Vorname.setText ("Vorname");

Text VornameFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
VornameFeld.setLayoutData (data);

Label Trennlinie = new Label (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
Trennlinie.setLayoutData (data);

Label Titel = new Label (shell, SWT.BORDER);
Titel.setText ("Titel");
data = new GridLayout ();
data.horizontalIndent = 10;
Titel.setLayoutData (data);

Text TitelFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
TitelFeld.setLayoutData (data);

Label Nachname = new Label (shell, SWT.BORDER);
Nachname.setText ("Nachname");

Text NachnameFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
NachnameFeld.setLayoutData (data);

Label Geschlecht = new Label (shell, SWT.BORDER);
Geschlecht.setText ("Geschlecht");
data = new GridLayout ();
data.horizontalIndent = 10;
Geschlecht.setLayoutData (data);

Text GeschlechtFeld = new Text (shell, SWT.BORDER);
GridLayout data = new GridLayout (SWT.FILL, SWT.CENTER, true, false);
GeschlechtFeld.setLayoutData (data);

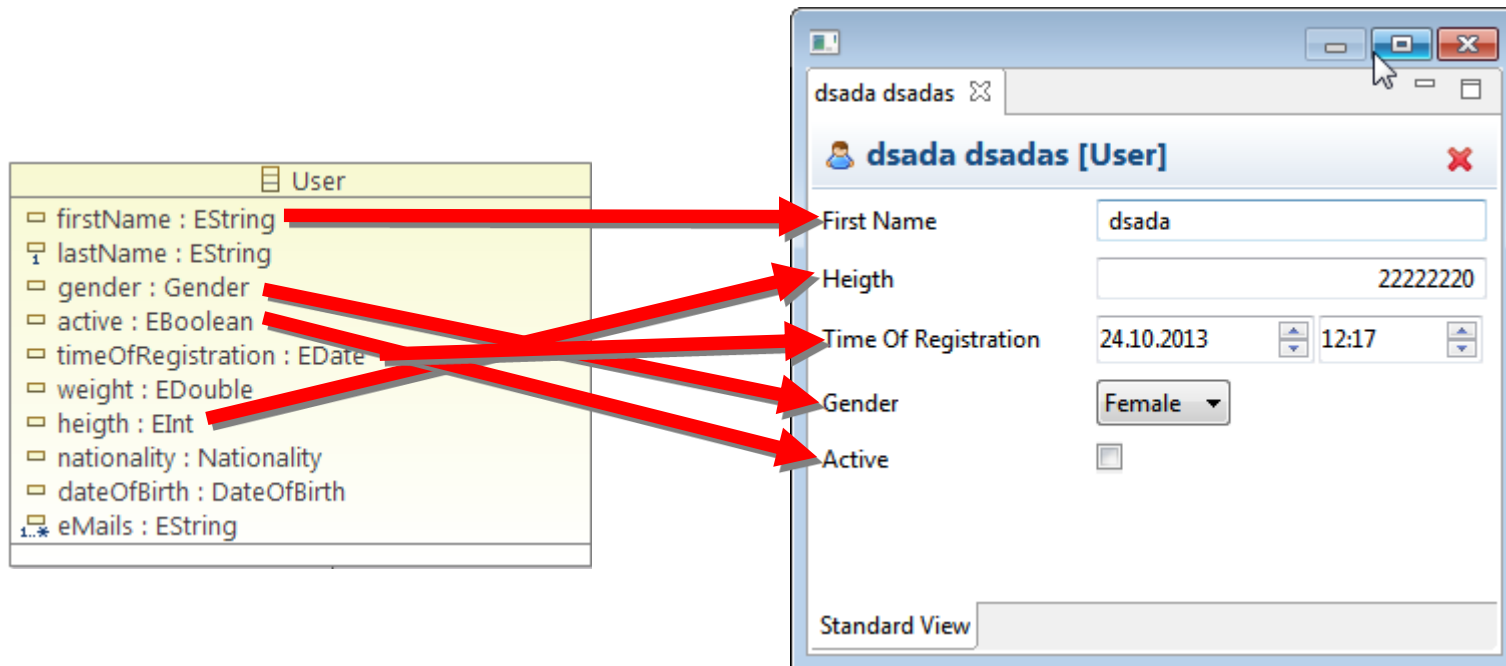
shell.pack ();

```

Effort spent for the UI of Business Applications

40%

Data model needs to be mapped to UI



Modeling the UI



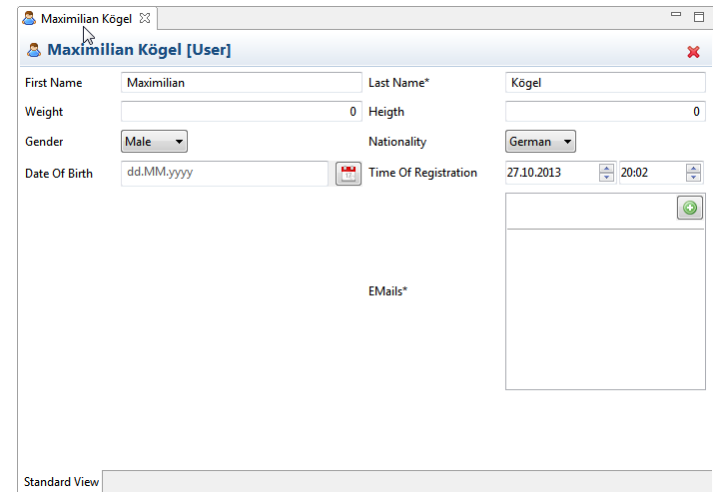
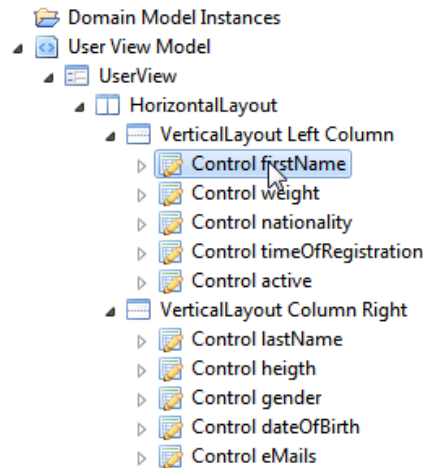
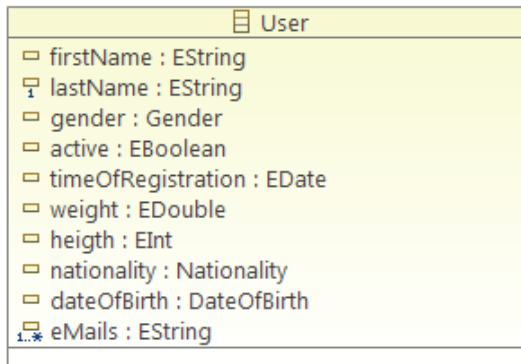
**Domain Model
(XSD, Ecore)**



View Model



User Interface



Demo

RCP Application

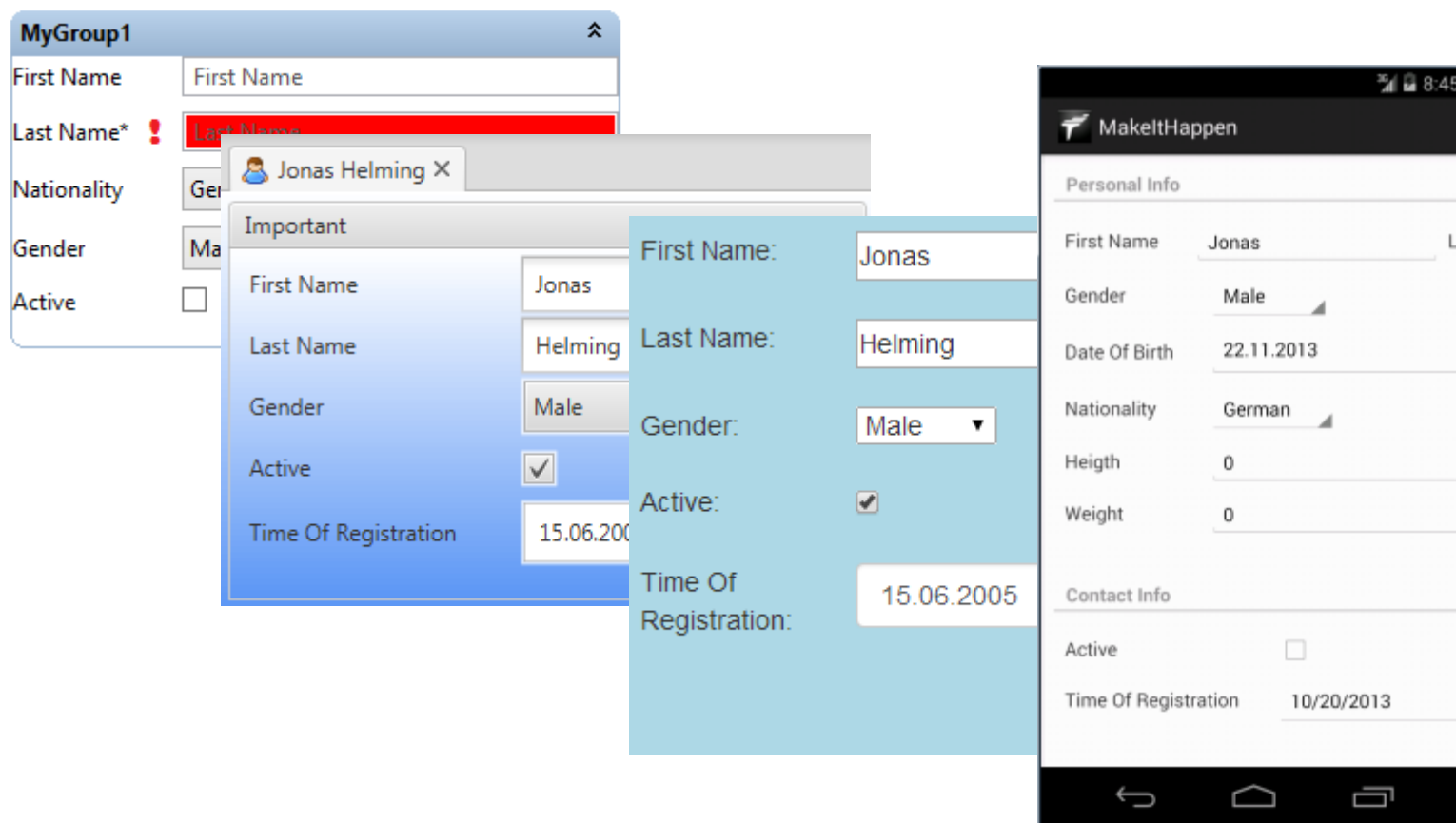
File

View

First Name	<input type="text" value="First Name"/>	Weight	<input type="text" value="0"/>
Last Name*	<input type="text" value="Last Name"/>	Height	<input type="text" value="0"/>
Gender	<input type="text" value="Male"/>	Nationality	<input type="text" value="German"/>
Active	<input type="checkbox"/>	Date Of Birth	<input type="text" value="dd.MM.yyyy"/>
Time Of Registration	No date set! Click to set date.		<input type="text" value=""/>
		Email*	<input type="text" value="Email"/>

Tasks

Rendering technologies



The image illustrates two different rendering technologies for EMF Forms. On the left, a desktop-style form titled 'MyGroup1' is shown. It features a standard desktop layout with labels on the left and input fields on the right. A modal dialog box is open over the form, displaying a user profile for 'Jonas Helming' with fields for First Name, Last Name, Gender, Active status, and Time Of Registration. On the right, a mobile-style form titled 'MakeItHappen' is shown. It features a mobile-optimized layout with a dark header, a status bar at the top showing signal strength and time (8:45), and a list of form fields with labels on the left and values on the right. The fields include First Name, Gender, Date Of Birth, Nationality, Height, Weight, Contact Info, Active status, and Time Of Registration. The mobile form also has a bottom navigation bar with back, home, and recent apps icons.

More Information

- Current Release 1.6 Mars
- Roadmap:
 - Improve Renderer SPI
 - Improve Web Renderer based on AngularJS

<http://JSONForms.org>

- **EMF Forms: <http://emfforms.org>**
- Twitter: #emfforms @EMFForms



follow us on
twitter

EMF Forms Features

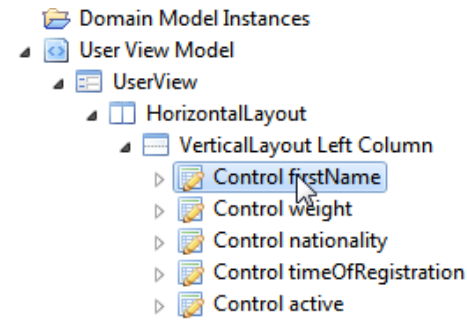
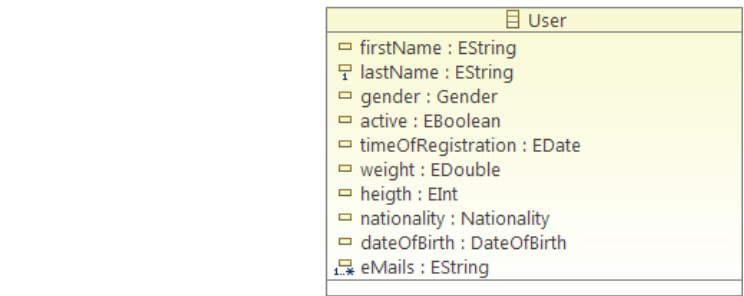
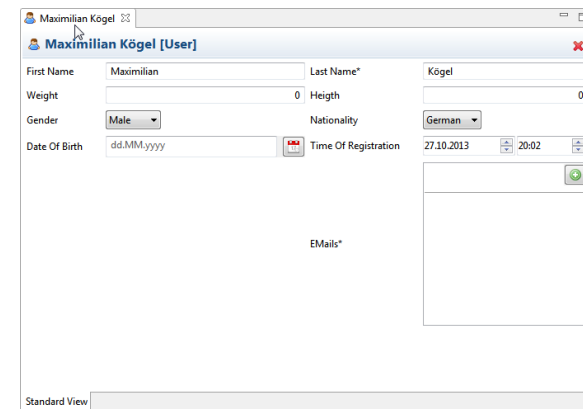


- Controls to edit data + Layouts to structure UI
- IDE-Tooling
- Many additional view model elements
- Live-Validation
- Rule-based visibility and enablement of controls
- Exchangeable UI Technology:
 - SWT (Production)
 - JavaFX (Development)
 - Browser/Web based on RAP (Production)
 - Browser/Web based on AngularJS (Development)
 - Mobile based on Tabris (Demo)

First Name	<input type="text" value="Maximilian"/>
Last Name*	<input type="text" value="Last Name"/>

When to use UI Modeling

- Large Domain Model
- Many different Views
- Frequent Domain Model changes
- Homogenous UI
- UI Technology Independence
- Improved Customer Involvement
 - Fast Turnaround + Rapid Prototyping
 - Easy-to-grasp UI concepts

Maximilian Kögel [User]

First Name	Maximilian	Last Name*	Kögel
Weight	0	Heigth	0
Gender	Male	Nationality	German
Date Of Birth	dd.MM.yyyy	Time Of Registration	27.10.2013 20:02
E-mails*			

Standard View