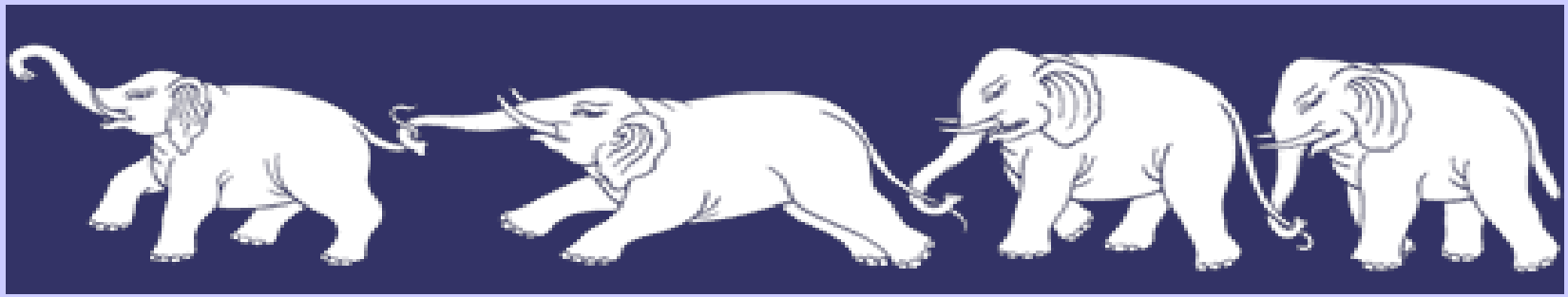
TEI for CSL

Workshop 24 Sep 04

Lou Burnard
Stuart Brown

http://www.tei-c.org/Projects/CSL



Aims of the workshop

- A brief demystification of acronyms
  - What is XML? What is TEI? Why should we care?
- Discussion of a proposed new XML-based workflow for the CSL
- Hands-on experience of a customized XML editor
- (time permitting) Demonstration of some XML text analysis software

A computer is not a typewriter...

- Texts are more than simply sequences of glyphs
  - They have *structure* and *context*
  - They also have multiple *readings*
- Encoding or markup provides a means of making such readings explicit
  - only that which is explicit can be digitally processed
- Digital processing is about more than reproducing paper

What is markup for?

- Markup is a way of making explicit the distinctions we want a computer to make when it processes a string of bytes (aka a text)
- It's a way of naming and identifying the parts of a document in a controlled way
- Consequently, it's (usually) more useful to markup what things *are* than what they *look like*

Textual ontologies

- Adding value by multiple annotations
- Facilitate re-use of digital resources
  - In different contexts
  - In different formats
  - For different audiences
  - For different purposes
- Texts can be *analysed* as well as *read*

XML: what it is and why you should care

- XML is a ***generic markup language***
- It simplifies the representation of structured data as linear character strings
- XML looks like HTML, except that:-
  - XML is extensible
  - XML must be well-formed
  - XML can be validated
  - XML is application-, platform-, and vendor- independent
- XML empowers the content provider and facilitates data integration

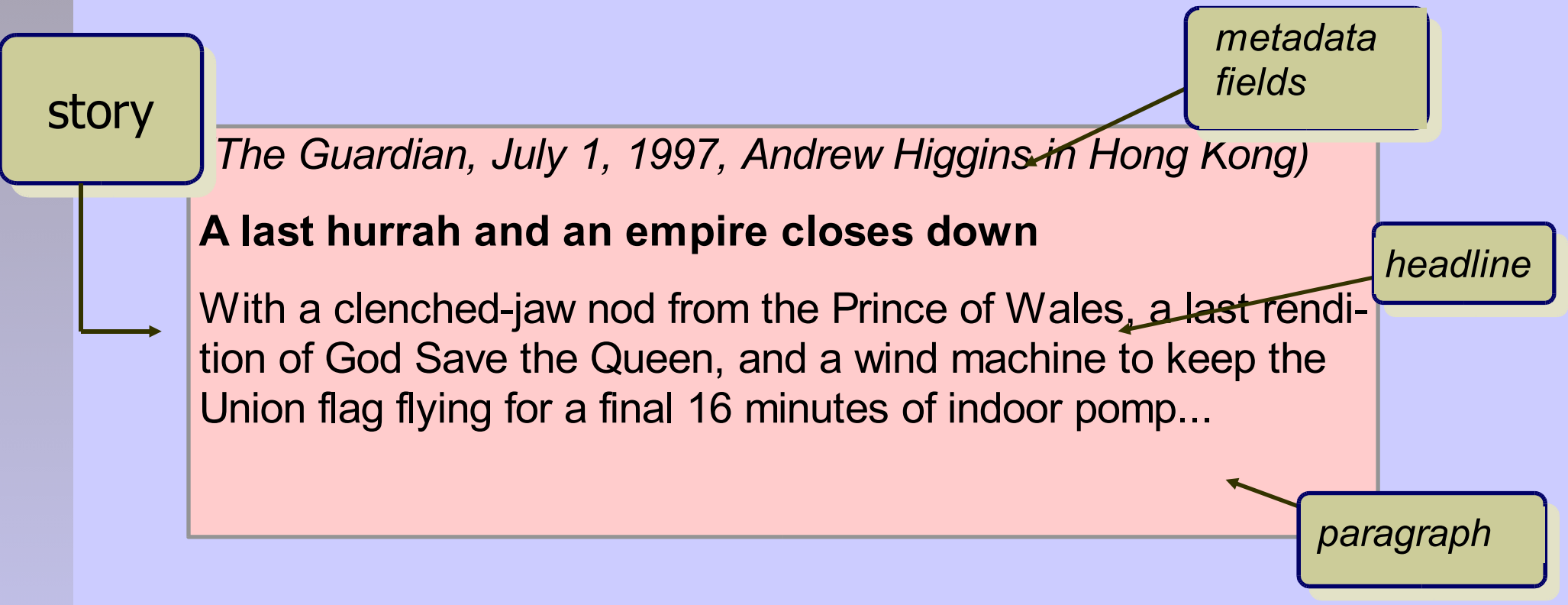
XML concepts: a review

- An XML object is composed of identifiable objects or *elements*
- Elements have a *type* (name, or GI)
- A textual grammar (a *schema*) may be defined which specifies
  - what elements exist
  - how they may be combined
- Elements also bear descriptive named *attributes*
- An XML object contains a single *hierarchy* of elements

For example:

- a newspaper story consists of metadata fields, followed by a headline, and a series of paragraphs, which may contain proper names or character data
- it also has an identifier and a language

… like this



A newspaper *story* consists of *metadata fields*, followed by a *headline*, and a series of *paragraphs;* it also has a *number* and a *language*

```
<!ELEMENT story (metadataField+, headline, paragraph+)>
<<ATTLIST story number CDATA #IMPLIED
                 language IDREF "ENG">
```

... or, in XML:

<story>

<metaDataField>*(The Guardian,* **</metaDataField>**

<metaDataField>*July 1, 1997,* **</metaDataField>**

<metaDataField>*Andrew Higgins in Hong Kong)***</metaDataField>**

<headLine>A last hurrah and an empire closes down</headLine>

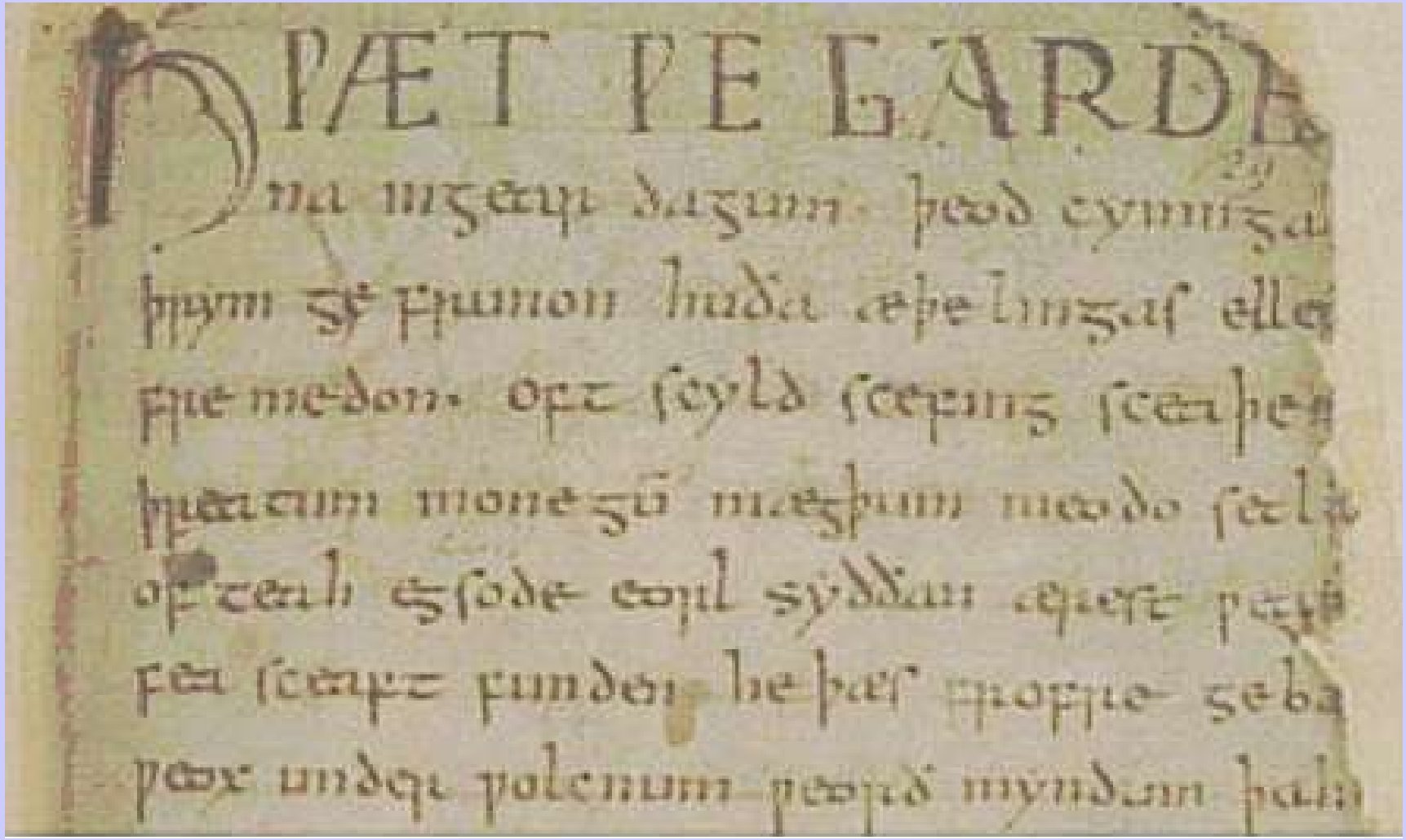
** **With a clenched-jaw nod from the Prince of Wales,
a last rendition of <title>God Save the Queen</title>, and a wind
machine to keep the Union flag flying for a final 16 minutes
of indoor pomp...** **

</story>

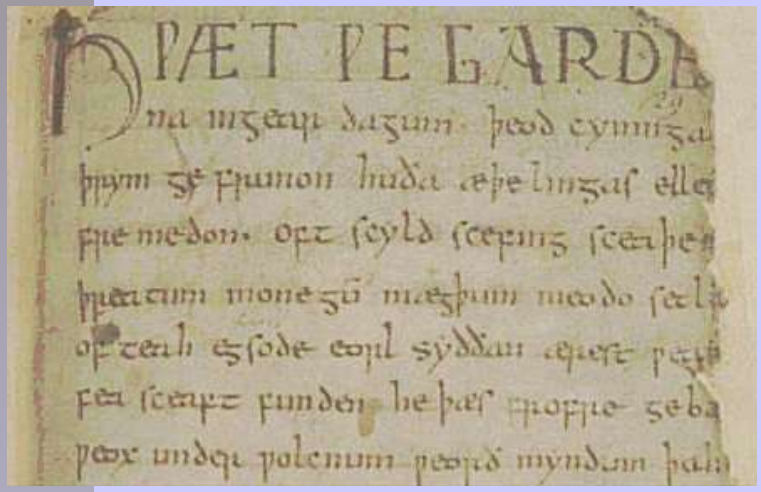
Encoding implies decisions

- We may wish to allow for many views of what a text “is”
- but avoid “markup voodoo”
- Necessarily, there must be compromise
  - what is needed now
  - what might be needed some time

The Beowulf Manuscript

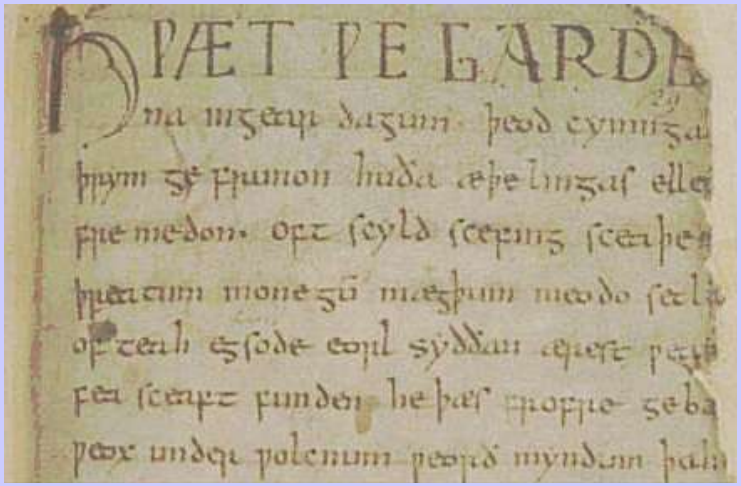


MS Cotton Vitellius A xv



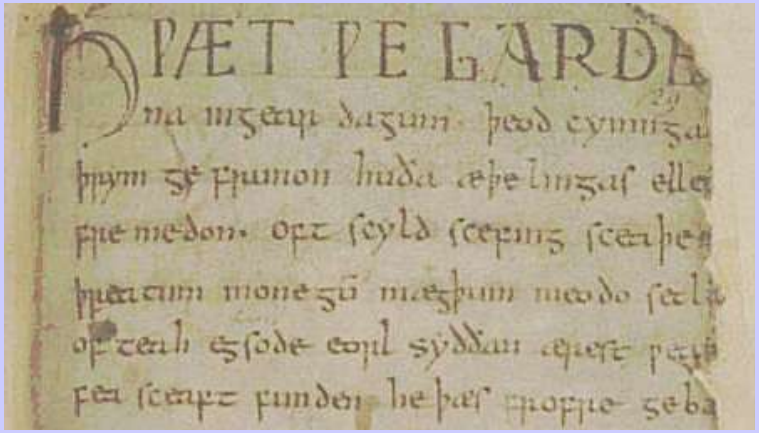
Printed version (Wrenn,1953)

Hwæt we Gar-Dena in gear-dagum
þeod-cyninga þrym gefrunon,
hu ða æþelingas ellen fremedon.
Oft Scyld Scefing sceaþena þreatum,
monegum mægþum meodo-setla ofteah;
egsode Eorle, syððan ærest wearð
feasceaft funden. He þæs frofre gebad…



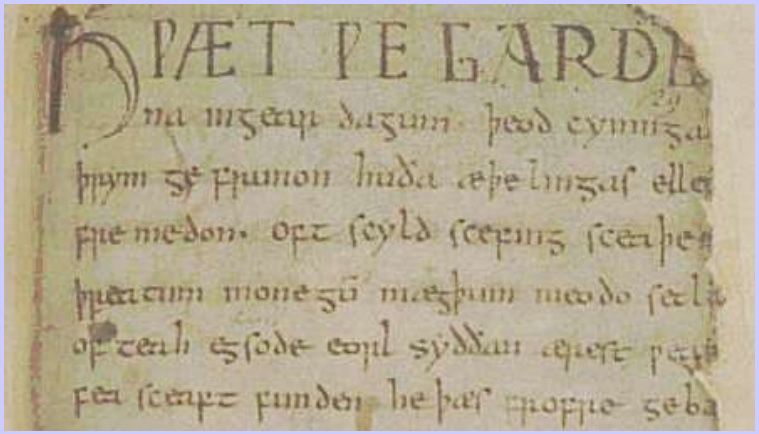
One encoding…

```
<lg><l>Hwæt we Gar-Dena in gear-dagum</l>
<l>þeod-cyninga þrym gefrunon,</l>
<l>hu ða æþelingas ellen fremedon.<l></lg>
<lg><l>Oft Scyld Scefing sceaþena
  þreatum,</l>
<l>monegum mægþum meodo-setla ofteah; </l>
<l>egsode Eorle, syððan ærest wearþ</l>
<l>feasceaft funden. He ...
```



… another encoding

```
<hi rend='caps'>&H;&Wyn;ÆT &Wyn;E GARDE</hi><lb/>na in gear-dagum þeod cyninga<lb/> þrym gefrunon hu ða æþelinga&s; ellen<lb/> fremedon. oft Scyld Scefing sceaþe<add>na</add><lb/>þreatum, moneg<expan>um</expan> mæ;gþum meodo-setla <lb/>
 of<damage desc='blot'/>teah egsode <sic corr='Eorle'>eorl</sic> syððan ærest wearþ<lb/> feasceaft funden...
```



…yet another encoding

```
<figure>
<!-- detailed description of digital image -->
</figure>
<sourceDesc>
<!-- detailed description of original source-->
</sourceDesc>
<publicationStmt>
<!- access control metadata -->
</publicationStmt>
<classCode>
<!- descriptive metadata -->
</classCode>
<!- etc -->
```

Where is XML used?

- On the web...
- In well-defined application areas
  - b2b
  - news stories
  - chemical modelling
- By well-defined user communities
  - EAD
  - electronic editors

XML: the very next thing

- XML defines a simple syntax for encoding tree structured data as strings. It is
  - extensible
  - verifiable
- XML is therefore being taken up enthusiastically as a way of
  - adding semantics to the web (RDF, Topic Maps)
  - standardizing application interfaces (SOAP, WSDL)
- ... even though XML is semantics-free

Reality check: what (exactly) is markup?

- markup makes explicit a theory about some aspect of a document
- some theories are more useful or generalizable than others
- … so no markup language can reasonably claim to be exhaustive
- … so are we doomed to a further confusion of tongues?

The risks of fragmentation

- If we have…
  - historical records using a “historical markup language”
  - linguistic data using a “linguistic markup language”
  - illustrations using a “visual markup lan-guage”
- How will we integrate these resources?
- Why did we get into this business?

We've been here before…

Loomings

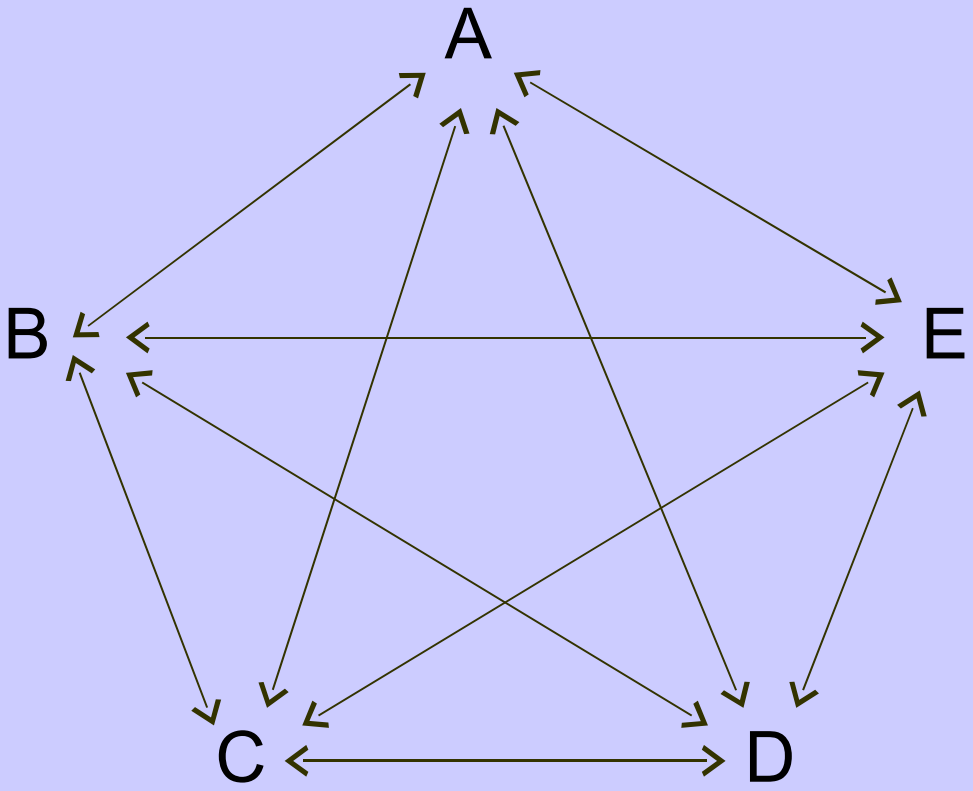
"CALL
mind ho
money i
terest m
a little a

er
no
to in-
bout
'

```
|chap1
<C 1> Loomings
\chapter
\chapter[1]{Loomings}
:h1.1.  Loomings
MOBY001001LOOMINGS
|C1
 chapter Loomings
~x
```

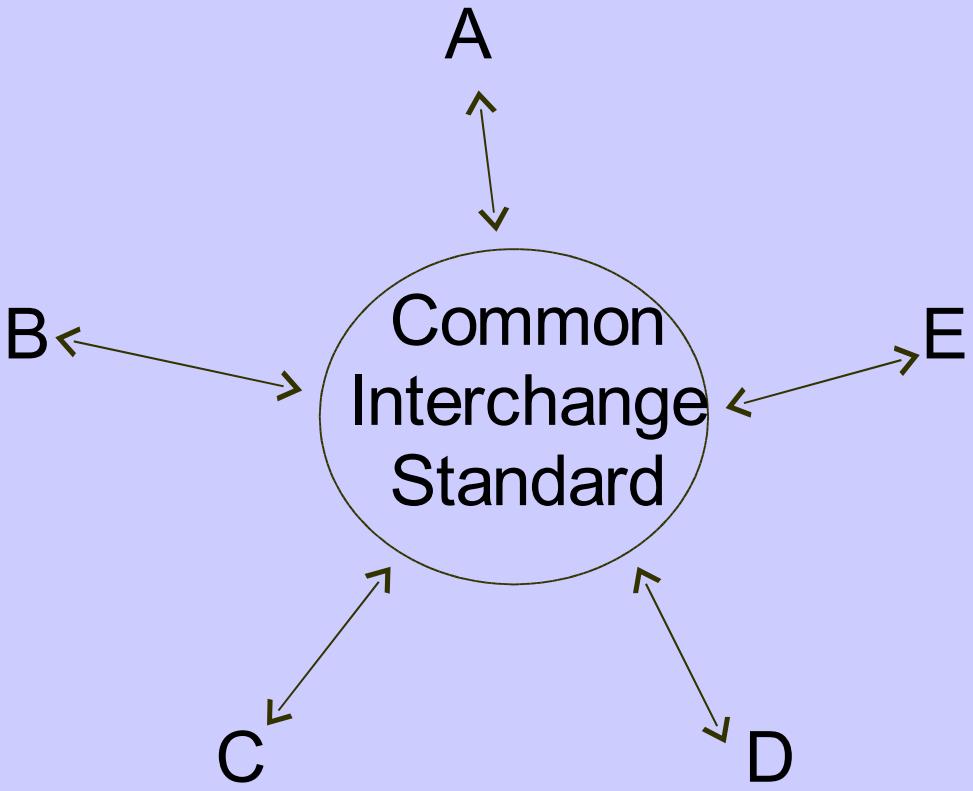
Bad news: there ARE 400 different encoding formats…

Information Interchange (1)



20 translations required (n^2-n)

Information Interchange (2)



10 translations required (2n)

The T E what?

- Originally, a research project within the humanities
  - Sponsored by ALLC, ACH, ACL
  - Funded 1990-1994 by US NEH, EU LE Programme *et al*
- Major influences
  - digital libraries and text collections
  - language corpora
  - scholarly datasets
- Now an international membership consortium incorporated Jan 2001

http://www.tei-c.org

Goals of the TEI

- interchange and integration of scholarly data
- support for all texts, in all languages, from all periods
- guidance for the perplexed: *what* to encode
  - hence, a user-driven codification of existing best practice
- assistance for the specialist: *how* to encode
  - hence, a loose framework into which unpredictable extensions can be fitted

Legacy of the TEI

- The TEI Guidelines: a comprehensive way of looking at what texts are and how to organize them
  - Expressed as a very large set of c. 600 element definitions, tied into a rather loose DTD
- A mechanism for customization and specialization of the above
- Tutorials, Guides,codification of shared practice etc.
- .... and a *lot* of experience

Who uses TEI?

- Digital librarians and text archivists
- Creators of language corpora
- Language engineers, lexicographers, and terminologists
- Literary scholars
- In most languages of the world, alive and dead

http://www.tei-c.org/Applications/

Current TEI activity (1)

- First AGM and elections in Pisa, November 2001
- Elected TEI Council met in London, January 2002
- XML revision (P4X) approved at Board meeting in Prague, May 2002
- XML edition published in print, June 2002

`http://www.tei-c.org/Services/order/`

Current TEI activity (2)

- 2003: work on TEI P5 began
- New work groups on
  - character set issues: convergence with Unicode
  - manuscript description
  - hyperlinking/stand off markup
  - SGML/XML conversion
  - New schema language and new customization features
- TEI P5 will be available on sourceforge end of 2004

The TEI was designed for scholarly use

- all texts are alike -- but every text is different
- multiple perspectives are the norm
- not *one size fits all* but *who would you like to be today*?
  - one construct, many views
  - each view a selection from the whole
- Standardization vs customization

The TEI architecture

- Elements represent agreed *categories*
- Elements are grouped into *modules*
- And assigned to semantic *classes*
- Wherever possible, elements are defined in terms of the classes they reference
- A *schema* is constructed by combining modules and (possibly) redefining elements within them
- A single XML language (ODD) is used both to document and to define all parts of the system

`http://www.tei.oucs.ox.ac.uk/Roma/`

TEI as an interlingua

- TEI defines generic classes of textual object

  , <ab>, <seg> rather than *chapter, paragraph, metaphor*

- Modification allows these to be more tightly constrained without loss of generality

  <metaphor TEIform="seg">fresh ideas</metaphor>

- And to add new elements as necessary
  - eg. <address> and <bibl>

SGML, XML, and …

- The TEI originally used SGML
  - for pragmatic reasons
    - existing standard, widely used
  - for theoretical reasons
    - declarative, verifiable
    - expressive power adequate to needs of research
- It is now re-expressed in XML…

… after XML?

- In fact, the TEI expresses an abstract model, which can be represented in a variety of concrete syntaxes:
  - SGML or XML DTD language
  - RelaxNG schema
  - W3C schema
- Integration of the documentation with the definition makes it independent of any particular syntax

Why bother?

- The TEI is a well-known reference point
- Using the TEI enables
  - sharing of data and resources
  - shared modular software development
  - lower learning curve and reduced training costs
- The TEI is stable, rigorous, and well-documented
- The TEI is also flexible, customizable, and extensible in documented ways
- Its architectural approach offers a good practical compromise between generality and implementability

Transmitting the hermeneutic

- scholarship depends on continuity
- it is not enough to preserve the bytes of an encoding
- there must also be a continuity of com-prehension: the encoding must be self-descriptive

The wider picture

- TEI is not just about exchanging data between machines
  - It's also about communication between humans
- TEI/XML is not just about the web
  - It's about information in general
- TEI is not just about technology
  - It's about the relationship between content creators and software developers
  - It’s also about scholarship

Using the TEI

- Which modules will you use?
- How will you customize them?
- What additional constraints will you need?
- What software will you develop?
- Where will it all be documented?