XML und TEI

Grundlagen der Textkodierung

XML und TEI

- XML = Stellt nur die grundlegenden Regeln bereit:
  - Baumstruktur (ein Root-Element, korrekte Schachtelung)
  - Konventionen für die Darstellung von
    - Elementen (`<element/>`)
    - Attributen (`<element attribut= " wert" />`)
    - Entity-Referenzen (`&entity;`)
    - Kommentaren (`<!-- … -->`) usw.
  - Benennungsregeln für Elemente und Attribute

XML und TEI

- TEI (Text Encoding Initiative) „ is a consortium which collectively develops and maintains a standard for the representation of texts in digital form”
- D.h. bei Interesse können Personen oder Organisationen Mitglied werden, an der Weiterentwicklung von Standards mitarbeiten, ‚Special Interest Groups‘ organisieren usw.
- Üblicherweise wird „TEI“ aber auch als Synonym für den Standard, den die TEI entwickelt, benutzt
- http://www.tei-c.org/

TEI Guidelines

- TEI-Standard ist eine Einschränkung der im Prinzip unendlichen Möglichkeiten von XML
- wichtigstes ‚Produkt' der TEI sind die ‚Guidelines' (1.605 Druckseiten Text) und formalisierte Schemata zur Validierung von XML-Dateien (bzw. Werkzeuge um diese zu erstellen)
- Fragen, die geklärt werden müssen:
  - Welche Tags und Attribute werden bereitgestellt?
  - Wie dürfen die Tags verschachtelt werden?
- Erste Version der Guidelines wurde 1988 entwickelt (SGML-basiert), derzeit Version P5 (proposal 5) aktuell

TEI und ‚Customisations'

- TEI-Standard stellt mehrere hundert Elemente (Tags) und Attribute bereit, z.B. für
  - ‚Normale' Textkodierung
  - Textkritische Editionen
  - Linguistische Corpora
  - Bibliographische Beschreibung von Handschriften
  - Verknüpfung von Texten mit digitalen Bildern
- In den seltensten Fällen alle benötigt

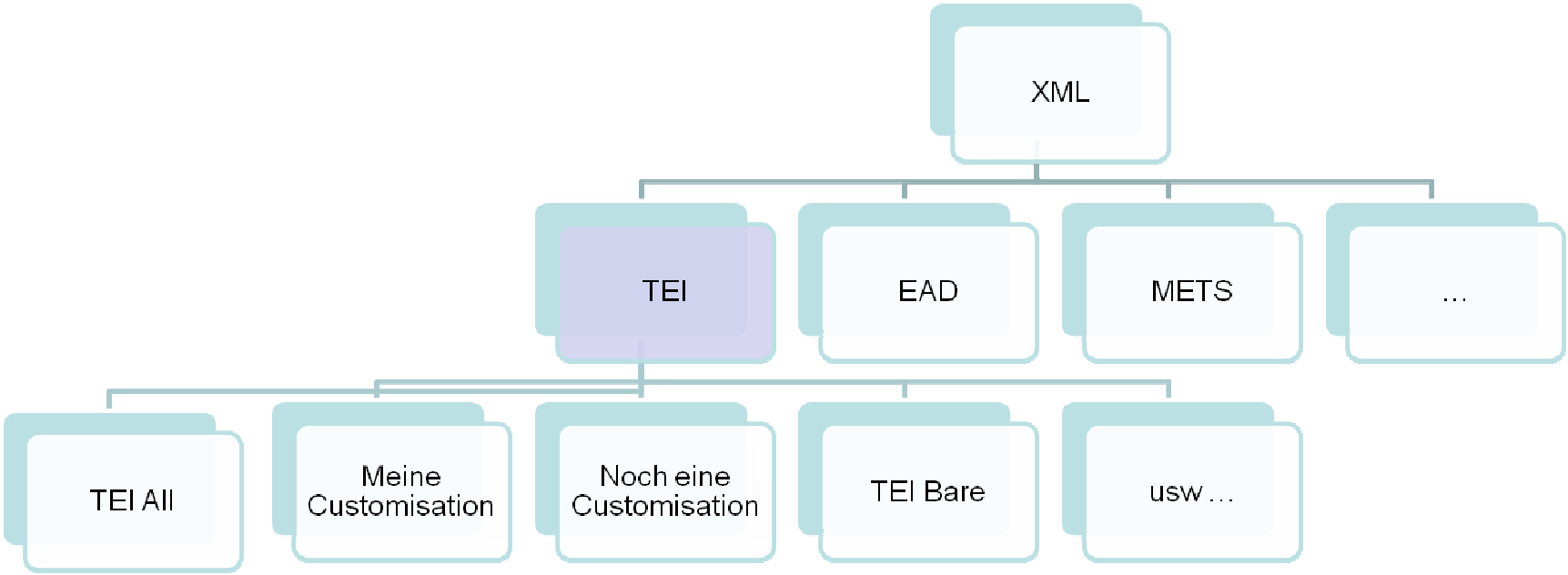
TEI und ‚Customisations'

- Modularer Aufbau der TEI erlaubt Definition von Untermengen des TEI-Tagsets
- d.h. mein Schema muss nicht alle Elemente und Attribute der TEI enthalten (customisations)
- Module u.a.
  - **core (Basiselemente)**
  - **header (Metadaten)**
  - **textstructure (grundlegende Textstrukturen)**
  - drama (Dramen u.ä.)
  - msdescription (Handschriftenbeschreibungen)
  - gaiji (Sonderzeichen)

TEI und ‚Customisations'

- Möglichkeit, angepasste Schemata zu generieren mit dem Tool Roma (http://www.tei-c.org/Roma/)
- Oxygen ermöglicht Einbindung einiger vorgefertigter Schemata, u.a.
  - TEI All (alle Elemente, Maximalschema)
  - TEI Bare (nur das allernotwendigste)
  - TEI Lite (die wichtigsten Elemente)

Übersicht


XML
TEI
EAD
METS
…
TEI All
Meine Customisation
Noch eine Customisation
TEI Bare
usw…

Grundstruktur einer TEI-Datei

- Rootelement `<TEI>`
- Enthält mindestens zwei Unterelemente, nämlich
  - `<teiHeader>` (muss immer vorhanden sein)
  - `<text>` und/oder
  - `<facsimile>` (für Verknüpfung mit Bildern) oder
  - `<fsdDecl>` ‘Feature Structure Declaration‘ (v.a. für Textanalysen, Linguistische Merkmale u.ä.)

Grundstruktur einer TEI-Datei

- Sonderfall `<teiCorpus>` besteht aus
  - `/teiCorpus/teiHeader`
  - 1..n `/teiCorpus/TEI`
  - Vorteil: Trennung von Metadaten, die sich auf das Gesamtkorpus beziehen („Goethes Briefe“), und Metadaten, die sich auf die Teile beziehen („Brief an Eckermann v. 14.8.1830“)
  - Geeignet z.B. für:
    - sprachwissenschaftliche Korpora
    - Sammeleditionen aus mehreren Quellen
    - Briefeditionen
    - Nachlässe

<teiHeader>

TEI Guidelines: 2 The TEI Header

- enthält Metadaten zum TEI-Text:
  - Autor, Titel usw.
  - wann erstellt?
  - Quelle(n), Editionsrichtlinien, Versionsgeschichte
  - …
- 4 Teile des Headers:
  - <fileDesc>: notwendig
  - <encodingDesc>: fakultativ
  - <profileDesc>: fakultativ
  - <revisionDesc>: fakultativ

Header T. 1: <fileDesc>

- Bibliographische Beschreibung des TEI-Dokuments (Autor, Titel, Editor, Projekt, Erstellungsdatum usw.)
- Beschreibung der Quelle(n), z.B. einer Druckausgabe, einer Handschrift, eines Archivguts usw.
- Muss enthalten:
  <titleStmt>, <publicationStmt>, <sourceDesc>
- Kann enthalten:
  <editionStmt>, <extent>, <seriesStmt>, <notesStmt>

Header T. 1: <fileDesc>

- <titleStmt>
  - Angaben zu Autor, Titel usw., bezogen auf die digitale Edition (nicht die edierte Vorlage o.ä.)
  - Beispiel 1
  - Beispiel 2

Header T. 1: <fileDesc>

- <publicationStmt>
  - Publikationsdaten **der elektronischen Ausgabe** (nicht der Vorlage)
  - Beispiel 1
  - Beispiel 2

Header T. 1: <fileDesc>

- <sourceDesc>
  - Beschreibung **der edierten Quelle**
  - Im einfachsten Fall digital erstellt
  - Freie Beschreibung möglich, z.B. Brief von Heine an Nicolai, Berlin SBB-PK, Nachlass Nicolai, Kasten 7
  - Bibliographische Aufnahmen mit: <bibl>, <biblStruct> oder <biblFull>
  - Handschriftenbeschreibungen mit <msDesc>

Header T. 1: <fileDesc>

- <bibl>, <biblStruct>, <biblFull> nicht allein in der <sourceDesc>, sondern auch im Dokument selbst (z.B. bei Fußnoten, Bibliographien o.ä.) möglich
  - <bibl> lässt unstrukturierte Beschreibung zu
  - <biblStruct> geeignet für Beschreibung gedruckter Vorlagen
  - <biblFull> ursprünglich entwickelt für Beschreibung von digitalen Ressourcen, bei Beschreibung von Druckvorlagen häufig problematisch
  - <msDesc> speziell für mittelalterliche Handschriften
  - Beispiele

Header T. 2: <encodingDesc>

- Beschreibung der editorischen Praxis, u.a.
  - Projektbeschreibung: <projectDesc>
  - Editionsrichtlinien (Normalisierung u.ä.): <editorialDecl>
  - Für Korpora: <samplingDecl>
  - Beschreibung der Verwendung von Tags und ggf. Verknüpfung mit bestimmter Darstellungsweise: <tagsDecl>
  - Evtl. Definition eigener Sonderzeichen
  - …
  - Beispiel

Header T. 3: <profileDesc>

- Kodierung inhaltlicher Informationen über den Text
  - Entstehungszeit
  - Sprache
  - Textsorte
  - v.a. für Sprachcorpora von Interesse, bei 'normalen' Editionen eher selten verwendet
  - Beispiel

Header T. 4: <revisionDesc>

- Informationen über die 'Geschichte' des Dokuments
  - Wann erstellt?
  - Wann wurden welche Veränderungen vorgenommen?
  - evtl. Begründung der Veränderungen usw.
  - v.a. sinnvoll bei großen Editionsprojekten mit mehreren MitarbeiterInnen
  - Beispiel

Übung zu `<teiHeader>`

- Erstellen Sie in Oxygen ein TEI-Dokument mit der Vorlage "TEI All"
- Erstellen Sie ein `<titleStmt>` und `<publicationStmt>` für Ihr eigenes Projekt unter Angabe von Autor, Titel, Herausgeber
- Setzen Sie entweder eine Quellenangabe für Ihr eigenes Projekt oder die folgende Angabe mit `<biblStruct>` um:
- J.W.v.Goethe: Die Leiden des jungen Werthers. Erste Fassung. In: Goethes poetische Werke. Hg. von Liselotte Lohrer. Stuttgart: Cotta 1950. Bd 6, S. 7-130.
- Erstellen Sie eine rudimentäre `<revisionHistory>`

<text>

- Das <text>-Element enthält den eigentlichen Text
  - Enthält i.d.R. ein <body>-Element
  - dazu fakultativ <front> und/oder <back>
  - Oder <group>
- Sonderfall <group>: enthält 1..n <text>-Elemente
- Unterschied zw. <teiCorpus> und <group>: bei <teiCorpus> hat jeder Text einen eigenen Header, bei <group> nicht
- Die Entscheidung zw. <teiCorpus> und <group> hängt v. Editionsaufbau ab

<text>

- Beispiel mit <text>
- Beispiel mit <group>

Wichtige Gliederungselemente

(TEI-Guidelines: 4 Default Text Structure)

- `<div>` (Division): Abschnitte im Dokument (z.B. Buch, Kapitel, einzelne Gedichte, Akte/Szenen u.ä.)
- Wichtige Attribute:
  - `@n` (Nummerierung, z.B. "1.1.2.a", entweder aus der Quelle übernommen oder selbst erstellt)
  - `@type` (z.B. "book", "chapter", "poem")
  - `@xml:id` (eindeutiger Identifikator, muss dokumentenweit eindeutig sein, und mit einem Buchstaben beginnen, i.d.R. selbst vergeben oder automatisch erzeugt)

Wichtige Gliederungselemente

- `<div n="x">` vs. `<div1>` bis `<div7>`
- Alternativ zu `<div>`, bei dem die Hierarchiestufe durch die Schachtelung innerhalb des XML-Dokuments definiert ist, können die Hierarchiestufen auch explizit gemacht werden mit `<div1>`, `<div2>` usw.
- Entscheidung durch den Editor
- `<div>` ist flexibler
- `<div1>` usw. z.T. leichter in der nachfolgenden Verarbeitung
- Mischung nicht möglich

Wichtige Gliederungselemente

- Beispiel mit `<div n="x">`
- Beispiel mit `<div1>` bis `<div7>`

<front> und <back>

- Spezialelemente für Vorstücke (Titelblatt, Vorwort, Inhaltsverzeichnis u.ä.) und Nachstücke (Register, Nachwort usw.)
- V.a. bei der Umsetzung gedruckter Vorlagen wichtig
- Enthaltene Elemente können z.B. sein:
  - <titlePage>, <docImprint>, <byline>, usw.
- Im Prinzip alle Elemente verfügbar, die auch in <body> verfügbar sind. Eher eine vom Herausgeber bestimmte Gliederung
- Beispiel

Grundlegende Tags

(TEI-Guidelines: 3 Elements Available in All TEI Documents)

- `<p>` (paragraph): Absatz
- `<ab>` (anonymous block): Irgendein Textblock
- `<head>` (headline): Überschrift
- `<lb/>` (linebreak): Zeilenumbruch (z.B. wenn Zeilenumbrüche der Vorlage mit transkribiert werden)
- `<pb/>` (pagebreak): Seitenumbruch, normalerweise der der Vorlage
- Mit `@n, @type, @xml:id` spezifizierbar
- Beispiel

Hervorhebung und wörtliche Rede

- `<hi>` (highlighted): allgemeiner Tag für Hervorhebungen, z.B. Kursiv o.ä., spezifizierbar durch `@rend`
- `<foreign>, <emph>, <distinct>`: Verschiedene Hervorhebungsarten bzw. Markierung ‚ungewöhnlicher' Textteile (Fremdsprachiges, Slang, Archaismen)
- `<q>` für wörtliche Rede (in Anführungsstrichen)
- `<quote>` und `<cit>` für Zitate
- Beispiel

Sonderfälle: Gedichte und Dramen

- `<lg>` und `<l>` für Gedichte, Versdramen, gebundene Sprache
- `<l>` bezeichnet die metrische Zeile, `<lb/>` markiert den gedruckten oder handschriftlichen Zeilenumbruch
- Für Dramen stellt die TEI ein umfangreiches Vokabular zur Auszeichnung von Sprechern, Sprechtexte, Bühnenanweisungen usw. zur Verfügung, die wichtigsten sind:
- `<sp>, <speaker>, <stage>`
- Beispiele

Weitere wichtige Tags

- `<list>, <item>, <label>` für Listen
- `<listBibl>` für Literaturlisten
- `<note>` für Anmerkungen (z.B. Fußnoten, Marginalien)
- `<figure>` und `<graphic>` für Illustrationen u.ä.

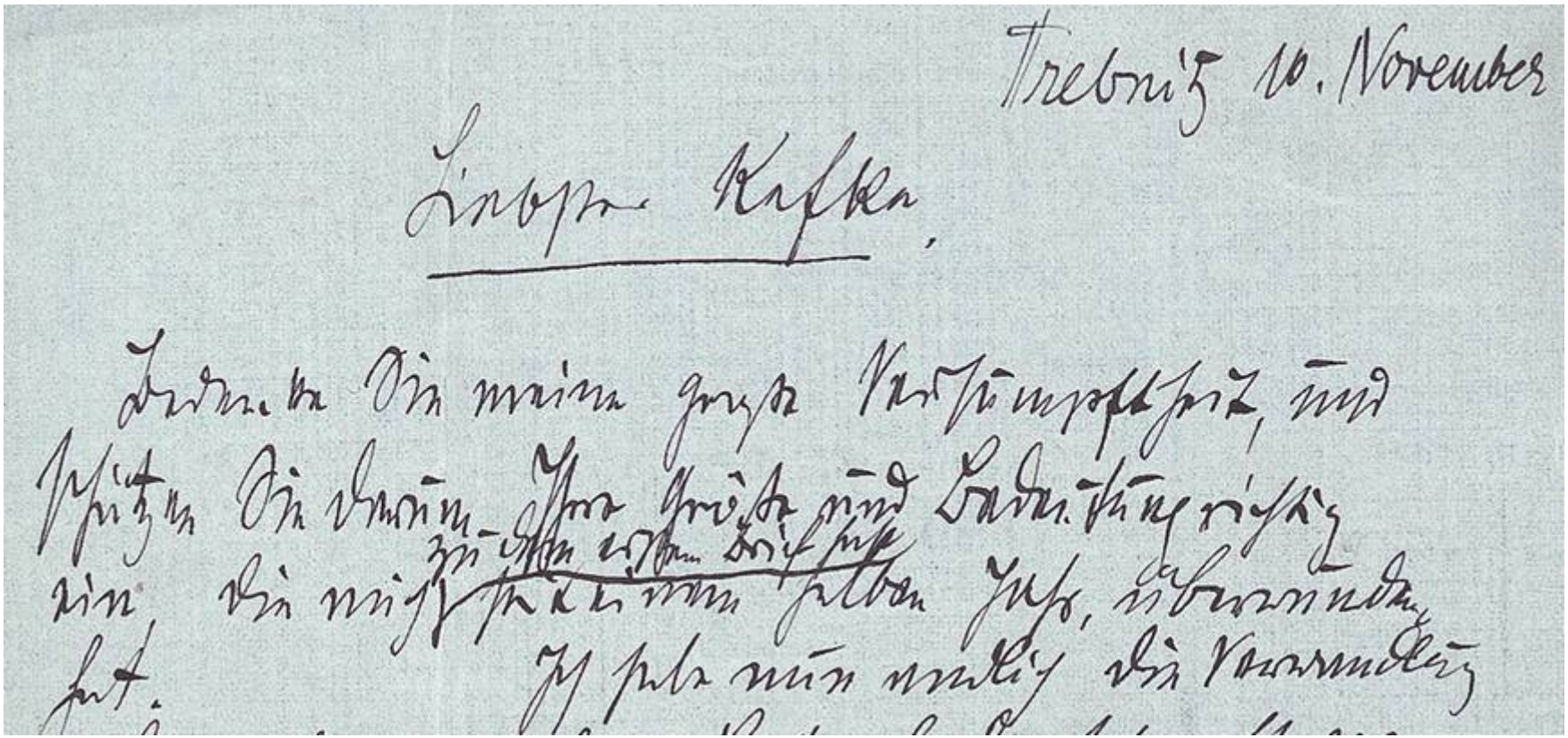
Übung zu `<text>`

- Erstellen Sie eine Textgliederung in `<front>`, `<body>` und `<back>`
- Fügen Sie eine fiktive Titelseite ein
- Transkribieren Sie auszugsweise den Anfang der Wahlverwandschaften
- Kodieren Sie Zeilenumbrüche
- Zeichnen Sie ein beliebiges Stück mit `<hi>` aus
- Kodieren Sie ein Stück wörtliche Rede
- Transformieren Sie das ganze in HTML mit den mitgelieferten XSLT-Skripten

Spezialfall Briefe

- Briefe können als eigenes Dokument, als Texte innerhalb einer `<group>` oder als `<div>`-Elemente ediert werden
- Ggf. mit `<teiCorpus>` gruppieren
- `<opener>` und `<closer>` als Container-Elemente, z.B. für
  - `<dateline>`: Ort und Datum
  - `<byline>`: Verfasserangabe
  - `<salute>`: Gruß
  - `<signed>`: Unterschrift

Beispiel



Transkription vs. TEI

Brief von Franz Werfel an Franz Kafka

Trebnitz 10. November

Liebster Kafka.

Bedenken Sie meine große Versumpftheit, und schätzen Sie darum Ihre Größe und Bedeutung richtig ein, die mich (zu dem ersten Brief heute) seit einem halben Jahr, überwunden hat.

TEI-Kodierung

Behandlung von Sonderzeichen

- Bei Transkriptionen älterer und/oder handschriftlicher Texte häufig Sonderzeichen
- Inzwischen zahlreiche Sonderzeichen im Unicode-Standard definiert
- Außerdem bietet die TEI im gaiji-Modul Elemente an, durch die Sonderzeichen definiert, beschrieben und in der Transkription eingesetzt werden können

Was ist Unicode?

- „Internationaler Standard, in dem langfristig für jedes sinntragende Schriftzeichen oder Textelement aller bekannten Schriftkulturen und Zeichensysteme ein digitaler Code festgelegt wird“ (http://de.wikipedia.org/wiki/Unicode)
- Bzw. festgelegt werden soll. (OD)

Warum Unicode?

- Ältere Zeichencodierungen konnten lediglich 128 (ASCII, 7 bit) oder 256 (z.B. ISO-8859, 8 bit) Zeichen codieren
- Folge: für unterschiedliche Schriftsysteme mussten verschiedene Zeichencodierungen entwickelt werden und ggf. angegeben werden, in welcher Zeichencodierung eine Datei gespeichert ist (z.B. ISO-8859-1, ISO-8859-5 usw.)

Warum Unicode?

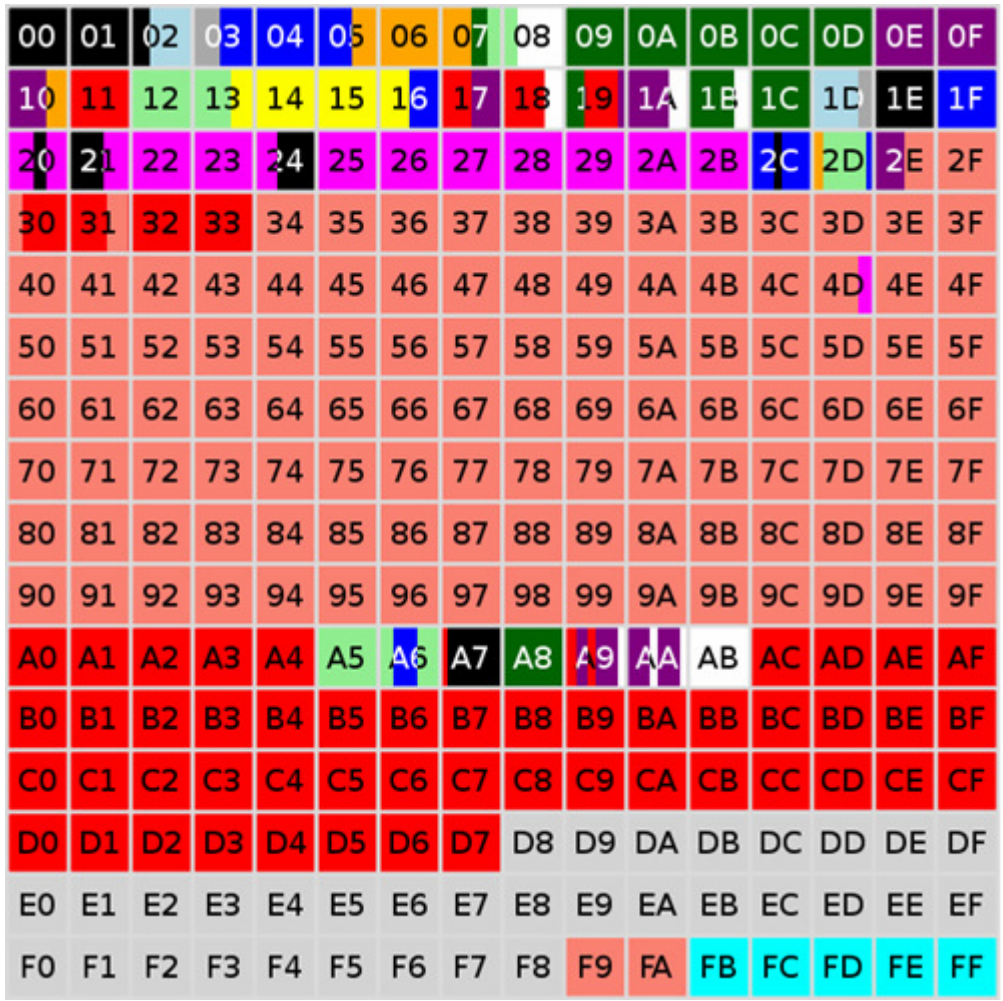
- Unicode soll die verschiedenen miteinander inkompatiblen Zeichenkodierungen ersetzen
- In Unicode 1.0 sollten alle Schriftzeichen der Welt durch 65.536 (2^{16}) sog. „codepoints“ repräsentiert werden
- Inzwischen erweitert auf 17 Bereiche („planes“) von je 65.536 codepoints -> 1.114.112 mögliche Zeichen

Warum Unicode?

- Unicode-Standard wird ständig durch das „Unicode Consortium“ kontinuierlich weiterentwickelt
- Aktuelle Version ist Unicode 6.0.0 (Jan. 2011)
- Lateinisch, Griechisch, Kyrillisch, Arabisch, Hebräisch, CJK
- Aber auch so Schriften wie Balinesisch, Gotisch, Glagolitisch, Ogham, Linear B usw.
- Mehrere „Private Use Areas“ (PUA)
- Ergänzungswünsche können (und sollten) dem Unicode Consortium gemeldet werden

Was gibt es in Unicode?

- „Normale“ Schriftzeichen: a b c δ Д א ى ת
- Satzzeichen „ “ ? ! ,
- Whitespace
- Combining Diacritical Marks: ◌̈ ◌̋ ◌̎
- Vorkombinierte Zeichen á ä ť ů ﭙ ḗ ḝ
- Symbole ☭ ♘ ♂ ☿ ∑
- Steuerzeichen Wagenrücklauf, EOF
- …



Quelle: http://de.wikipedia.org/wiki/Datei:Roadmap_to_Unicode_BMP_de.svg

Wie finde ich das Zeichen, das ich brauche?

- Codecharts unter www.unicode.org
- Datenbank unter www.decodeunicode.org
- Oder www.isthisthingon.org/unicode/index.php (The UniSearcher)

Kodierung von Unicode in XML-Dateien

- Entweder Zeichen direkt einfügen, z.B. mit Oxygen:
  - α (intuitiv lesbar, wird aber – je nach Zeichensatz – nicht angezeigt)
- Oder mit Zeichenentitäten:
  - Hexadezimal: `&#x0364;` (gut, entspricht dem Codepoint)
  - Dezimal: `&#945;` (bitte nicht!)

Kombinierende diakritische Zeichen

- Z.B. übergestelltes ᵘ (codepoint U+0367)
- oͧ -> o◌ͧ vs. o□
- Generelles Problem:
  - Ungewöhnliche Zeichen werden nur mit entsprechenden Zeichensätzen und entsprechender Software ordentlich angezeigt
  - Empfehlenswerte Schriften u.a. Arial Unicode MS, Junicode, Code2000, Schriften des GW
  - Weniger empfehlenswert: Mediaevum

Was tun, wenn Unicode nicht weiterhilft?

- Möglichkeit, die Private Use Areas zu verwenden (U+E000-F8FF, Planes 16 u. 17)
- TEI bietet mit den Elementen <char>, <glyph> und <g> eine Methode zur Definition von Sonderzeichen an
- Character -> ein bestimmter „Buchstabe“ (z.B. ein A)
- Glyph -> eine bestimmte Ausführung eines Buchstabens („langes s“, „rundes r“)

Das Element <charDecl>

- Teil von teiHeader/encodingDesc
- Enthält `<char>`- und `<glyph>`-Elemente
- Darin u.a.:
  - `<charName>` bzw. `<glyphName>`
  - `<charProp>`
  - `<desc>`
  - `<mapping>`
  - `<figure>`

Ein Beispiel


Incipit tractatus de efficacia aque bene
dicte, per venerandū magistrū Johannē de turre cremata, sacre
theologie pfessorem, ordinis predicatorz, tpe concilij Basilienſis
cōpilatus, cōtra Petrū anglicū hereticorz defensorē in bohemia

Beispiel: 

```
<encodingDesc>
  <charDecl>
    <char xml:id="pstroke">
      <charName>LATIN SMALL LETTER P WITH STROKE</charName>
      <desc>unten durchgestrichenes p, meist als Abbreviatur für per</desc>
      <charProp>
        <unicodeName>general-category</unicodeName>
        <value>Ll</value>
      </charProp>
      <mapping type="standardized">p</mapping>
      <figure>
        <graphic url="min_per01-01.jpg"/>
      </figure>
      <note>ganz häufig verwendet</note>
    </char>
  </charDecl>
</encodingDesc>
```

```
<TEI>
…
<text<
  <body>
    <p>t<g ref="#pstroke">empor</g>e concilij
  Baſilienſis</p>
  </body>
</text>
</TEI>
```

Beispiel

Pause