Production of Learning Modules in the MiLCA project

The MiLCA Markup Language and Workflow

Table of Contents

GiLES

- Giessen Learning and Education Schema
- XML Markup Language for structuring learning content
  - XML Document Type Definition
  - XML Schema Description
  - XSLT script allows for transforming Learning Objects into XHTML for stand-alone presentation and ILIAS 2.x XHTML
  - XSLT script allows for transforming Learning Objects into XSL-FO (PDF)

GiLES

- Developed at Giessen university as part of the MiLCA project
- Enhanced in cooperation with MiLCA partners
- Base for the XML DTD used in ILIAS 3 for structuring content

GiLES

GiLES is built on top of these open standards:

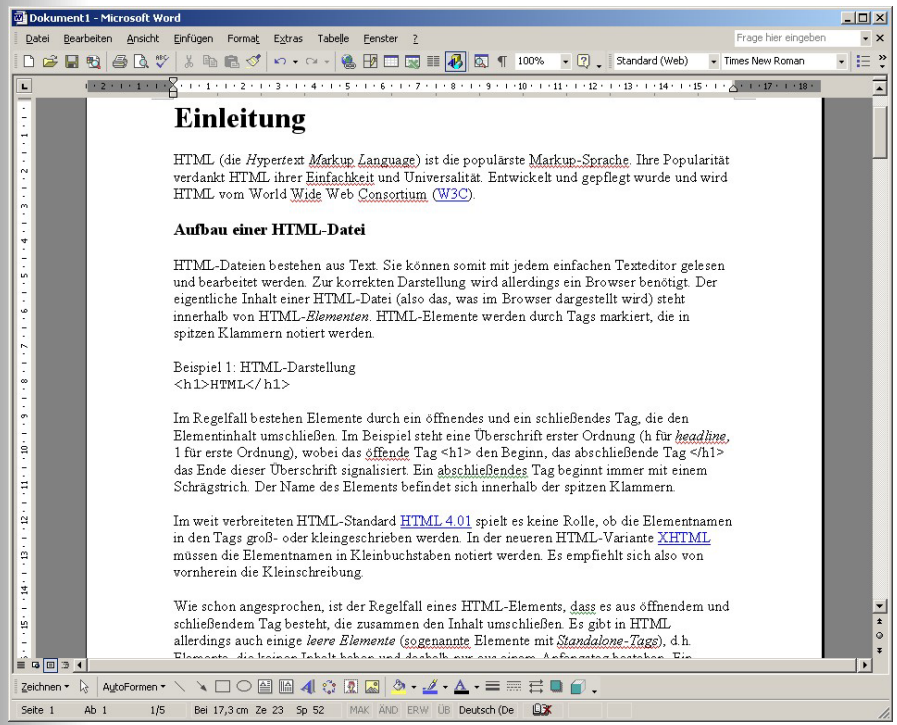
- IEEE Learning Object Metadata (LOM)
- XML
- MathML
- SVG
- [BibTeX]

Structuring content with GiLES

Do we have to structure eLearning content in a special way?

Structuring content with GiLES

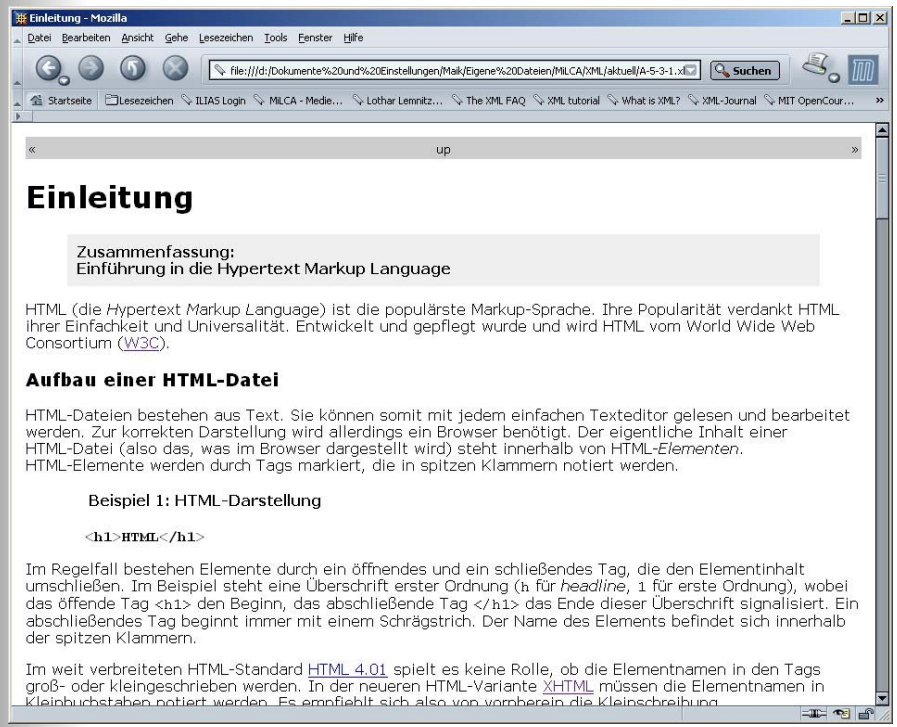
Consider the following:



- \+ Easy to use
- \+ WYSIWYG
- \+ Create content in a reasonable time
- \- Content structure is similar to a book
- \- No metadata
- \- No information about the motivation of different formattings of text
- \- Paper output only!

Structuring content with GiLES

Consider the following:



+ Easy to use (if you use a professional editor)
+ Create content in a reasonable time
+ Linking ability
- No metadata
- HTML was developed to structure web pages – not eLearning content

Structuring content with GiLES

More problems:

- eLearning content has to be enhanced with metadata
  - enables human and machine users to find the content of his/her/its needs
- eLearning content should be designed in a modular way
  - learner is allowed to drop in and out at his/her will
  - use of modular eLearning bricks provides reuse and hence higher sustainability of the content

Structuring content with GiLES

What we need is a markup language capable of

- **structuring eLearning content**

and

- **publishing it in several ways and media**

Structuring content with GiLES

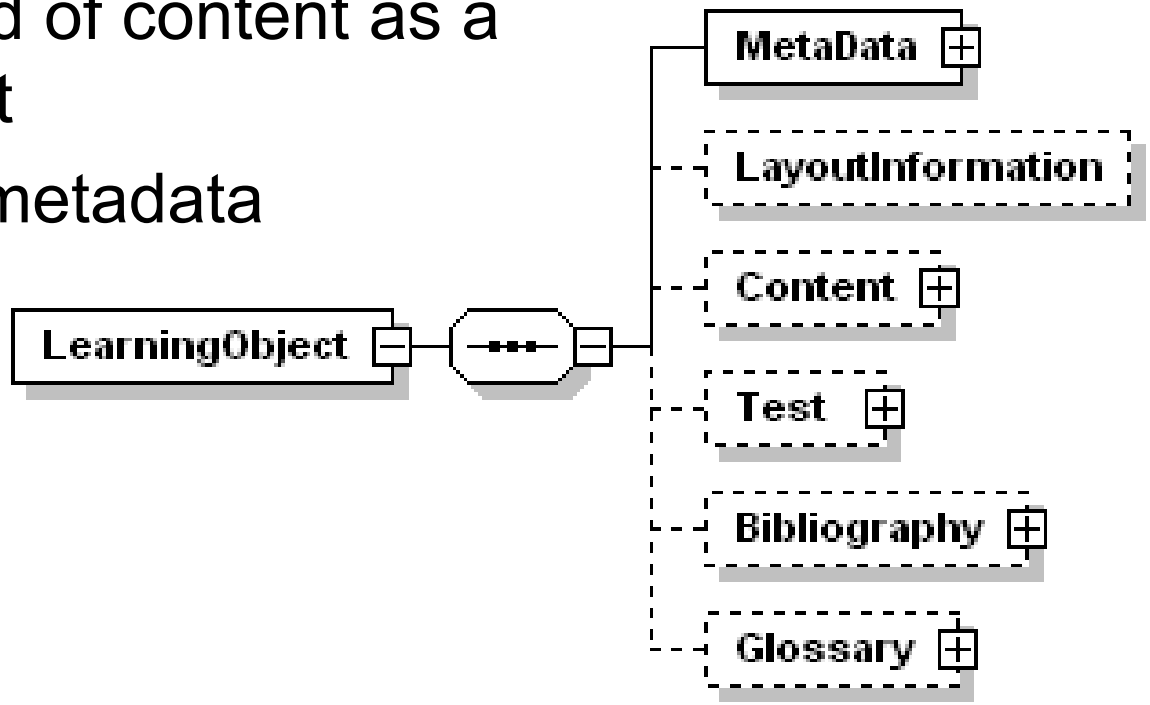
GiLES

```
<?xml version="1.0" encoding="UTF-8"?>
<LearningObject>
 <MetaData>
  <General Structure="Collection" AggregationLevel="2">
   <Identifier Catalog="MiLCA" Entry="A-5-1"/>
   <Title Language="de">Einführung in die Texttechnologie</Title>
   <Description>Diese Lerneinheit führt in die Grundlagen der Texttechnologie ein.
   </Description>
   <Keyword>Texttechnologie</Keyword>
   <Keyword>Markup</Keyword>
 </MetaData>
 <Content>
  <Text>
   <Paragraph Characteristic="Headline">Textttechnologie</Paragraph>
   <Paragraph>…</Paragraph>
  </Text>
  <LearningObject>
  …
  </LearningObject>
 </Content>
</LearningObject>
```

Structuring content with GiLES

GiLES

- defines *any* kind of content as a Learning Object
- provides LOM metadata



Structuring content with GiLES

eLearning ≠ traditional paper text types:

- No chapter, sections, etc.!
- No course, module, lesson!

GiLES supports Learning Objects exclusively!

Structuring content with GiLES

But

Learning Objects have different Aggregation Levels (1 – 4)

and

Learning Objects may contain other Learning Objects!

Structuring content with GiLES

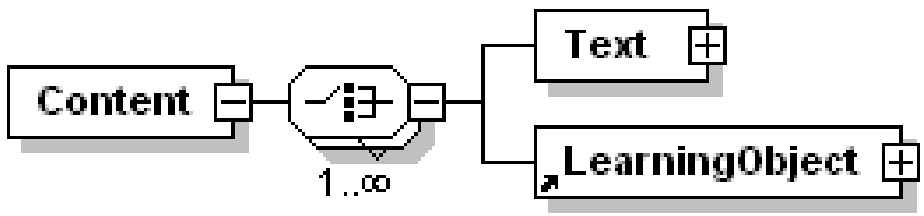
Overview of Aggregation Levels (according to LOM):

- Level 1 Smallest level of aggregation
  Raw media data
- Level 2 Collection of level 1 Learning Objects
  Text
- Level 3 Collection of level 2 Learning Objects
  Course
- Level 4 Largest level
  Set of courses that lead to a certificate

Structuring content with GiLES

Use of recursively nested Learning Objects:

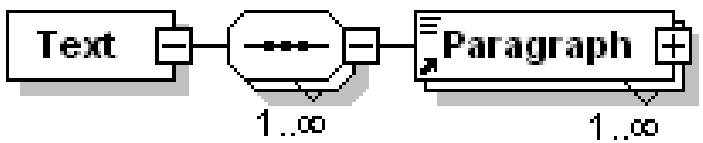
AggregationLevel	**Possible Content**
4 [not used in MiLCA]	Learning Objects with an AL of 3
3	Learning Objects with an AL of 3 or 2
2	Learning Objects with an AL of 1, Text
1	-



Structuring content with GiLES

Use the **`Text`** element for textual information:

- Text is divided into paragraphs
- Paragraphs can include – amongst others – lists, tables, citations and MathML formula
- Inline elements may be used for semantic annotation, including references, comments and glossary entries



Structuring content with GiLES

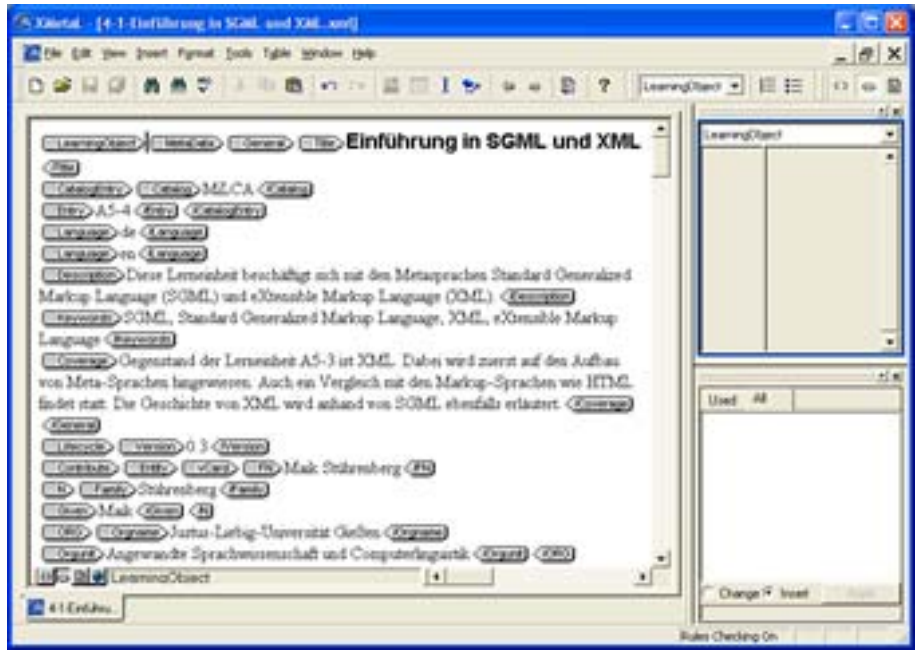
Some real life examples – first step:

- Think about your course content!
  - ➢ What kind of information do you want to convey?
    - Knowledge
    - Skills
    - …
- Think in terms of your course content‘s structure!
  - ➢ Think modular!

Structuring content with GiLES

Second step:

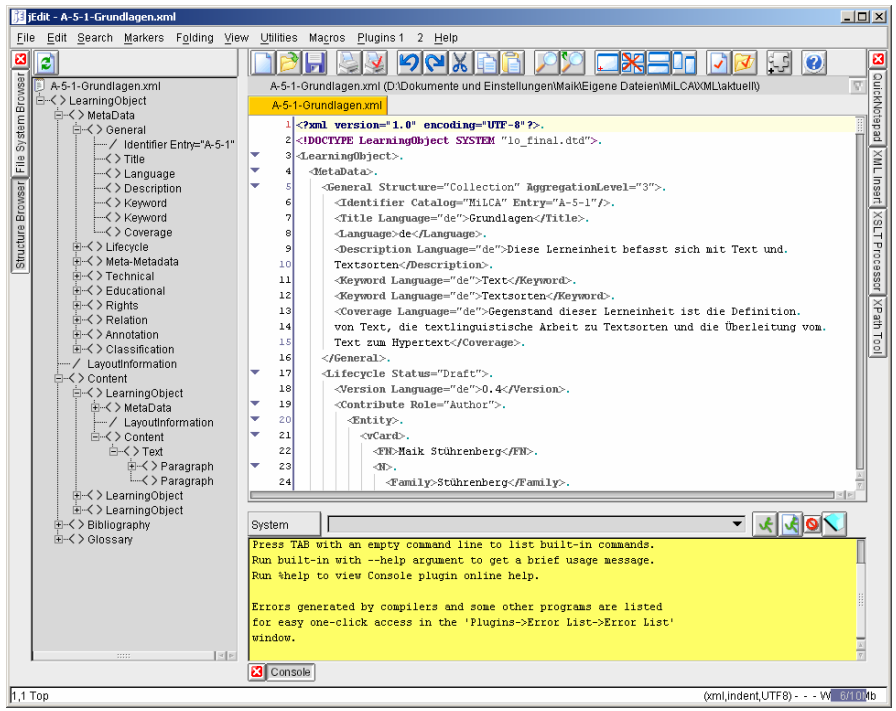
- use an XML editor to annotate your content
  - use a commercial product (like XMetaL)



Structuring content with GiLES

Second step:

- use an XML editor to annotate your content
  - or an Open Source editor (like jEdit)



Structuring content with GiLES

Sad but true:

There is no Word2GiLES converter!

Structuring content with GiLES

But think positive:

Creating XML Learning Objects manually helps you realizing the structural differences between eLearning content and your next paper ;-)

Structuring content with GiLES

Next step:

- Check your Learning Objects with regard to well-formedness and XML validity

➢ use an XML parser of your choice …

  - XP
  - Xerces
  - xmllint
  - …

- … or the parser built into your XML editor

Structuring content with GiLES

Next step:

- Assure correct nesting of Learning Objects with the help of the **`validate.xslt`** stylesheet

➢ Use an XSLT processor of your choice

  - Saxon
  - libxslt
  - Xalan

➢ Experience the full power of XSLT 2.0 functionality by using Saxon 7.x

Structuring content with GiLES

Next step:

- Transform you Learning Objects into the desired output format

➢ Use **`lom2xhtml.xslt`** for XHTML output

  – use the **`ILIAS`** parameter to produce ILIAS 2 compliant output

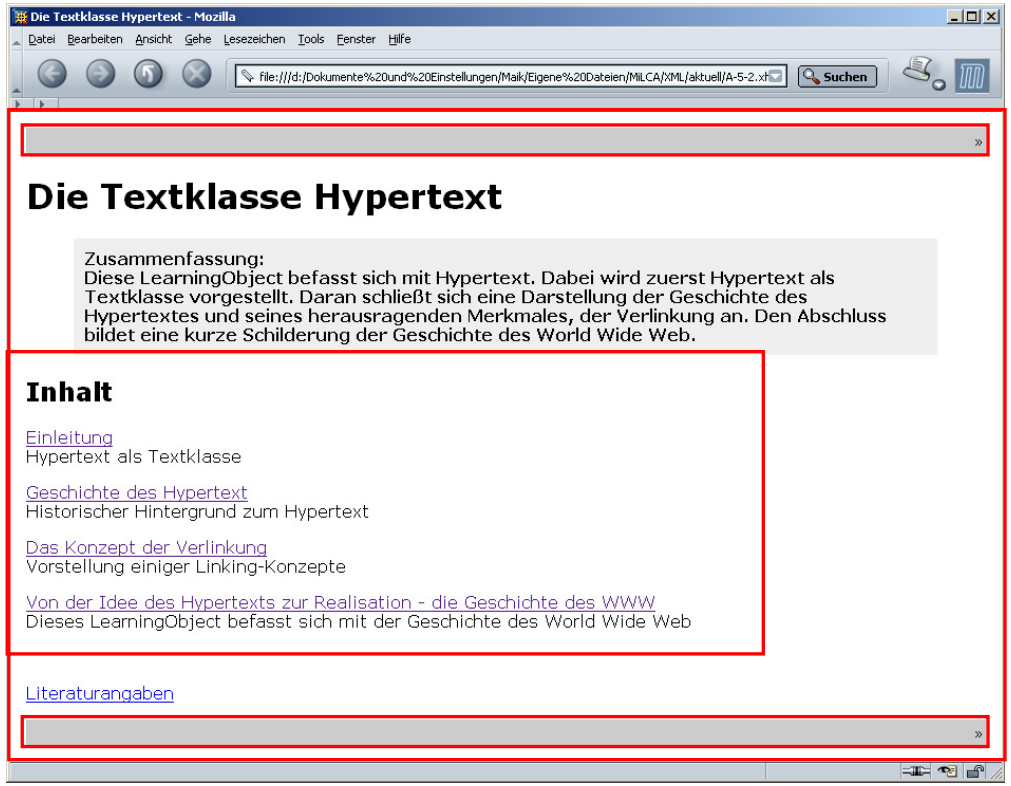
➢ Use **`lom2fo.xslt`** for XSLFO output

  – use an XSLFO renderer (fop, xep, etc.) to produce PDF formatted content

Structuring content with GiLES

lom2xhtml.xslt output:

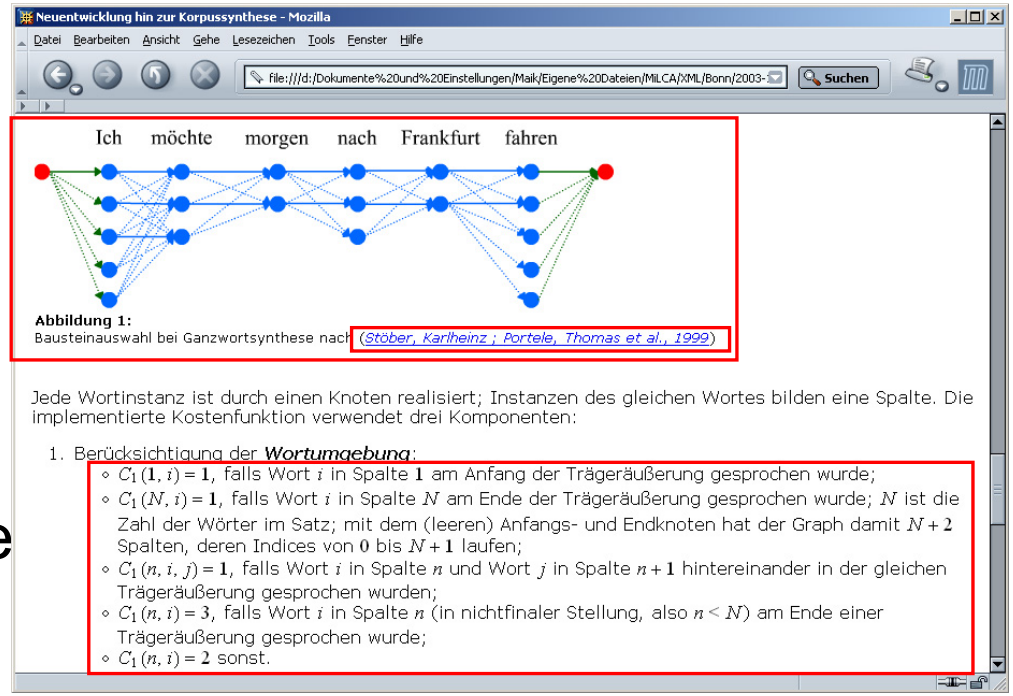
- Automatically generated table of contents
- Automatically inserted navigation elements
- Layout can be modified easily by providing CSS style sheets



Structuring content with GiLES

lom2xhtml.xslt output:

- Use of hyperlinks (e.g., bibliographical references)
- Support of embedded MathML formulas
- Embedded SVG images (if an SVG-aware browse used)
- Flash support (still images and movie files)



Structuring content with GiLES

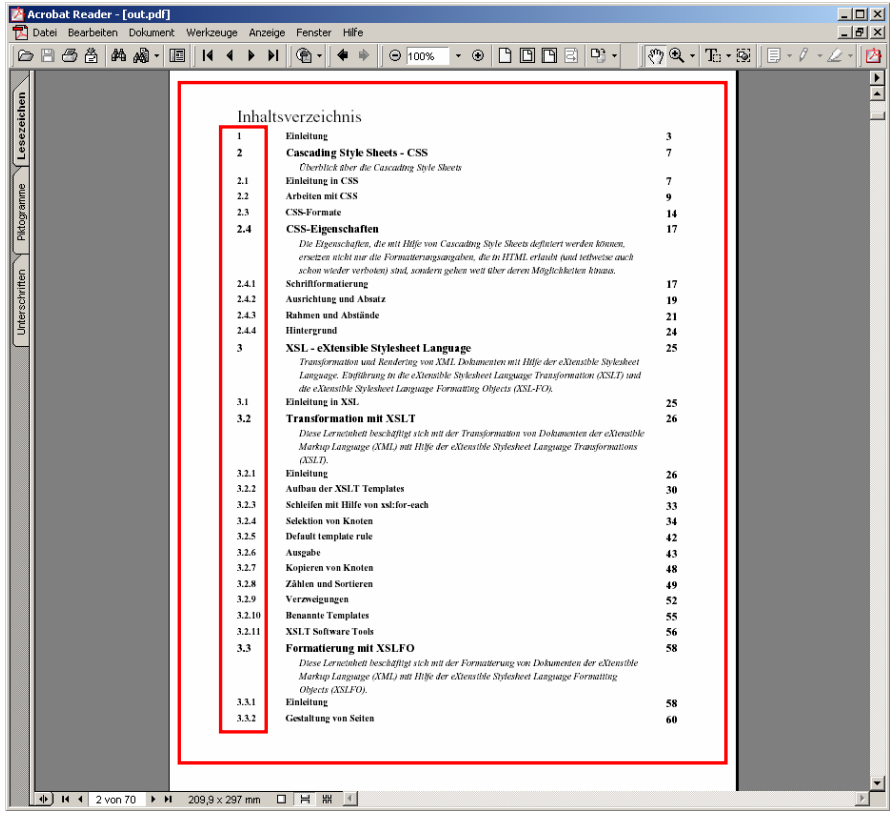
The **`ILIAS`** parameter:

- External hyperlinks only
- Formatted using **`style`** attributes
  - External CSS stylesheets cannot be used
- Comments are inserted (e.g., around media objects) to ease copying and pasting the content into the ILIAS editor

Structuring content with GiLES

lom2fo.xslt output:

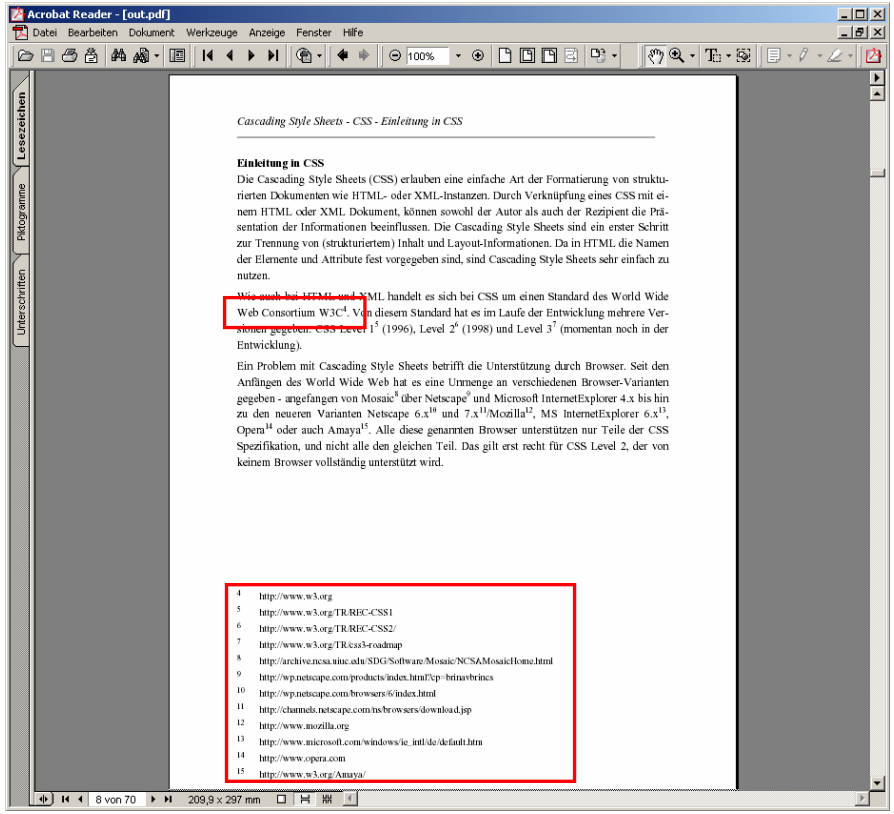
- Automatically generated table of contents (with pagination)
- Automatic numbering of LearningObjects



Structuring content with GiLES

lom2fo.xslt output:

- Different handling of hyperlinks
- Different handling of embedded multimedia objects
  (e.g. SVG images are automatically converted)



Advantages

General benefits when using XML:

- Open, well supported standard
- Variety of software available – commercial, shareware and Open Source
- XML community

Advantages

Benefits when using GiLES:

- Content is structured in a straight-forward and efficient way
- Import, export and exchange of content with MiLCA project partners
- Automatically transform your content into different output formats with no extra costs at all

➢ Real single source publishing

Advantages

Don‘t like the default layout and style?

Create your own transformation stylesheets – it‘s XML!

Problems

Some drawbacks:

- XML and XML editors are not as easy to use as MS Word or other text processing tools
- Neither XML DTDs nor XSDs support some of the structural limitations

➢ hence the need for **`validate.xslt`**

Outlook

The MiLCA project ends in late 2003…

… but GiLES is here to stay!

Outlook

Future work:

- Development of XSLT conversion scripts after release of stable ILIAS 3
  - ILIAS 3 – GiLES and vice versa
- Support of additional output formats
  - Mobile devices
- Bugfixing

Thank you for your attention!

Questions? Comments?

Contact me at

maik.stuehrenberg@uni-giessen.de