



7363 - Web-basierte Anwendungen

Eine Vertiefungsveranstaltung
mit Schwerpunkt auf XML-Technologien



Organisatorisches

Platzvergabe, Teambildung
Regeln zur Teilnahme und Scheinvergabe
Gliederung
Literatur



Gliederung des Kurses

Ergänzungen zur Web-Seite des Kurses



Organisatorisches



- Regeln zur Teilnahme und Scheinvergabe
 - *website* des Kurses
- Platzvergabe, Teambildung
 - *website* des Kurses
- Zeitplan (vorläufig)
 - Excel-Tabelle



- Wiederholung; Dokumentenerstellung mit XML
 - XML 1.1 verabschiedet
 - Die DocBook-DTD und ihre Stylesheets
 - **Meilenstein 1**: Erstellung eines Artikels mit DocBook
 - Inhalt zu Papier bringen: DocBook und XSL-FO
 - Optional: Webseiten mit DocBook & Website-DTD
- Grundlagen Web-basierter Anwendungen
 - Schicht-7-Protokolle: SMTP, HTTP
 - HTML und XHTML (Grundlagen)
 - HTTP-Server, CGI-Programmierung, Formulare
 - **Meilenstein 2**: HTTP-Server aufsetzen (Apache 2.x)



- Web Services-Grundlagen
 - XML-RPC
 - SOAP; SWA (opt.)
 - WSDL; UDDI (opt.)
- **Projektarbeit** (Hauptteil des Praktikums)
- Weitere Theorie-Themen, je nach Bedarf und Zeit:
 - Mehr zu XSL-FO
 - DOM und SAX
 - JavaScript
 - XForms
 - Semantic Web: RDF, DAML
 - ebXML



- Bücher
 - Apache. The Definitive Guide
Ben Laurie & Peter Laurie. 2nd ed. O'Reilly, Sebastopol, CA, 1999.
ISBN 1-56592-528-9
 - Programming Web Services with XML-RPC
Simon St. Laurent, Joe Johnston, Edd Dumbill. 1st ed. O'Reilly, Sebastopol,
CA, 2001. ISBN 0-596-00119-3
 - Programming Web Services with SOAP
James Snell, Dough Tidwell, Pavel Kulchenko. 1st ed. O'Reilly, Sebastopol,
CA, 2002. ISBN 0-596-00095-2
 - Professional XML Web Services
Patrick Cauldwell et al. 1st ed. Wrox Press, Birmingham, UK, 2001.
ISBN 1-861005-09-1
 - HTTP kurz & gut
Clinton Wong. 1. Aufl. O'Reilly, Köln, 2000. ISBN 3-89721-230-7
 - CGI Programmierung mit Perl
Scott Geulich, Shishir Gundavaram, Gunther Birznieks. 2. Aufl. O'Reilly, Köln,
2001. ISBN 3-89721-167-X



- Links
 - Siehe *website* des Kurses:
<http://www.informatik.fh-wiesbaden.de/~werntges/lv/wba01.html>



Grundlagen: Die XML 1.1-Spezifikation



Warum XML 1.1 ?



- Unicode-Grundlage
 - XML 1.0 basierte auf Unicode 2.0 (Juli 1996) und bezog sich damit auf 2^{16} Zeichen. Diese Grundlage hat sich weiterentwickelt: Aktuell gilt Unicode 4.0.1 (März 2004) mit vergebenen Codes bis 0xE01EF und reservierten Codes im Bereich
 - 0x0F0000 - 0x0FFFFFD (Supplementary Private Use Area-A)
 - 0x100000 - 0x10FFFFD (Supplementary Private Use Area-B)(also über 1 Mio. Zeichen statt ca. 65000 !)
Mit weiteren Zeichen ist zu rechnen.
 - Konsequenz: Neue Logik der erlaubten Zeichen, insb. bei *names*
 - XML 1.0: Alle nicht erlaubten Zeichen sind verboten
 - XML 1.1: Alle nicht verbotenen Zeichen sind erlaubt (auch neue!)
- Einfacher zu benutzen
 - Die XML-Spezifikation ist komplett neu geschrieben worden.
 - Damit entfällt das lästige Arbeiten mit Basisdokument, Errata und Zusätzen! Auch viele Formulierungen sind nun noch lesbarer.



Warum XML 1.1 ?



- Zeilenende-Zeichen auf Mainframes
 - Bedarf der Mainframe-Anwender insb. um IBM wird nun berücksichtigt:
 - NEL (0x85) zählt nun zu den Zeilenende-Zeichen (erstaunlicherweise aber nicht zu "S" (*whitespace*))
 - Bei dieser Gelegenheit...
 - Unicode-Zeichen 0x2028 (*line separator char*) ebenfalls
- Control-Codes
 - 0x1 ... 0x1F: In XML 1.1-Dokumenten per *char ref.* (etwa: &0xA;) nun erlaubt, wenn auch nicht "direkt".
 - 0x0 bleibt ausgeschlossen !
 - 0x7F ... 0x9F: **EINSCHRÄNKUNG:**
In XML 1.0 frei verwendbar, in XML 1.1 nur noch per *char. ref.*!



Warum XML 1.1 ?



- Neues Konzept: "Vollständig normierte Dokumente"
 - Autoren sollen diese schaffen, XML-Prozessoren sollen die Einhaltung prüfen
 - **Ziel:**
Tests auf gleiche Namen, Attributwerte und Zeichen(inhalte) sollen einfach durch **Binärvergleich** von Unicode-Strings möglich sein.
 - Einzelheiten:
Siehe Anhang B (*Definitions for Character Normalization*)
 - ziemlich formal; Konsequenzen aus Unicode-Besonderheiten, hier nicht weiter verfolgt. Ähnliche Problematik wie bei Punicode...
- Konkrete Produktionsregeln
 - Siehe Spezifikation bzw. Vorlesungs-Skripte zu LV 7438



DocBook

Eine große Dokumententyp-Definition (DTD)
zur Publikation (informations-)technischer
Dokumente
(einschließlich XSL-Scriptsammlung)



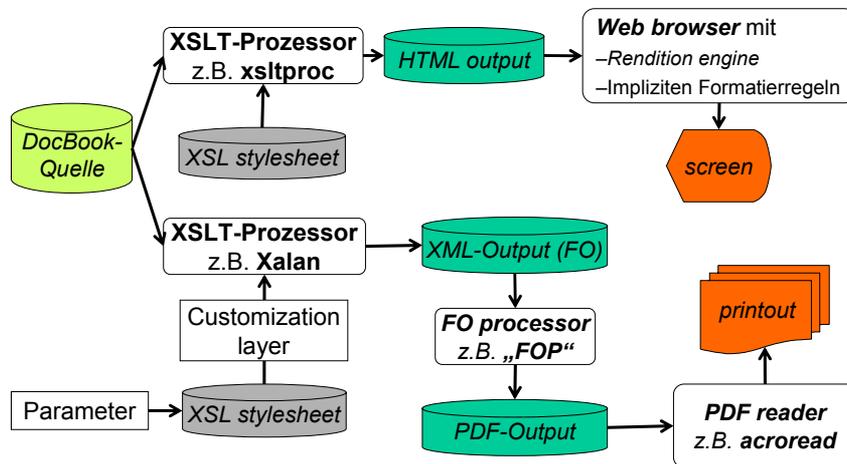
Lernziele



- Umgang mit realistischen, umfangreichen DTDs
 - XML-Kataloge, zahlreiche Entities, Tool-Unterstützung, ...
- Befähigung zur medienunabhängigen, strukturierten Erstellung technischer Dokumente
 - Projektdokumentation (!)
 - Eventuell: BPS-Bericht, Diplomarbeit, ...
- Umgang mit Stylesheet-Paketen (XSLT)
 - Benutzung
 - Parametrisierung
 - Anpassung (*customizing*)
- Fortgeschrittene Publikationstechniken
 - Profiling (redundanzarme Variantenpflege)
- Schließlich:
 - Einstimmung auf den eigenverantwortlichen Arbeitsstil der LV



... nun auf DocBook gemünzt:



- Infrastruktur
 - /etc/xml/catalog, /etc/xml/docbook-xml.xml
 - Funktionsweise erläutern!
 - Erinnerung an XML-Grundkurs
 - /usr/share/xml/docbook vorstellen
 - schema/dtd-Ast, allg. Struktur der DTD
 - stylesheet/nwalsh-Ast
 - html vs. fo vs. ...
 - docbook vs. chunk (!)
 - website-Ast (Ausblick, optional)
 - xsltproc, nsgmls (via Emacs)
 - Emacs' XML-Mode
 - Eventuell Probleme mit catalog
 - Ggf. soft link "ent"



- Herkunft:
 - SGML-Vergangenheit, inzwischen auch in XML brauchbar
 - Ausgereift, lange am Markt: V 4.3 inzwischen stabil
 - Ideal für die Belange der Informatik!
 - Wartung erfolgt durch OASIS (www.oasis-open.org)
- Dokumentation
 - *DocBook: The Definitive Guide*
 - *Style guides*: TLDP, KDE
 - Weitere Links
- Makefile für Docbook
 - Vorstellen, Funktion erläutern



- Wichtige DocBook-Dokumenttypen
 - book: Für Bücher, Diplomarbeiten, etc.
 - article: Zeitschriftenartikel, Veröffentlichungen, Reports, Whitepaper
- Wichtige Gliederungsentscheidung
 - Entweder:
`<sect1> <sect2><sect3>...</sect3>...</sect2>...</sect1>`
 - oder rekursiv:
`<section><section><section>...</section>...</section>...</section>`
- Unterscheide
 - Block-bildende Elemente ("para", "itemizedlist", ...)
 - In-line Elemente ("emphasis", "quote", "acronym", "abbrev", ...)



- Wichtige Struktur-Elemente
 - articleinfo, bookinfo
 - Umfassen allgemeine Angaben über das Dokument wie Autor, Copyright
 - programlisting, table, mediaobject, example
 - selbsterklärend
 - bibliography, glossary, index
 - selbsterklärend
- Wo fange ich an?
 - DocBook-Referenz, Ch. 2, insbesondere: "Writing an article,"
 - Dann: Autorenleitfäden für KDE und TLDP erarbeiten



- Demo
 - Original-Artikel in der c't (Druckversion)
 - HTML-Version der c't
 - Nach HTML gewandelte, eigene XML-Version
 - Ausblick: FO, PDF
- Einsatz der XSL(T)-Scripte von DocBook
 - In der Regel: "docbook.xsl" im jew. Unterverzeichnis
 - Bewirkt 1:1-Ausgabe (ein Ausgabedokument)
 - Ausnahme HTML: Hier auch "chunk.xsl" möglich
 - Bewirkt 1:n-Ausgabe (z.B. eine HTML-Seite pro Kapitel / Sektion) plus Navigationshilfen auf jeder Seite.