






# Engineering Model Recommender Foundations

From Class Completion to Model Recommendations

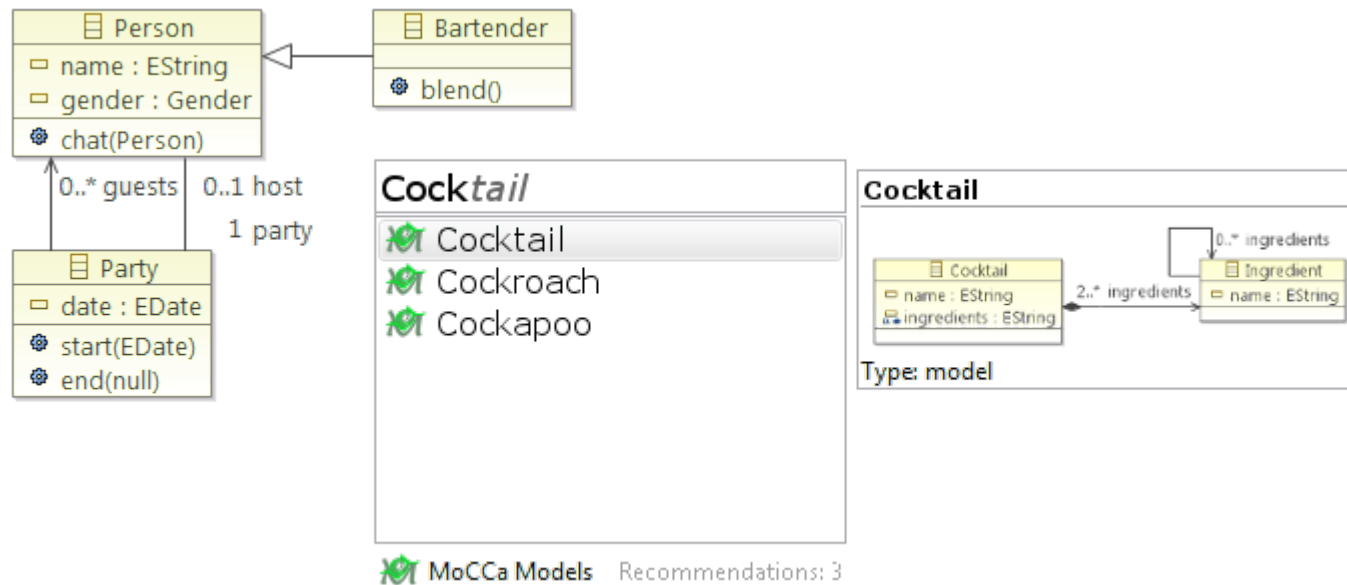
Andreas Ganser and Horst Lichter

RWTH Aachen University, Software Construction

-  If You Take One Thing
-  Setting the Scene
-  One Vision
-  Knowledge Libraries
-  Past, Present, Future

# If You Take One Thing ...

Model Recommender are fancy ;-)



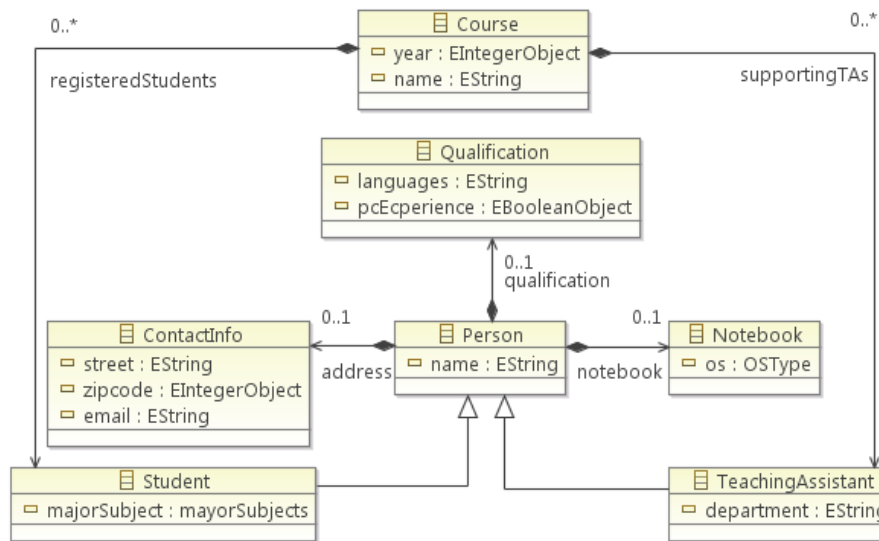
Why should I care ...?

# Example

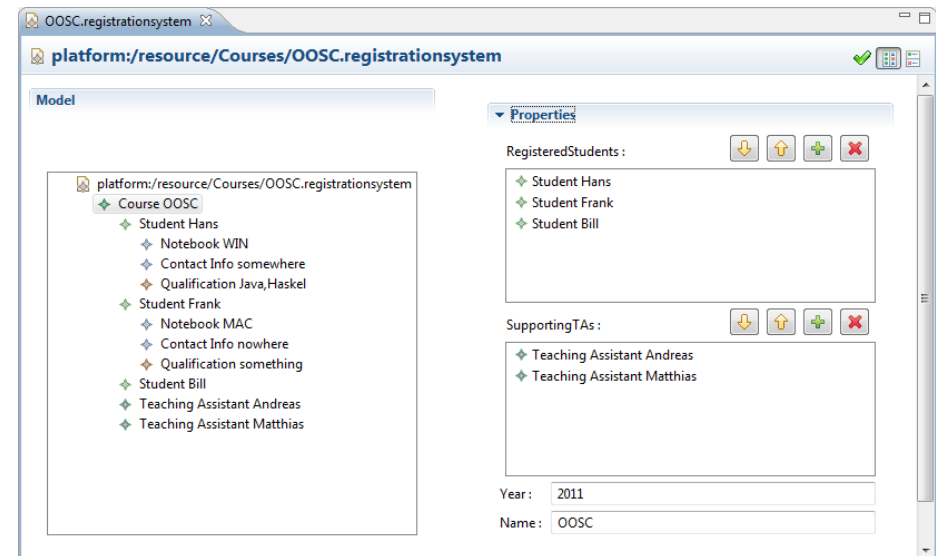
A Lecture Registration System (EMF/EEF)



## Domain Model



## Generated Editor (EEF)



How did this work ...?  
Which are the hard tasks ...?



# Setting The Scene

From Content Assist to Recommender Systems

## Content Assist

## Recommendation

```

public static void main(String[] args) {
    public Text swtText = new Text(container, SWT.NONE);
    String helloAustria = "Hallo Austria";
    String helloEngland = "Hallo England";
    String helloGermany = "Hallo Germany";
    String helloGreece = "Hallo Greece";
    String helloSwitzerland = "Hallo Switzerland";
    System.out.println(helloAustria);
}
    
```

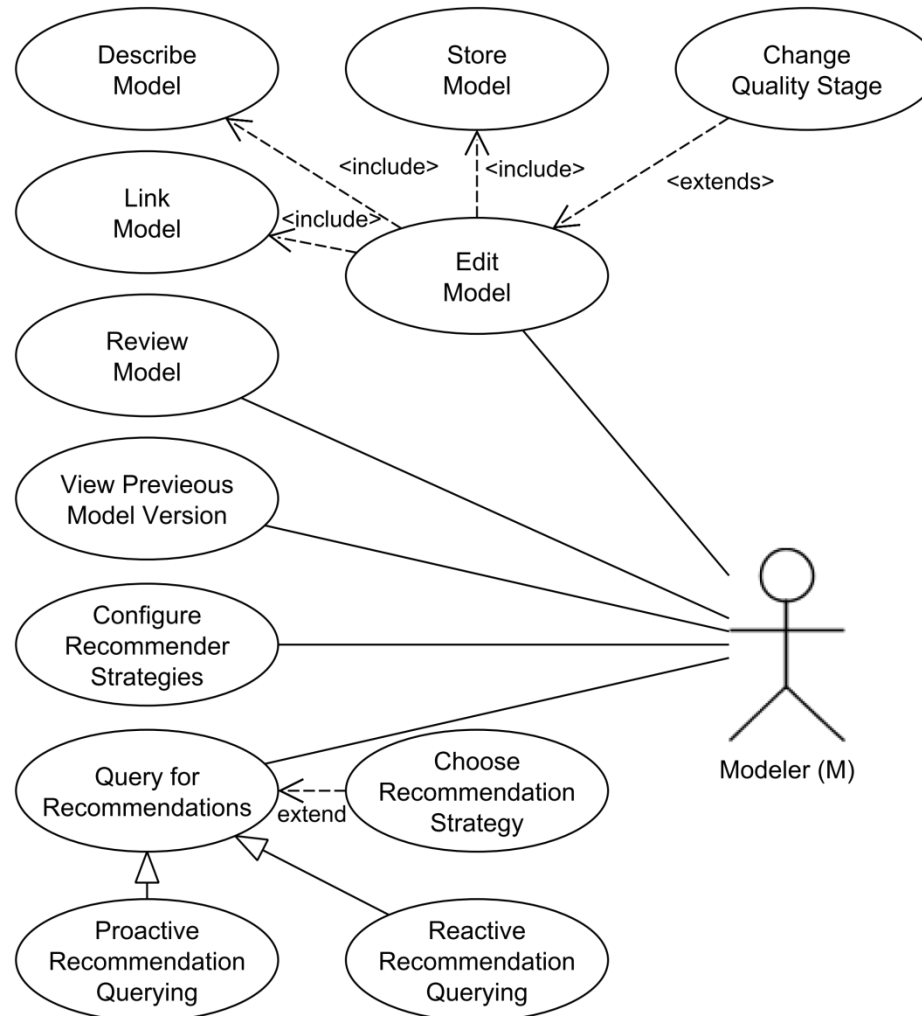
The screenshot shows an IDE with a code completion popup for the `Text` class. The popup lists various methods with their return types and class names, such as `setLayoutData(Object layoutData): void - Control - 75 %`, `setText(String string): void - Text - 44 %`, and `addListener(int eventType, Listener listener): void - Widget - 8 %`. A blue box highlights the `setText` method. A blue question mark icon is overlaid on the bottom left of the screenshot.

... and how could that look like?

Content Assist, Completion, and Recommendations for Models

What are the actual use cases?

# The Use Cases

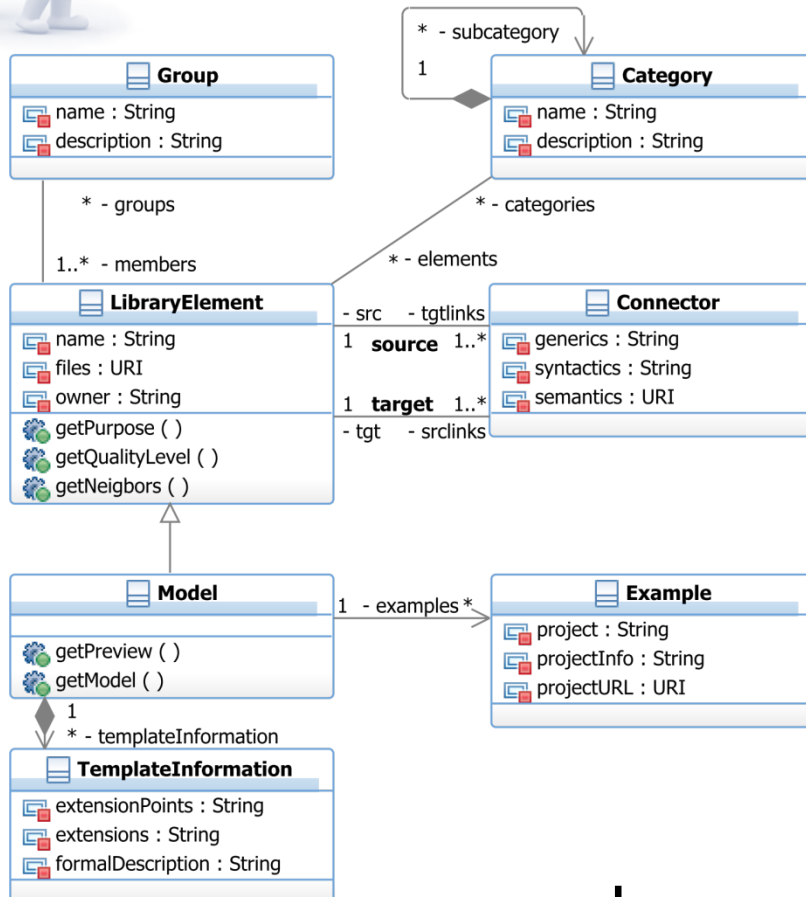


What do we need in terms of foundations?



What is the Data Structure for Suiting Recommenders best?

## Data Model



## Three Levels of Connections

- Generic
  - Description, no details
- Semantic
  - Design rationales, domain information
- Syntactic
  - Model information (UML relationships)

... so, how about recommenders?



# Model Recommendations

Multi and Hybrid-Recommender Systems

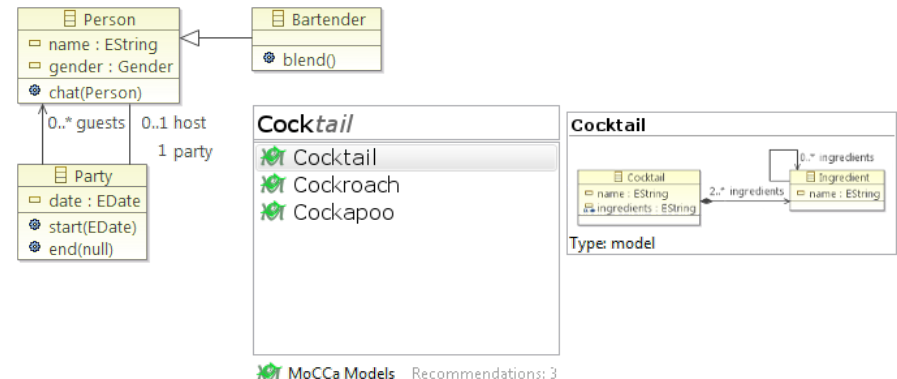


## Multi

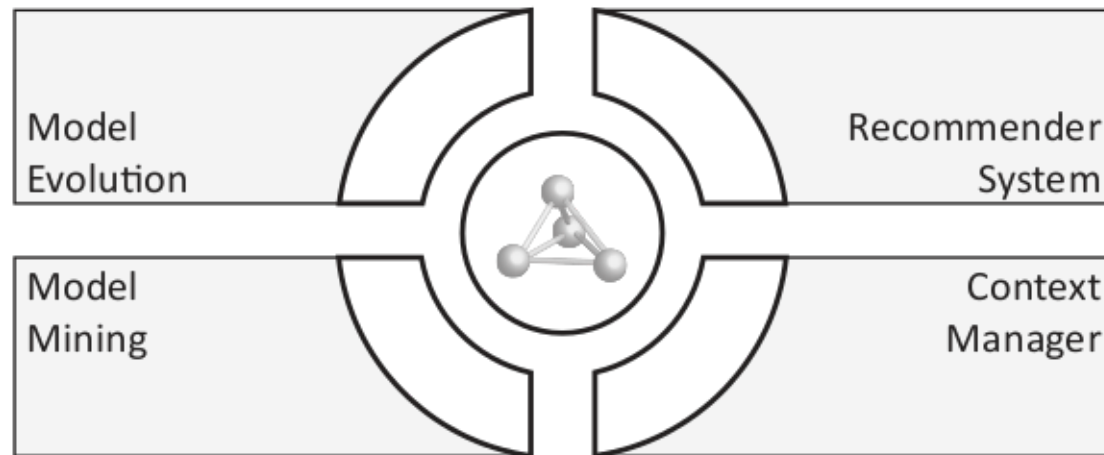
- Which are the data sources?
  - Requirements Specifications
  - Model Libraries
  - Web Services
- How to compose strategies?
  - Glossary + Synonyms
  - Glossary + Model Libraries
  - Glossary + Web Services
- Which misc. data to recommend?
  - Design Patterns
  - Domain Patterns

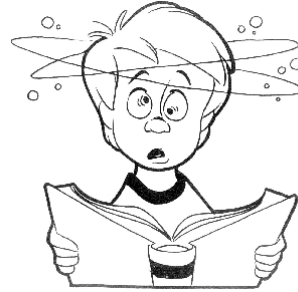
## Hybrid

- Content-Based Recommendations
  - Characteristics of items (cross-content)
- Collaborative Recommendations
  - Profiles disregarding item properties
- Chain Recommendations
  - Enhanced knowledge graph



Chain ... what!?!?





**Thanks for your attention**

... any questions?