

SWK
Übungsblatt 3
Gruppe 2: OH12 / 1.056

Max Springenberg, 177792

3.1 Die Modellierungs-Hierarchie

3.1.1

Ein Metamodell beschreibt Modelle. GraphML beschreibt xml-files, die als Graphen interpretiert werden können. Damit sind die Modelle von GraphML xml-files.

3.1.2

- (i) die abstrakte Syntax von GraphML definiert die zugrundelegenden Datenobjekte. So behandelt GraphML Knoten, Kanten, bzw. Graphen im Allgemeinen, sowie ihre Darstellung.
- (ii) Die konkrete Syntax von GraphML definiert Schlüsselerwörter und die Grammatik der Sprache, z.B. die Schlüsselwörter node, edge, graph, key, data und das Verwenden der xml-üblichen Klammerung.
- (iii) Die Semantik von GraphML ist durch das Auswerten der Inhalte für die jeweiligen geklammerten Schlüsselwörter zu Datenobjekten definiert.

3.1.3

XML ist Turingvollständig, damit können alle für die Informatik relevanten mathematischen Modelle beschrieben werden.

3.1.4

Wie gesagt ist XML Turingvollständig, damit kann man die Kontextsensitiven-Grammatiken als Metamodell für Programmiersprachen im Allgemeinen erachten.

XML ist in der Sprache der Kontextsitiven-Grammatiken enthalten, eine Modell-Instanz vom Metamodell 'Kontextsensitive-Grammatik' ist die Sprache XML.

Alternativ: XML Schema Definition Language, was im Internet als das offizielle Metamodell von XML gelistet wird.

3.2 UML

3.3 Anforderungs-DSLs

```
(i)
requirement IMPLEMENTATIONTHREAD organization {
    implemented-in developers;
    urgency must;
    description "customers implementation-requirements";
    action follow;
};
(i.i)
requirement OPENSOURCE_PL organization {
    required-by IMPLEMENTATIONTHREAD;
    implemented-in developers;
    urgency must;
    description "open source programming language";
    action follow;
};
(i.ii)
requirement OPENSOURCE_ALL organization {
    required-by IMPLEMENTATIONTHREAD;
    implemented-in developers;
    urgency should;
    description "everything should be open-source";
    action follow;
};
(ii)
requirement DATAEXPORT product {
    implemented-in developers;
    urgency should;
    automatic;
    description "automatic data export";
    action export;
};
(iii)
// Der Einleitungstext konnte auch als Anforderung aufgefasst werden,
// deshalb die zweiteilung der Anforderung.
requirement EXTENDTOOL system {
    implemented-in developers;
    urgency must;
    description "extend backup tool";
    action implement;
};
requirement PROHIBITDEL organization {
    required-by EXTENDTOOL;
    implemented-in developers;
    urgency must not;
    description "admin must not delete latest backup";
    action delete;
```

```
};  
(iv)  
requirement PROHIBITTOBI_WEB user {  
    implemented-in component WEB_FRONTEND;  
    urgency may not;  
    usecase-for TOBI;  
    description "don't let Tobi get access to salary-data";  
    action access;  
};
```