Make it happen!
Creating a complex form-based UI in 30 minutes

Maximilian Koegel
mkoegel@eclipsesource.com
Data is often viewed/edited in a form-based UI
Data model needs to be mapped to UI
Modeling the UI

Domain Model (XSD, Ecore) + View Model = User Interface

View Editor

- Registration
  - Categorization
    - General Information
      - VerticalLayout
      - Control title
      - Control firstName
      - Control lastName
    - Control dateOfBirth
      - Control gender
    - Dietary Requirements
      - VerticalLayout
    - Payment Information
      - VerticalLayout
Databinding with Controls

Domain Model (XSD, Ecore) + View Model = User Interface

- **Domain Model (XSD, Ecore)**: Represents the data model with attributes like `firstName` and `lastName`.
- **View Model**: Controls for `firstName` and `lastName` are created to bind data to the UI components.
- **User Interface**: The UI elements show the connected controls with their respective values.
Modeling Structure with Layouts

Domain Model (XSD, Ecore) + View Model = User Interface
Data Model and UI Model: Iteration 1
Data Model and UI Model: Iteration 2

- PersonalData
  - firstName : EString
  - lastName : EString
  - title : Title = Mrs
  - gender : Gender = Female
  - dateOfBirth : EDate

- Registration
  - id : EString
  - registrationTime : EDate

- NutritionInformation
  - diet : Diet = Carnivore

- PaymentData
  - cardnumber : EInt =
    - creditCardType : CreditCardType = Visa
    - cardholder : EString
    - expirationMonth : EInt =
    - expirationYear : EInt =
    - CVV : EInt =

- FoodRestrictions
  - peanuts : EBoolean = false
  - lactose : EBoolean = false
  - gluten : EBoolean = false
  - pork : EBoolean = false
  - other : EString

- [0..1] personalData
- [0..1] paymentData
- [0..1] nutritionInformation
- [0..1] foodRestrictions
EMF Forms Core 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 (Demo)
  • Browser/Web based on RAP (Production)
  • Browser/Web based on AngularJS (Development)
  • Mobile based on Tabris (Demo)
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
More Information

- **EMF Forms**: [http://emfforms.org](http://emfforms.org)
  - => Getting started Tutorial
  - => EMF Forms and JSON Forms Integration Tutorial
- Next Release 1.8 Mars 2
  - API/SPI improvements
  - EMF Forms Ecore Tooling
  - Web Renderer based on AngularJS: [http://JSONForms.org](http://JSONForms.org)
- Twitter: #emfforms @EMFForms
- Presentation on JSON Forms
  - Title: EMFForms goes AngularJS
  - Time and Location: Today 13:30, Town Center AB
Evaluate this Sessions

Sign in and vote at eclipsecon.org

WITH

-1 0 +1
Backup Slides
EMF Forms vs. JSON Forms

EMF Forms

- Domain Model
  (XSD, Ecore)

- View Model

- User Interface
  SWT/JavaFX/RAP/Tabris
  Renderer

JSON Forms Exporter

- JSON Forms

- JSON Schema

- UI Schema

- User Interface
  AngularJS Renderer