

# Inside Eclipse

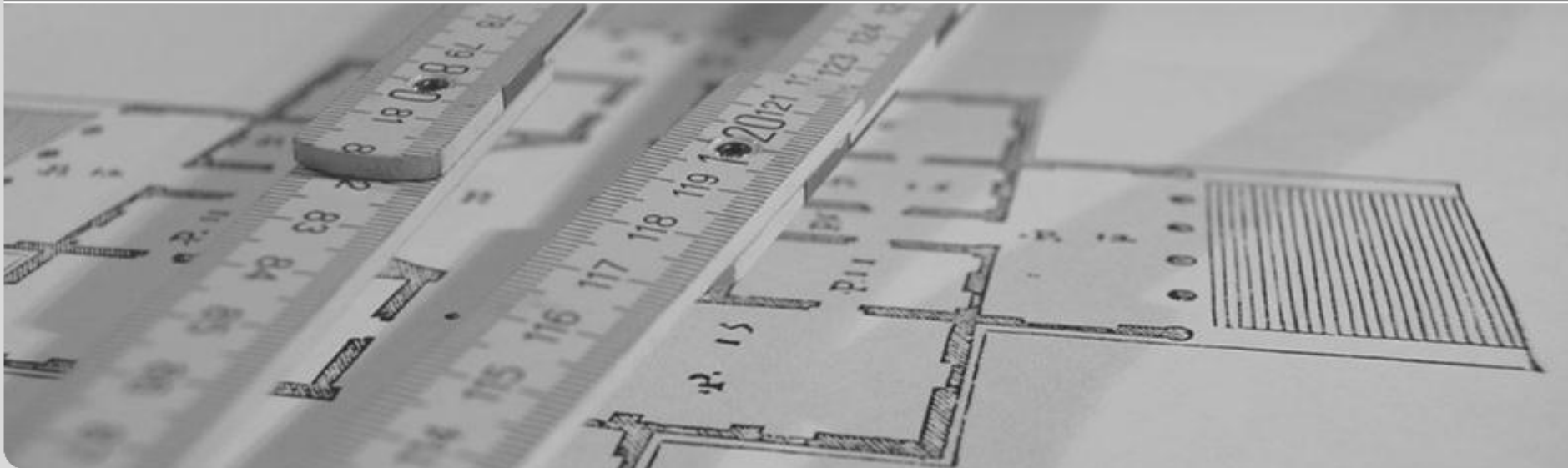
**Vorbesprechung Proseminar SomSem 2011**

**Prof. Dr. Ralf Reussner**

**Dipl.-Inform. Qais Noorshams**

SOFTWARE DESIGN AND QUALITY GROUP

INSTITUTE FOR PROGRAM STRUCTURES AND DATA ORGANIZATION, FACULTY OF INFORMATICS



- **Alle Informationen (inklusive dieser Folien) sind auch in unserem Wiki verfügbar: <http://sdqweb.ipd.uka.de/wiki/>**
- **Sie brauchen die Inhalte der Folien also nicht mitzuschreiben.**

- **Rahmenbedingungen**
- **Hinweise zur Bewertung**
- **Vorstellung der Themen**
- **Auswahl der Themen**
- **Kontakt-Informationen**

# Rahmenbedingungen

- Erstellung einer Ausarbeitung von ca. 15 Seiten (\*)
- Erstellung von Vortragsfolien (30 min Vortrag) (\*)
- Vorträge im Blockseminar, Datum: TBD
- Interne Review-Phase durch Studenten

(\*) Vorlagen im Wiki: .tex, .ppt

- Pflichtveranstaltung: Einführung in wissenschaftliches Arbeiten, Mittwoch, 20.04.2011 14:00-15:30 SR348

# Rahmenbedingungen – Termine

- **14.04.11 (Do) Vorbesprechung + Festlegung der Themen**
- **20.04.11 (Mi) Einführung ins Wissenschaftliche Arbeiten**
- **11.05.11 (Mi) Abgabe der Gliederung und Literaturliste**
- **15.06.11 (Mi) Abgabe einer Review-Version**
- **22.06.11 (Mi) Abgabe der Reviews**
- **29.06.11 (Mi) Abgabe der Ausarbeitung**
- **04.07.11 (Mo) Abgabe der Folien**
- **04.07.11 (Mo) Feedback zur Ausarbeitung durch Betreuer**
- **08.07.11 (Fr) Feedback zu den Folien durch Betreuer**
- **17.07.11 (So) Abgabe der finalen Ausarbeitung + Folien**
- **TBD (-- ) Blockseminar (wird angekündigt, vsl. Ende Juli)**
- **Weitere Termine mit den Betreuern nach Absprache**

- **Zum Bestehen des Proseminars sind notwendig:**
  - **Abgabe von Ausarbeitung, Reviews, und Folien**
  - **Teilnahme am Blockseminar**
  - **Teilnahme an der Einführung ins Wissenschaftliche Arbeiten**
- **Punktesystem**
  - **Abgaben**  
(Gliederung und Literaturliste, Ausarbeitung 1. Version, Reviews, Ausarbeitung 2. Version, Präsentation 1. Version, Ausarbeitung finale Version, Präsentation finale Version)
  - **pünktliche Abgabe: 4 Punkte**
  - **am Tag nach der Deadline: 2 Punkte**
  - **2 Tage nach der Deadline: 1 Punkt**
  - **3 und mehr Tage nach der Deadline: 0 Punkte**

# Hinweise zur Bewertung

- **Punktesystem (Fortsetzung)**
  - **vollständige Gliederung: 2 Punkte**
  - **eigene Literatur gesucht, gefunden: 2 Punkte**
  - **vollständiges Review: 2 Punkte**
  - **Ausarbeitung nach Richtlinien: bis zu 8 Punkte**
  - **Folien nach Richtlinien: bis zu 6 Punkte**
  - **guter Vortrag: bis zu 2 Punkte**
- **Maximalpunktzahl 60 Punkte**
- **Notenbereich:**
  - **30-37 Punkte: 4**
  - **38-45 Punkte: 3**
  - **46-53 Punkte: 2**
  - **54-60 Punkte: 1**

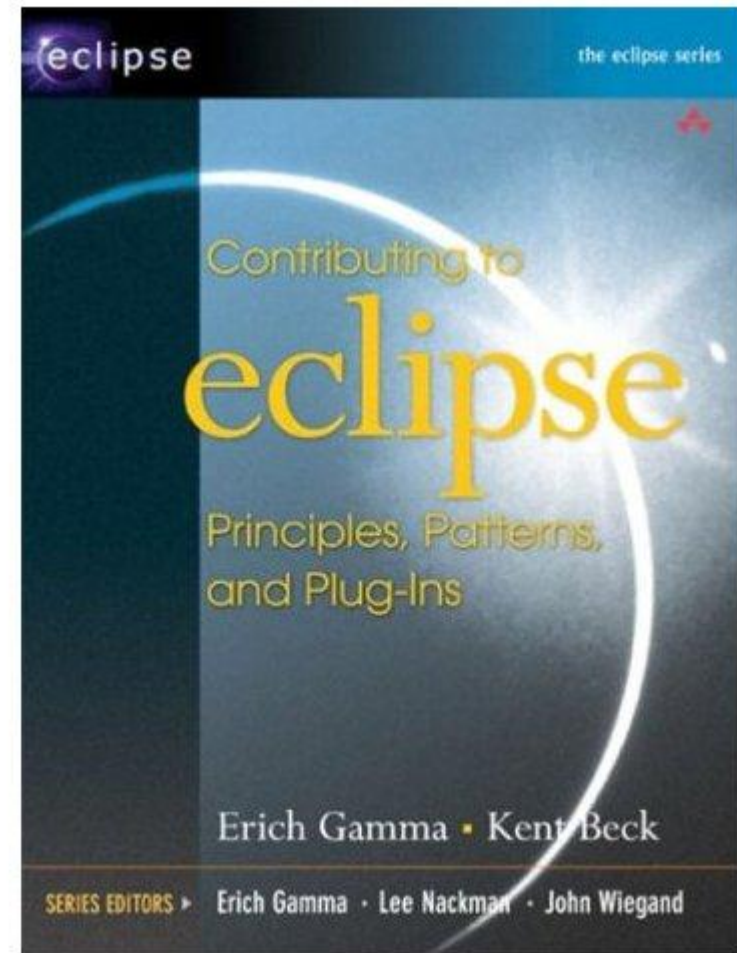
- **Punktesystem (Fortsetzung)**
  - **Innerhalb des Notenbereichs kann sich die Note noch um +/- 1 verbessert/verschlechtern. Dazu fließen in die Note noch zusätzlich ein:**
    - **Inhalt/Qualität der Ausarbeitung**
    - **Inhalt/Qualität der Folien**
    - **Qualität der Diskussionsbeiträge im Blockseminar**



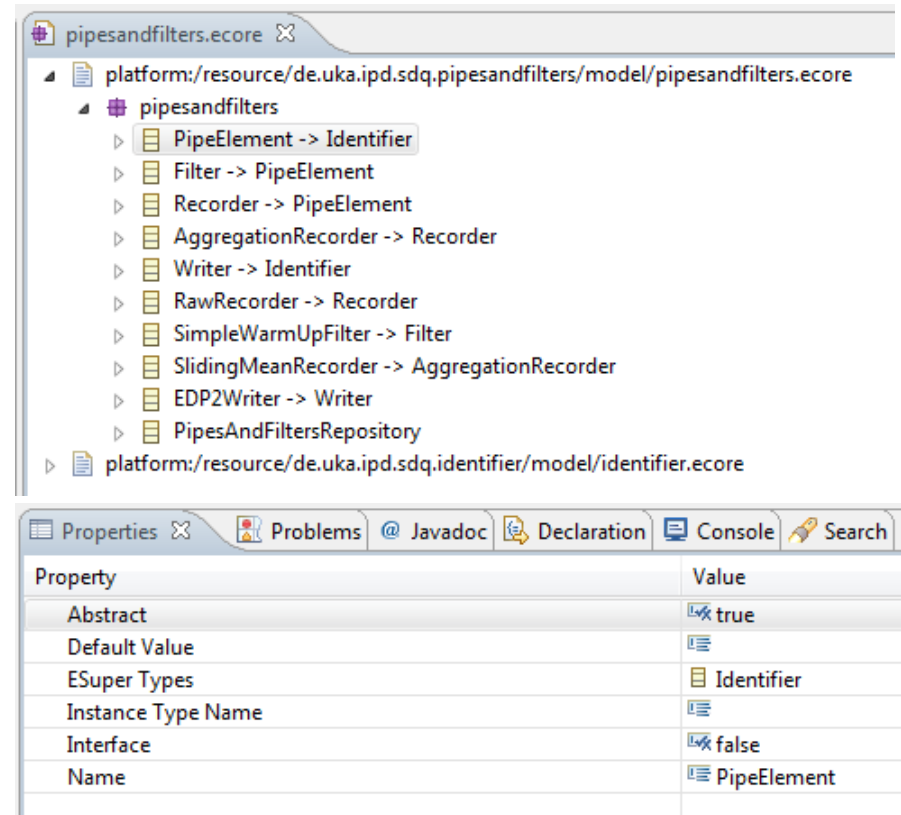
- **Themenblock: Musterverwendung in Eclipse**
  - **Adapter**
  - **Fabrik**
  - **Model View Controller**
  - **Presentation Abstraction Control**
- **Plug-in-Mechanismus von Eclipse**
  - **Versionierung von Plug-Ins in Eclipse**
- **Mylyn**

# Themenblock Muster

- **Im Eclipse Projekt werden viele Entwurfsmuster benutzt.**
- **Vorteile von Mustern:**
  - bewährte Lösungen
  - Design leichter verständlich
  - Vokabular
- **Im Seminar:  
Lernen, wie Muster in der Praxis angewandt werden.**
  - Benutzung in Eclipse
  - Prinzipien und Nutzen



- Wird benutzt um UI Funktionalitäten hinzuzufügen ohne die Basisklassen zu "verschmutzen".
- Beispiel: Properties View
- AdapterFactories: Interfaces zu bestehenden Typen hinzufügen

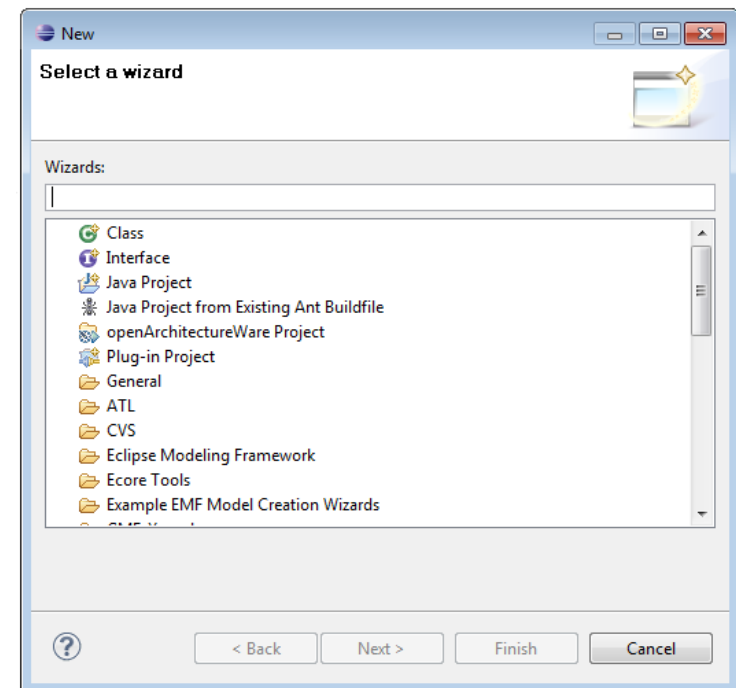


The screenshot shows the Eclipse IDE interface. The top part displays a project structure for 'pipesandfilters.ecore'. The 'pipesandfilters' package contains several classes and interfaces, including 'PipeElement -> Identifier', 'Filter -> PipeElement', 'Recorder -> PipeElement', 'AggregationRecorder -> Recorder', 'Writer -> Identifier', 'RawRecorder -> Recorder', 'SimpleWarmUpFilter -> Filter', 'SlidingMeanRecorder -> AggregationRecorder', 'EDP2Writer -> Writer', and 'PipesAndFiltersRepository'. Below the project structure, the 'Properties' view is open, showing the following table:

Property	Value
Abstract	true
Default Value	
ESuper Types	Identifier
Instance Type Name	
Interface	false
Name	PipeElement

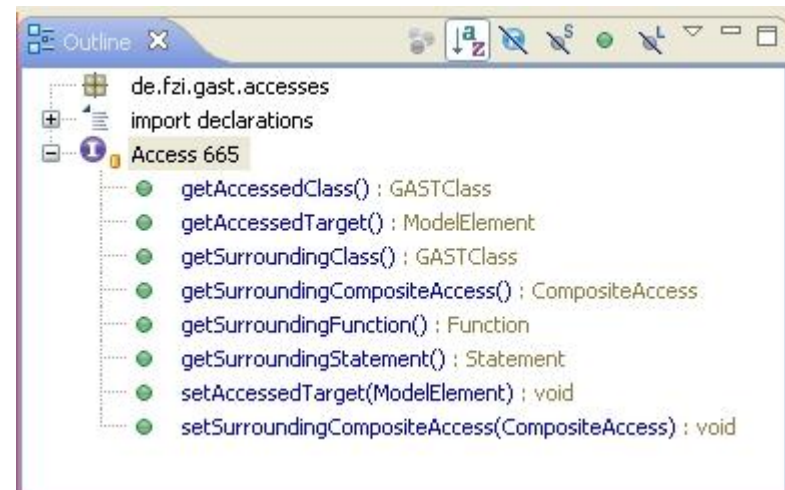
# Fabrik

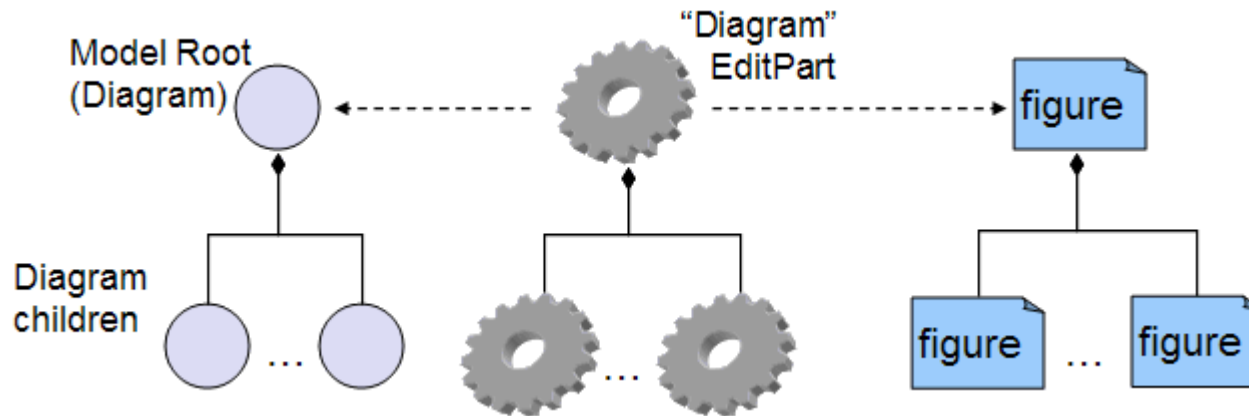
- **Objektgenerierung**
- **die instantziierte Klasse wird austauschbar**
- **Abstrakte Fabrik:**
  - **Zentralisiert die Entscheidung welche Familie von Produkten erzeugt wird**
- **Fabrikmethode:**
  - **Den Unterklassen die Entscheidung überlassen, welche Klasse instantziiert wird.**
- **Beispiel: ActionFactory**
  - **Zugriff auf Standardoperationen die die Workbench bereit stellt**



# Model View Controller (MVC)

- **Teilung von Daten, Visualisierung und Kontrolle**
- **Spezielle Anwendung des Beobachter Musters**
- **Beispiel: Kombination des Java-Editors und Outline-Ansicht**





- Hierarchisches MVC
- Wiederfindung in Eclipse GEF und GMF
- Propagierung von Änderungen in der Hierarchie



- **Erweiterbarkeit ist eine Grundanforderung an Software**
- **Eclipse kennt saubere Schnittstellen für die Integration von Erweiterungen:  
Der Plug-In-Mechanismus**
- **In XML spezifizierte Extensions und Extension Points geben verfügbare und benötigte Softwaredienste von Plugins an**



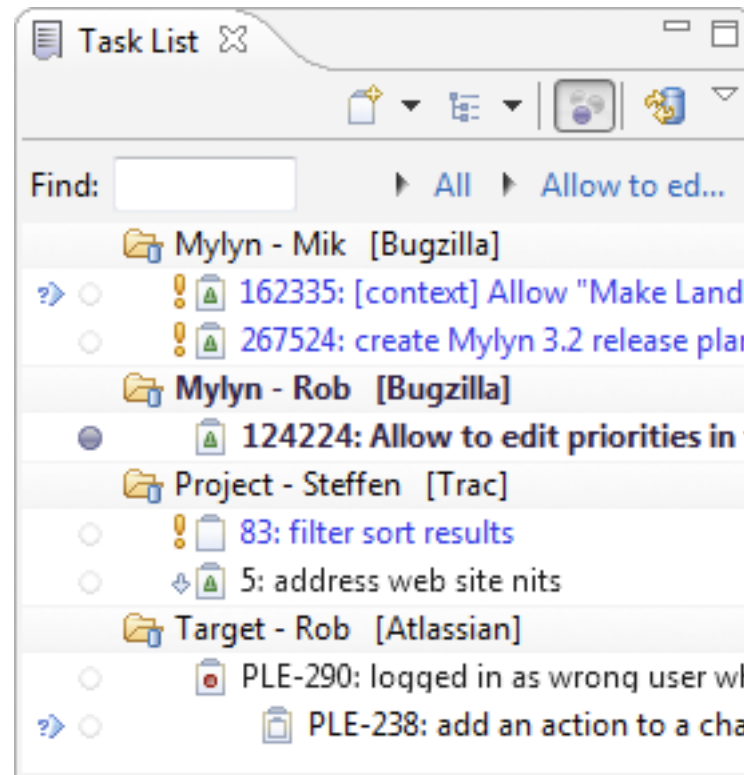
Version 1.0.3

- Dependencies:
- Min: 0.1.0
- Max: 1.3.1

- Die Versionierung von Plug-Ins ist notwendig um eine unabhängige und verteilte Entwicklung von Plug-Ins zu ermöglichen
- Ziel bei Eclipse: Vermeidung der Microsoft „DLL-Hell“
- Wie lautet die Version eines Plugins?
- Welche Plugins funktionieren zusammen?
- Funktioniert ein Plugin auch mit einem älteren / neueren anderen Plugins zusammen?



- Task-fokusiertes Interface
- Integration von Bugzilla, Trac und JIRA



# Weitere Lehrveranstaltungen

## IPD Reussner im Sommersemester 2011

- **2 Vorlesungen**, prüfbar (MSc, Diplom) im Vertiefungsfach "Softwaretechnik und Übersetzerbau"
  - **Komponentenbasierte Software-Architektur (neu)**, Prof. Dr. Ralf Reussner, Dienstags, 15.45-1715 im Gaede-HS (Gebäude 30.22)
  - **Performance Engineering of Enterprise Software Systems**, Dr. Samuel Kounev, Mittwochs, 11:30-13:00 im HS -101
  
- **Lesegruppe Softwaretechnik**
  
- **Weitere Infos in unserem Wiki:**  
<http://sdqweb.ipd.kit.edu/wiki>

- **Diplom-/Master-/Studien-/Bachelor-Arbeiten & Hiwi-Jobs**

[http://sdq.ipd.kit.edu/diploma\\_theses\\_study\\_theses/](http://sdq.ipd.kit.edu/diploma_theses_study_theses/)

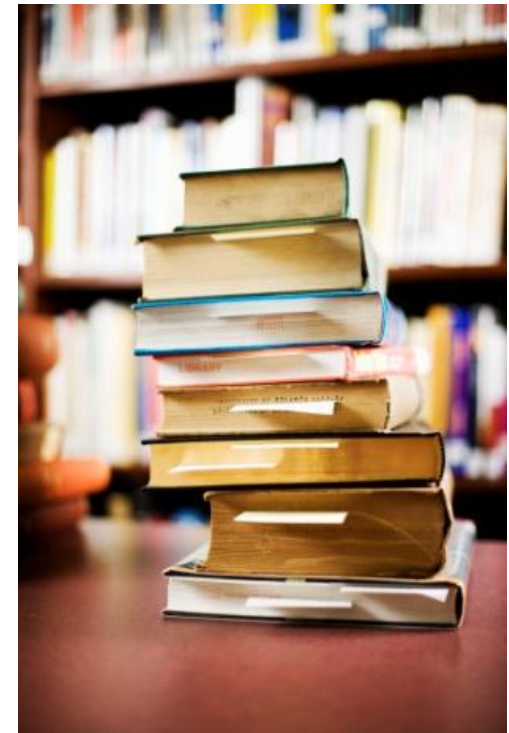


- **Themen vorschlagen**

[http://sdq.ipd.kit.edu/diploma\\_theses\\_study\\_theses/search\\_topic\\_the\\_ma\\_suchen/](http://sdq.ipd.kit.edu/diploma_theses_study_theses/search_topic_the_ma_suchen/)

- **Weitere Infos in unserem Wiki:**

<http://sdqweb.ipd.kit.edu/wiki>



- **Themenblock: Musterverwendung in Eclipse**
  - Adapter
  - Fabrik
  - Model View Controller
  - Presentation Abstraction Control
- **Plug-in-Mechanismus von Eclipse**
  - Versionierung von Plug-Ins in Eclipse
- **Mylyn**

- **Wenn Sie weitere Fragen haben:**
  - **Adresse unseres Wikis:**  
<http://sdqweb.ipd.uka.de/wiki/>
  - **Kontakt:**  
[noorshams@kit.edu](mailto:noorshams@kit.edu)