Web-based Modeling Tools

Jonas Helming, Maximilian Koegel
IDEs in the Web?!
Modeling in the web??!
Challenges and Mitigations

● Huge investment in existing tools! Need to reuse…
  ○ Abstract features into services (e.g. code generation, LSP)
  ○ Use declarative approaches (Modeling for Modeling Tools)

● No clear technology winner
  ○ Abstract from concrete implementation (e.g. graph engine)
  ○ Use declarative approaches (Modeling for Modeling Tools)

● No clear IDE winner
  ○ Use thin integration layer
Example: Ecore Tooling
Tree-based Editor

Ecore Model

View Model

EMF Forms

JSON Schema

UI Schema

JSON Forms
Graphical Editor

- Ecore Model
- Sirius Model
- Sirius
- JSON Schema
- JSON Graph Schema
- JSON Graph
- JointJS
- Sprotty
Integration into Che and Theia

- Che Ecore Tooling
  - Eclipse Che
- Theia Ecore Tooling
  - Standalone Editors for Ecore
  - Theia
Conclusion

● Follow design principles:
  ○ Declarative (model-based) development
  ○ Service-orientation
  ○ Abstract concrete technologies from your custom code

● When to get started?
  ○ Evaluate advantages
  ○ Iterative migration
  ○ Develop a strategy now

● Talk to us at our booth or get in contact with us:
  ○ jhelming@eclipsesource.com
  ○ mkoegel@eclipsesource.com
  ○ Developing business web applications with form-based UIs, today 14:00, Silchersaal
Evaluate the Sessions

Sign in and vote at eclipsecon.org

WITH

-1 0 +1

Thank you!