Our responsibility

- **Backbone** - underlying support for the rest of prediction & modeling tools *(WP 4.1)*

  - Analysis support
  - Alternatives & models repository
  - *Result visualization*

- **Text-based model editors**
Core concepts – alternatives & models

- Each Q-ImPrESS project contains a repository of alternatives

- An alternative contains a set of models
  - SAMM models
  - Decorators models
  - Result models

- Alternatives constitute a tree structure
  - To support simple versioning
  - Copy of alternative includes copy of contained models
Backbone Eclipse support

Q-ImPrESS project

Working alternative

Alternative models

Repository of model alternatives

Editable model entities
Textual editor

- Edit models in textual format
- Based on Durga’s work
- Editor still needs improvements
  - Propagation of model changes
  - Template system improvements
Textual editors
Meeting

- FZI, Karlsruhe
- Participants
  - FZI, CUNI, Itemis
Meeting walkthrough agenda

- Scenarios for tools and editors
- Reverse engineering
- Modeling
- Prediction
- Trade-off analysis
- Displaying results

- Proof-of-concept implementation
  - **How to integrate tools on top of backbone plugins?**
  - Create tool flow based on described roads in D6.1
Alternatives

- Alternatives used in
  - Reverse engineering
    - Modeling is an entry point for alternatives, not reverse engineering
  - Modeling
    - Create mechanism to create, edit, store alternatives
  - Prediction
    - Select an alternative for running prediction, required for writing resulting models
  - Trade-off analysis
    - Select **alternatives**!
    - Determine trade-off between alternatives
  - Displaying results
    - Trace from alternatives to results
    - Decorators
    - Dedicated model for results?
Tools feature list

❖ Reverse engineering tools (FZI)
  ❖ Create a base SAMM model
    ▶ SiSSy
    ▶ SoMoX

❖ Input
  ▶ Top-level alternative
    ▶ Alternative Selection dialog widget is required

❖ Output
  ▶ Model filled with data
  ▶ Tool access backbone API to create own model files if needed
Tools feature list

- Editors (graphical - Itemis, textual - CUNI)
  - Input
    - Backbone plugin provides root model object
    - Backbone plugin can display model file in textual syntax
    - Graphical editors would also be called from the backbone repository alternative tree
      - One model entity can be opened by different editors

- Quality annotations
  - Create QoSAnnotation model file from the backbone repository
  - QoSAnnotationEditor ?
Tools feature list

- Analysis tools (CUNI)
  - Analysis tools started from Eclipse run dialog
    - Refactoring was proposed: workflow engine integration

- Result models are decorator models on SAMM model
  - Should a generic result model be provided?
  - How to visualize result?
    - FZI visualization library presented
Tools feature list

- Workflow engine (FZI)
  - Developed & presented by FZI
  - Job-based
  - Integrated into Eclipse
  - It would wrap current LaunchConfiguration for Q-ImPrESS tools
  - Provide common RunConfiguration tab

- Execute analysis sequentially
Meeting summary & results

- Presentation of the current tools-state
  - Refactoring tools
    - SiSSy, SoMoX
  - Backbone plugins
  - Editors

- Workflow from code to models tested

- Implementation part
  - Successful integration SoMoX with backbone plugins
  - API polished