



Technische
Universität
Braunschweig

Institut für Softwaretechnik
und Fahrzeuginformatik



BibTeX2HTML

HTML5-Visualisierung von BibTeX-Einträgen mit Hilfe einer DSL

Marcus Stelke, Sven Frank, Frieder Berthold und Alexander Knüppel,
14. Februar 2015

Motivation

- **Gegeben:** eine (mehrere) BibTeX-File(s)
- **Gewünscht:** angepasste und individuelle Darstellung der Inhalte von BibTeX-Files als HTML (z.B. für Webseiten)
- **Probleme:**
 - händisch durch Copy & Paste zu aufwendig
 - Schlecht wartbar
 - Fehleranfällig
 - Daten synchronisieren
- **Lösungsansatz:** Entwerfen einer textuellen DSL

Projektziel - HTML5-Visualisierung von BibTeX-Dateien

Modellierung mit Xtext

- Sprache für das BibTeX-Format
- Sprache zur Konfiguration der Visualisierung
 - Sortierung
 - Benutzerspezifische Gestaltung (Farbe, Schrifttyp etc.)
 - Ausgabe eines Referenzkürzels
- Validierung von Constraints



Transformation

- T2M-Transformation durch Xtext (Ecore-Modelle für beide DSLs)
- M2T-Transformation durch Xtext/Xtend2 (Templating)

Ausgabe

- Valides HTML5-Dokument mit Einträgen aus angegebenen BibTeX-Dateien inklusive CSS

Workflow

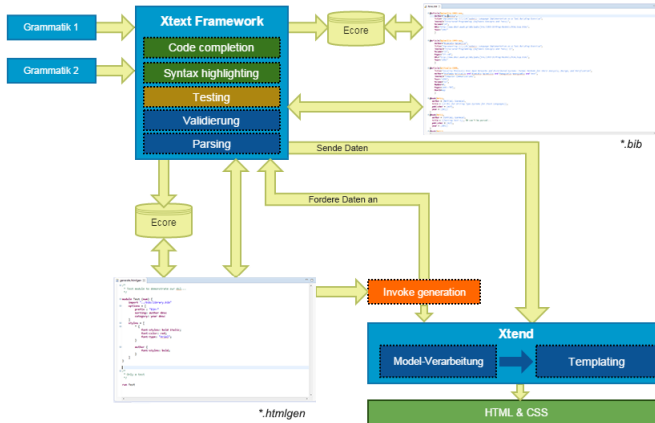


Abbildung 1: Workflow

BibTeX-Dateiformat

- '@' + Publikations-Typ
- Schlüsselwort
- Notwendige und optionale Tags

```
@article{article01:2015,  
  Author="Marcus Stelke and Sven Frank and Frieder Berthold and Alexander Kn\"{u}ppel",  
  Title={BibTeX to HTML5: simple as that},  
  Journal="MBSE (Software Concepts and Tools)",  
  Volume=1,  
  URL="http://www.comingsoon.de/",  
  Year=2015,  
  month=jan  
}
```

Ecore-Metamodell: BibTeX

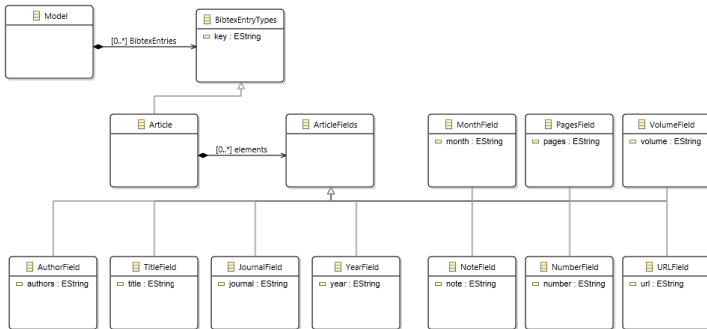


Abbildung 2: BibTeX: Metamodell

Validierung mit Xtext

Validierung der BibTeX-Einträge

- Notwendige Felder **müssen** genau einmal vorhanden sein
- Optionale Felder **dürfen** genau einmal vorhanden sein
- Unbekannte Felder werden ignoriert
- BibTeX-Keys einzigartig

Feldspezifische Validierung (Ausschnitt)

- *AuthorsField*
 - mehrere Autoren durch 'and' getrennt
- *PageField*
 - Positive Zahl größer 0
- ...

Beispiel 'Unknown field warning'

In Klasse *BibTeXValidator*:

```
55 @Check
56 def checkForUnknowns(UnknownField unknown) {
57     warning(
58         'Field "' + unknown.getType().getType() + '" is unknown or not implemented yet. Hence it will be ignored.',
59         BibTeXPackage.Literals.UNKNOWN_FIELD__TYPE, UNKNOWN_FIELD)
60 }
```

Im Editor der Eclipse-Instanz:

```
@article{article01:2015,
  Author="Marcus Stelke and Sven Frank and Frieder Berthold and Alexander Knüppel",
  Title={BibTeX to HTML5: simple as that},
  Journal="MBSE (Software Concepts and Tools)",
  Volume=1,
  ThisFieldIsUnknown = "Unknown",
  URL="http://www.comingsoon.de/",
  Year=2015,
  month=jan,
  UnknownAsWell = {Unknown as well}
}
```


Beispiel 'Unknown field warning'

In Klasse *BibTeXValidator*:

```
55 @Check
56 def checkForUnknowns(UnknownField unknown) {
57     warning(
58         'Field "' + unknown.getType().getType() + '" is unknown or not implemented yet. Hence it will be ignored.',
59         BibTeXPackage.Literals.UNKNOWN_FIELD_TYPE, UNKNOWN_FIELD)
60 }
```

Im Editor der Eclipse-Instanz:

```
@article{article01:2015,
  Author="Marcus Stelke and Sven Frank and Frieder Berthold and Alexander Knüppel",
  Title={BibTeX to HTML5: simple as that},
  Journal="MBSE (Software Concepts and Tools)",
  Volume=1,
  Year=2015,
  month=jan,
  UnknownAsWell = {Unknown as well}
}
```

Field "ThisFieldIsUnknown" is unknown or not implemented yet.
Hence it will be ignored.

Ecore-Metamodell: HTML-Generator

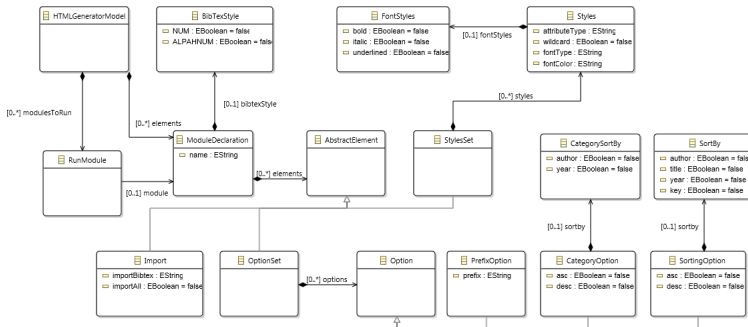


Abbildung 3: HTML Generator: Metamodell

Beispiel einer Generator-Datei

- Unterteilung in Module
- Importieren von *.bib-Dateien
- Diverse Optionen
- Unterschiedliche Gestaltung einzelner Felder

```
1
2 module TestGeneration (num) {
3     import "../bib/library.bib"
4     options = [
5         prefix : "userprefix-"
6         sorting: author desc
7         category: year desc
8     ]
9     styles = [
10        author {
11            font-styles: bold;
12            font-color: red;
13            font-type: "Arial";
14        }
15        title {
16            font-styles: italic;
17        }
18    ]
19 }
20
21 run TestGeneration
22 |
```

Lessons Learned

Allgemein zu Xtext

- Gute Trennung der einzelnen Aufgabenbereiche (Grammatik, Validierung, Generierung)
- Einfach zu benutzen, schwer zu meistern!
- Ressourcen oftmals unbrauchbar

Grammatik

- Erweiterung zu EBNF
- Mehrdeutigkeiten oftmals schwer aufzulösen

Validierung

- Positionierung von Fehlermeldungen schwierig

Xtend

- Mächtige Spracherweiterung von Java



Live-Demo :-)