

Abschlusspräsentation

Praktikum: Modellgetriebene Software-Entwicklung

Dirk Neumann, Patrick Mehl, Jakob Höfker | 8. Dezember 2020

Inhaltsverzeichnis

1. M1: Meta-Modellierung
2. M2: Xtext - textuelle Syntax
3. M3: QVTo - Modelltransformation
4. M4: Xtend - Code-Generierung
5. M5: Sirius - graphischer Editor
6. Evaluation
7. Key Learnings
8. Zusammenfassung

M1: Metamodell
○○○

M2: Xtext
○○

M3: QVTo
○○

M4: Xtend
○○

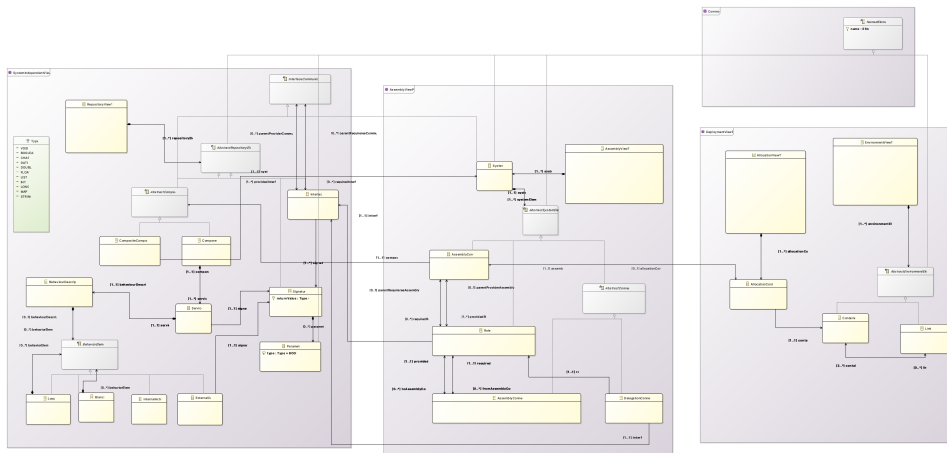
M5: Sirius
○○

Evaluation
○

Key Learnings
○

Zusammenfassung
○

Meta-Modellierung



M1: Metamodell

●○○

M2: Xtext

○○

M3: QVTo

○○

M4: Xtend

○○

M5: Sirius

○○

Evaluation

○

Key Learnings

○

Zusammenfassung

○

Entwurfsentscheidungen für die Meta-Modellierung

- Verwendung von 4 Packages
 - klare Trennung der View Points
 - Nachteil für spätere Entwicklung
- Common Package für wiederverwendbare Elemente
- ENUM zur Modellierung des Type
- Verwendung von OCL

Erstellung der Komponenten für den Media Store

- mediastore_allocation.deploymentviewpoint
- mediastore_environment.deploymentviewpoint
- mediastore.assemblyviewpoint
- mediastore.systemindependentviewpoint

Xtext - textuelle Syntax

- Ergebnis: mediastore.envvt

```
EnvironmentViewType {
    environmentElements {
        Container ApplicationServer,
        Container DatabaseServer,
        Link Network {
            containers (ApplicationServer, DatabaseServer)
        }
    }
}
```

Entwurfsentscheidungen und Probleme

- import von Packages
 - Lösung:
- OCL Constraints

- ## Zusammenfassung

Entwurfsentscheidungen für die Modelltransformation

- Eine Datei, da Import Probleme gegeben hat
- Probleme
 - OCL Constraints

Code-Generierung mit Xtend

```

package DBCache;

import repository.AudioDB;

public class DBCacheImpl implements AudioDB {
    private AudioDB audioDB;

    public DBCacheImpl() {}

    public void setAudioDB(AudioDB audioDB) {
        Helper.assertNull(this.audioDB);
        this.audioDB = audioDB;
    }

    //Implementing addFile from interface AudioDB
    @Override
    public void addFile () {
        Helper.assertNotNull(this.audioDB);
        //TODO: implement
    }

    //Implementing queryDB from interface AudioDB
    @Override
    public void queryDB () {
        Helper.assertNotNull(this.audioDB);
        //TODO: implement
    }
}

```

Entwurfsentscheidungen für Code-Generierung

- Mapping der einzelnen Modelle Java-Code
- Probleme
 - OCL Constraints, gelöst mit eigenem Validator

Sirius - graphischer Editor

M1: Metamodell
○○○

M2: Xtext
○○

M3: QVTo
○○

M4: Xtend
○○

M5: Sirius
●○

Evaluation
○

Key Learnings
○

Zusammenfassung
○

Entwurfsentscheidungen und Probleme

- eigene Datentypen

M1: Metamodell
○○○

M2: Xtext
○○

M3: QVTo
○○

M4: Xtend
○○

M5: Sirius
○●

Evaluation
○

Key Learnings
○

Zusammenfassung
○

Evaluation



M1: Metamodell
○○○

M2: Xtext
○○

M3: QVTo
○○

M4: Xtend
○○

M5: Sirius
○○

Evaluation
●

Key Learnings
○

Zusammenfassung
○

Key Learnings



M1: Metamodell
○○○

M2: Xtext
○○

M3: QVTo
○○

M4: Xtend
○○

M5: Sirius
○○

Evaluation
○

Key Learnings
●

Zusammenfassung
○

Zusammenfassung



M1: Metamodell
○○○

M2: Xtext
○○

M3: QVTo
○○

M4: Xtend
○○

M5: Sirius
○○

Evaluation
○

Key Learnings
○

Zusammenfassung
●