

Beispiel: Javaklassengenerierung für eine DTD

Sven Eric Panitz

TFH Berlin

Version 5th March 2004

The source of this paper is an XML-file. The sources are processed by the XQuery processor *quip*, XSLT scripts and $\text{\LaTeX}$ in order to produce the different formats of the paper.

Contents

1 Example	1
1.1 Ausgangs-DTD	1
1.2 Konvertierung nach $\text{\LaTeX}$	1

1 Example

1.1 Ausgangs-DTD

```

1  _____ play.dtd _____
2  <!DOCTYPE PLAY SYSTEM "play.dtd" [
3  <!ELEMENT PLAY
4      (TITLE, FM, PERSONAE, SCNDESCR, PLAYSUBT, INDUCT?
5      , PROLOGUE?, ACT+, EPILOGUE?)>
6  <!ELEMENT TITLE      (#PCDATA)>
7  <!ELEMENT FM          (P+)>
8  <!ELEMENT P           (#PCDATA)>
9  <!ELEMENT PERSONAE    (TITLE, (PERSONA | PGROUP)+)>
10 <!ELEMENT PGROUP      (PERSONA+, GRPDESCR)>
11 <!ELEMENT PERSONA      (#PCDATA)>
12 <!ELEMENT GRPDESCR     (#PCDATA)>
13 <!ELEMENT SCNDESCR     (#PCDATA)>
14 <!ELEMENT PLAYSUBT     (#PCDATA)>
15 <!ELEMENT INDUCT
16     (TITLE, SUBTITLE*, (SCENE+ | (SPEECH | STAGEDIR | SUBHEAD)+))>
17 <!ELEMENT ACT
18     (TITLE, SUBTITLE*, PROLOGUE?, SCENE+, EPILOGUE?)>
19 <!ELEMENT SCENE
20     (TITLE, SUBTITLE*, (SPEECH | STAGEDIR | SUBHEAD)+)>
21 <!ELEMENT PROLOGUE     (TITLE, SUBTITLE*, (STAGEDIR | SPEECH)+)>
22 <!ELEMENT EPILOGUE     (TITLE, SUBTITLE*, (STAGEDIR | SPEECH)+)>
23 <!ELEMENT SPEECH       (SPEAKER+, (LINE | STAGEDIR | SUBHEAD)+)>
24 <!ELEMENT SPEAKER      (#PCDATA)>
25 <!ELEMENT LINE         (#PCDATA | STAGEDIR)*>
26 <!ELEMENT STAGEDIR     (#PCDATA)>
27 <!ELEMENT SUBTITLE     (#PCDATA)>
28 <!ELEMENT SUBHEAD      (#PCDATA)>]>

```

1.2 Konvertierung nach $\text{\LaTeX}$

```

1  _____ PLAYToLaTeXVisitor.java _____
2  package examples.theatre;
3
4  import de.tfhberlin.crempel.util.Visitor;
5
6  import java.util.List;

```

```

6 import de.tfhberlin.crempel.util.Maybe;
7 import java.io.*;
8
9 public class PLAYToLaTeXVisitor
10     extends PLAYVisitor<StringWriter>{
11     StringWriter out=new StringWriter();
12
13     public StringWriter eval(PLAY pl){
14         final TITLE title=pl.getTheTITLE();
15         final FM fm=pl.getTheFM();
16         final PERSONAE personae=pl.getThePERSONAE();
17         final SCNDESCR scndesc =pl.getTheSCNDESCR();
18         final PLAYSUBT playsubt=pl.getThePLAYSUBT();
19         final Maybe<INDUCT> induct = pl.getTheMaybe_INDUCT_();
20         final Maybe<PROLOGUE> prologue= pl.getTheMaybe_PROLOGUE_();
21         final List<ACT> acts=pl.getTheList_ACT_();
22         final Maybe<EPILOGUE> epilogue = pl.getTheMaybe_EPILOGUE_();
23
24         out.write("\\documentclass[10pt,a4paper]{article}\\n");
25         out.write("\\usepackage[english]{babel}\\n");
26         out.write("\\begin{document}\\n");
27         out.write("\\title{"+title.getPcdata()+"}\\n");
28         for (ACT act:acts){
29             act.visit(new ACTToLaTeXVisitor(out));
30         }
31         out.write("\\end{document}\\n");
32
33         return out;
34     }
35
36     public static void main(String [] args) throws Exception{
37         final String arg= args[0];
38         PLAY play
39             = (PLAY)de.tfhberlin.crempel.test.MainParser.pFile
40               ("examples.theatre.PLAYParser",arg);
41         StringWriter tex = play.visit(new PLAYToLaTeXVisitor());
42         Writer out
43             = new FileWriter(
44                 arg.substring(0,arg.lastIndexOf('.')+".tex");
45         out.write(tex.toString());
46         out.close();
47     }
48 }

```

ACTToLaTeXVisitor.java

```

1 package examples.theatre;
2
3 import de.tfhberlin.crempel.util.Visitor;
4

```

```

5 import java.util.List;
6 import de.tfhberlin.crempel.util.Maybe;
7 import java.io.*;
8
9 public class ACTToLaTeXVisitor extends ACTVisitor<StringWriter>{
10     StringWriter out;
11     public ACTToLaTeXVisitor(StringWriter o){out=o;}
12     public StringWriter eval(ACT act){
13         final TITLE title = act.getTheTITLE();
14         final List<SUBTITLE> subtitles = act.getList_SUBTITLE();
15         final Maybe<PROLOGUE> prologue=act.getTheMaybe_PROLOGUE();
16         final List<SCENE> scenes=act.getList_SCENE();
17         final Maybe<EPILOGUE> epilogues=act.getTheMaybe_EPILOGUE();
18
19         out.write("\\section*{"+title.getPcdata()+"}\\n");
20         for (SCENE scene:scenes){
21             scene.visit(new SceneToLaTeXVisitor(out));
22         }
23         return out;
24     }}

```

```

SceneToLaTeXVisitor.java
1 package examples.theatre;
2
3 import de.tfhberlin.crempel.util.Visitor;
4
5 import java.util.List;
6 import de.tfhberlin.crempel.util.Maybe;
7 import java.io.*;
8
9 public class SceneToLaTeXVisitor extends SCENEVisitor<StringWriter>{
10     StringWriter out;
11     public SceneToLaTeXVisitor(StringWriter o){out=o;}
12
13     public StringWriter eval(SCENE sc){
14         final TITLE title=sc.getTheTITLE();
15         final List<SUBTITLE> subtitles=sc.getList_SUBTITLE();
16         final List<Choice_SPEECH_STAGEDIR_SUBHEAD> spDirSubheads
17             =sc.getList_Choice_SPEECH_STAGEDIR_SUBHEAD();
18         out.write("\\subsection*{"+title.getPcdata()+"}\\n");
19         for (Choice_SPEECH_STAGEDIR_SUBHEAD spDirSubhead:spDirSubheads)
20             spDirSubhead.visit(new SpeechStageDirSubheadToLaTeX(out));
21         return out;
22     }}

```

```

SpeechStageDirSubheadToLaTeX.java
1 package examples.theatre;
2
3 import de.tfhberlin.crempel.util.Visitor;

```

```

4 import java.io.*;
5 import java.util.List;
6 import de.tfhberlin.crempel.util.Maybe;
7
8 public class SpeechStageDirSubheadToLaTeX
9     extends Choice_SPEECH_STAGEDIR_SUBHEADVisitor<StringWriter>{
10
11     StringWriter out;
12     public SpeechStageDirSubheadToLaTeX(StringWriter o){out=o;}
13
14     public StringWriter eval
15         (CChoice_SPEECH_STAGEDIR_SUBHEADSPEECH sp){
16         SPEECH speech = sp.getTheSPEECH();
17         out.write("\\begin{description}\\n");
18         out.write("\\item[");
19         for (SPEAKER speaker:speech.getList_SPEAKER()){
20             out.write(" "+speaker.getPcdata());
21         }
22         out.write("]");
23         List<Choice_LINE_STAGEDIR_SUBHEAD> lineStagedirSubheads
24             = speech.getList_Choice_LINE_STAGEDIR_SUBHEAD();
25
26         for (Choice_LINE_STAGEDIR_SUBHEAD lineStagedirSubhead
27             :lineStagedirSubheads){
28             lineStagedirSubhead.visit(new LineStageDirSubheadToLaTeX(out));
29         }
30
31
32         out.write("\\end{description}\\n");
33         return out;
34     }
35     public StringWriter eval
36         (CChoice_SPEECH_STAGEDIR_SUBHEADSTAGEDIR st){
37         STAGEDIR stageDir = st.getTheSTAGEDIR(); ;
38         return out;
39     }
40     public StringWriter eval
41         (CChoice_SPEECH_STAGEDIR_SUBHEADSUBHEAD subH){
42         SUBHEAD subhead = subH.getTheSUBHEAD(); ;
43         return out;
44     }
45     public StringWriter eval(Choice_SPEECH_STAGEDIR_SUBHEAD xs){
46         throw new RuntimeException(xs.getClass().toString());
47     }
48 }

```

```

1 package examples.theatre;
2

```

```
3 import de.tfhberlin.crempel.util.Visitor;
4 import java.io.*;
5 import java.util.List;
6 import de.tfhberlin.crempel.util.Maybe;
7
8 public class LineStageDirSubheadToLaTeX
9     extends Choice_LINE_STAGEDIR_SUBHEADVisitor<StringWriter>{
10
11     StringWriter out;
12     public LineStageDirSubheadToLaTeX(StringWriter o){out=o;}
13
14     public StringWriter eval(CChoice_LINE_STAGEDIR_SUBHEADLINE sp){
15         LINE line = sp.getTheLINE();
16         for (Choice_String_STAGEDIR stringStageDir:line.getXs()){
17             if (stringStageDir instanceof CChoice_String_STAGEDIRString){
18                 out.write
19                     (((CChoice_String_STAGEDIRString) stringStageDir)
20                     .getTheString());
21                 out.write("\\\\");
22             }
23         }
24         return out;
25     }
26     public StringWriter eval
27         (CChoice_LINE_STAGEDIR_SUBHEADSTAGEDIR st){
28         STAGEDIR stageDir = st.getTheSTAGEDIR(); ;
29         return out;
30     }
31     public StringWriter eval
32         (CChoice_LINE_STAGEDIR_SUBHEADSUBHEAD subH){
33         SUBHEAD subhead = subH.getTheSUBHEAD(); ;
34         return out;
35     }
36 }
```