

# The TEI Overview

Feb 2006

# Basic concepts

- The TEI is a *modular* system: you use it to build an encoding scheme *appropriate to your needs*, by selecting specific modules
- Each module defines a group of elements and attributes
- Elements are classified structurally and semantically
  - semantic classes group elements which have similar meanings — elements like names, or like editorial interventions for example
  - structural classes group elements which behave similarly in the structure — elements like paragraphs, or like phrases for example
  - we also talk of attribute classes: these group elements which all have the same attributes

**Define your goals *before* using the TEI!**

# Mandatory (ish) modules

- infrastructural

**tei** defines common element classes and macros

**textstructure** defines basic “book-like” structure for prose, verse, drama

- Core

**header** TEI metadata structure

**core** a wide range of elements “common to all kinds of text”

# Optional modules

- Alternative structures
  - transcribed speech
  - dictionaries and lexica
- Specialist applications
  - linking and alignment; non-standard characters and glyphs
  - abstract analysis; feature structures; certainty;
  - physical transcription; textual criticism; names and dates;
  - language corpora; manuscript description;
  - ... and not forgetting the ODD system

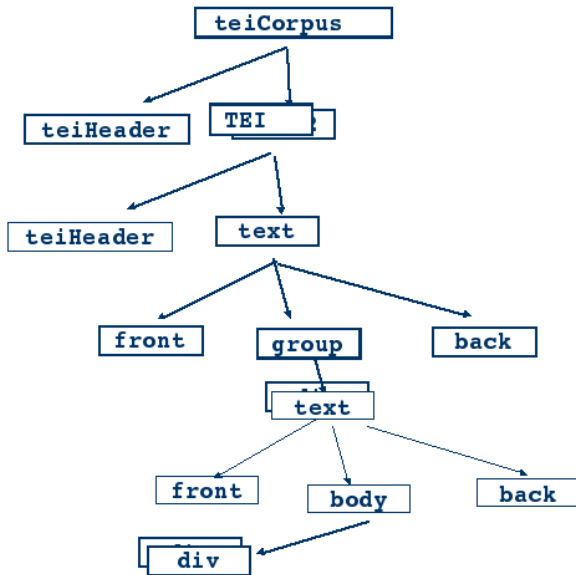
# Basic structure(s)

- Every TEI-conformant document comprises a *header* followed by (at least one) *text*
- the header contains:
  - mandatory file description
  - optional encoding, profile and revision descriptions
- the header is essential for:
  - bibliographic control and identification
  - resource documentation and processing

# Structure of a TEI text

- A text may be unitary or composite
- a **unitary** text contains
  - optional front matter
  - optional back matter
  - a body
- in a **composite** text, the body is replaced by a group of texts (or nested groups)
- A **corpus** is a collection of text and header pairs. It has its own header.

# TEI basic structure



# A text usually has divisions

- generic, hierarchic subdivisions, each incomplete
- the type attribute is used to label a particular level e.g. as "part" or "chapter"
- the n attribute gives a particular division a name or number
- the xml:id attribute gives a particular division a unique identifier
- associated `<head>` and `<trailer>` elements (from the divtop class) may also be supplied
- A `<divGen>` element can be used for 'generated' divisions



# For example...

```
<text>
<front> <!-- titlepage, etc here --> </front>
<body>
  <div type='book' n='I' xml:id='JA0100'>
    <head>Book I.</head>
    <div type='chapter' n='1' xml:id='JA0101'>
      <head>Of writing lives in general...</head>
      <!-- remainder of chapter 1 here -->
    </div>
    <div n='2' xml:id='JA0102'>
      <!-- chapter 2 here -->
    </div>
    <!-- remainder of book 1 here -->
  </div>
  <div type='book' n='II' xml:id='JA0200'>
    <!-- book 2 here -->
  </div>
  <!-- remaining books here -->
</body></text>
```

**NB. divisions always tessellate.**

# TEI global attributes

- The attribute class `att.global` defines these for all elements:
  - `xml:id` supplies a unique identifier
  - `n` supplies a (non-unique) name or number
  - `rend` gives a suggestion about rendition (appearance)
  - `xml:lang` identifies the language using an ISO standard code
- The linking module extends this class with:
  - `corresp`, `synch`, `ana` for specific association types
  - `next`, `prev` for aggregating fragmented elements

# Text components

What are divisions composed of?

- prose is mostly paragraphs (`<p>`)
- verse is mostly lines (`<l>`), sometimes in hierarchic groups (`<lgr>`)
- drama is mostly speeches (`<sp>`) containing `<p>` or `<l>` elements interspersed with stage directions (`<stage>`)

These may be mixed, and may also appear directly within undivided texts.

... but divisions can also contain embedded `<text>` or `<quote>` elements.

# For example

```
<div type="book">  
  <l>Of Man's first disobedience, and the fruit</l>  
  <l>Of that forbidden tree whose mortal taste</l>  
  <l>Brought death into the World, and all our woe,</l>  
  <l>With loss of Eden...</l>  
  ....  
</div>
```

```
<lg type="haiku">  
  <l n="1">Summer grass --</l>  
  <l n="2">all that's left</l>  
  <l n="3">of warriors' dreams</l>  
</lg>
```

# For example

```
<stage>Enter Barnardo and Francisco,  
two Sentinels, at several doors</stage>  
<sp who="Barnardo">  
  <l part="f">Who's there? </l></sp>  
<sp who="Francisco">  
  <l>Nay, answer me. Stand and unfold yourself.</l>  
</sp>  
<sp who="Barnardo">  
  <l part="i">Long live the king! </l></sp>  
<sp who="Francisco">  
  <l part="m">Barnardo? </l></sp>  
<sp who="Barnardo">  
  <l part="f">He. </l></sp>
```

# not to mention

```
<p>.... And he wrote on one side  
of the paper:  
<quote><p>HELP!</p>  
<p>PIGLIT (ME)</p>  
</quote>  
and on the other side:  
<quote>IT'S ME PIGLIT, HELP HELP</quote>  
</p><p>Then he put the paper in the bottle...  
</p>
```

# What are speeches, paragraphs, and lines made of?

- phrases that are conventionally typographically distinct
- “data-like” (names, numbers, dates, times, addresses)
- editorial interventions (corrections, regularizations, additions, omissions ...)
- cross references and links
- lists, notes, graphics, tables, bibliographic citations...
- all kinds of annotations!

**Which of these you need to markup will depend on your research agenda**

# for example...

```
<head>
Of writing lives in general, and particularly of
<title>Pamela </title>, with a word by the bye of
<name key="#CIBC03">Colley Cibber</name> and others.</head>
<p>It is a trite but true observation, that
<q>examples work more forcibly on the mind
than precepts</q></p>
<p><name key="#JA">Mr. Joseph Andrews</name>,
<rs>the hero of our ensuing history</rs>, was
esteemed to be ...
```



# Direct speech

- Use the who attribute to show speakers
- Speeches can be nested in other speeches

```
<q who="Wilson"> Spaulding, he came down into  
the office just this day eight weeks with  
this very paper in his hand, and he  
says: <q who="Spaulding">I wish to  
the Lord, Mr. Wilson, that I was a  
red-headed man.</q></q>
```

# Foreign language phrases

- The `xml:lang` attribute may be attached to any element
- Use `<foreign>` if nothing else is available
- Use ISO 639-2 code to identify language

```
Have you read  
<title xml:lang="deu">Die Dreigroschenoper</title>?
```

```
<mentioned xml:lang="fra">  
Savoir-faire </mentioned>  
  is French for know-how.
```

```
John has real <foreign xml:lang="fra">  
savoir-faire </foreign>.
```

# Is there life beyond Unicode ?

The gaiji module allows you to define and document non-Unicode characters or glyphs

```
<charDesc>
  <glyph xml:id="r1">
    <glyphName>LATIN R WITH ONE FUNNY STROKE</glyphName>
    <graphic url="r1img.png"/>
  </glyph>
  <glyph xml:id="r2">
    <glyphName>LATIN R WITH TWO FUNNY STROKES</glyphName>
    <charProp>
      <localName>entity</localName>
      <value>R2</value>
    </charProp>
    <graphic url="r2img.png"/>
  </glyph>
</charDesc>
```

# Character and glyph markup

Occurrence of these two special "r"s in the text can be marked up using the element `<g>`:

```
<p>Wo<g ref="#r1">r</g>ds in this  
manusc&r2;ipt are sometimes  
written in a funny way.</p>
```

What appears might be the letters `r1` and `r2`, or the graphics, or something else.  
The entity `r2` must also be defined.

# Names and People

The `<person>` element is used to hold data about a person; a name may refer to it.

```
... <name key="CIBC01">Colley  
Cibber</name>....  
  
<person xml:id="CIBC01">  
  <persName>  
    <forename>Colley</forename>  
    <surname>Cibber</surname>  
  </persName>  
  <birth date="1671-06-11">London</birth>  
  <death date="1757-11-12"/>  
  <occupation>playwright and actor</occupation>  
</person>
```

# Names and Referring Strings

- The `<rs>` (referring string) element is used for any kind of nominal reference

```
<q>My dear <rs type="person" key="BENM1">Mr.  
Bennet</rs>,</q> said <rs type="person" key="BENM2">  
his lady</rs> to him one day,<q>have you heard that  
<rs type="place" key="NETP1">Netherfield Park</rs>  
is let at last?</q>
```

# Correction and Regularization

- `<corr>` marks a correction
- `<sic>` marks a (deliberate) non-correction
- `<reg>` marks a regularization
- `<orig>` marks something deliberately un-normalized
- Use `<choice>` to indicate a combination of possible encodings

# For his nose was as sharp as a pen and a table of green feelds

```
... and <reg>he</reg>  
<corr resp="#Theobald">babbl'd</corr> ...
```

```
... and  
<choice>  
  <orig>a</orig>  
  <reg>he</reg>  
</choice>  
<choice>  
  <sic>table</sic>  
  <corr resp="#Theobald">babbl'd</corr>  
</choice>  
of green  
<choice>  
  <orig>feelds</orig>  
  <reg>fields</reg>  
</choice>
```



# 'Inter' class elements

- `<list>` lists of all kinds
- `<note>` notes (authorial or editorial)
- `<figure>` pictures or figures
- `<table>` tables
- `<bibl>` bibliographic descriptions

# Lists

- use `<list>` for lists of any kind (use type attribute to distinguish)
- use `<label>` in two-column lists as alternative to n attribute
- may be nested as necessary

# for example...

```
<list type="xmas">
  <label>For my true love</label>
  <item>
    <list type="bullets">
      <item>three calling birds</item>
      <item>two french hens</item>
      <item>a partridge in a pear tree</item>
    </list>
  </item>
  <label>For Uncle Joe</label>
  <item>socks as usual</item>
</list>
```

# Figures and graphics

The presence of a graphic is indicated by the `<graphic>` element, usually contained within a `<figure>` element which groups together:

- The title of the graphic (`<head>`)
- A description of the graphic (`<figDesc>`) for use by software unable to render the graphic
- The graphic resource itself is pointed to by an url attribute on the `<graphic>` element, and may also have attributes scale, height, width
- Alternatively, it may be directly embedded within a `<binaryObject>` element
- `<figure>`s may self-nest, and may also contain other display class items such as `<formula>`s

# Example



```
<figure>  
<head>Mr Fezziwig's Ball</head>  
<figDesc>A Cruikshank engraving showing  
Mr Fezziwig leading a  
group of revellers.</figDesc>  
<graphic url="fezz.gif"/></figure>
```

# Tables

- a `<table>` element contains `<row>`s of `<cell>`s
- spanning is indicated by `rows` and `cols` attributes
- `role` attribute indicates whether row or column holds data or a label
- embedded tables are permitted

# for example...

A three column table

Row1	123	4567
Row2	abc	defgh

```
<table>
<row>
  <cell cols="3" role="label">A three column table</cell>
</row>
<row>
  <cell role="label">Row1</cell>
  <cell>123</cell><cell>4567</cell>
</row>
<row>
  <cell role="label">Row2</cell>
  <cell>abc</cell><cell>defgh</cell>
</row>
</table>
```

# Bibliography

- The `<listBibl>` element lists bibliographic citations
- Individual citations may be represented loosely as `<bibl>` elements, or in a more structured way as `<biblStruct>` elements
- In either case, elements from the `model.biblPart` class are used, e.g.
  - `<author>`, `<editor>`, (generic) `<respStmt>` etc.
  - `<title>` with optional `level` attribute to distinguish monographic, analytic etc.
  - `<imprint>` groups publication info (publisher, date etc.)
  - `<biblScope>` adds page references etc.
- Individual citations may be linked to in the usual way



# Example

<p>See for example <ref  
target="#REG92">Regis (1992)</ref>...

```
<div><head>Bibliography</head>
<listBibl>
  <bibl xml:id="REG92">
    <author>Ed Regis</author>
    <title level="m">Great Mambo Chicken and
      the Trans-Human Experience</title>
    <pubPlace>London </pubPlace>
    <publisher>Penguin Books</publisher>
    <date>1992</date>
    <biblScope>pp 144 ff</biblScope>
  </bibl>
</listBibl>
</div>
```

# Notes

- Use `<note>` for notes of any kind (editorial or authorial)
- if in-line, use place attribute to specify location
- if out of line, either use
  - target attribute to specify attachment point
  - or mark attachment point as a `<ref>`

# for example...

```
<lg><l>The self-same moment I could pray></l>
<l>And from my neck so free</l>
<l>The albatross fell off, and sank</l>
<l>Like lead into the sea.
<note type="auth" place="margin">
The spell begins to break.</note> </l>
</lg>
```

or

```
...
<l>The albatross fell off, and sank</l>
<l xml:id="L213">Like lead into the sea. </l>
</lg>
...
<note type="auth" place="margin" target="#L213">
The spell begins to break.</note>
```

# Other Modules

Your choice from:

- 1 Transcription of spoken texts
- 2 Dictionaries and lexica
- 3 Varieties of linguistic annotation
- 4 Nonstandard characters and glyphs
- 5 Linking, alignment, non-hierarchic structures
- 6 Detailed metadata (the TEI Header)
- 7 Manuscript Description
- 8 Text-critical apparatus
- 9 Physical description
- 10 Onomastics and ontologies
- 11 The ODD system

# Exploring TEI P5

- Visit <http://www.tei-c.org/release/doc/tei-p5-doc/html/>
- alphabetical lists of classes, macros, elements
- each chapter describes a distinct module
- each module presents a semantically related list of elements, with examples of their use

**Feedback and advice available to all on  
[tei-l@listserv.brown.edu](mailto:tei-l@listserv.brown.edu)**