Implementing the TEI Feature System Declaration

Gary F. Simons
SIL International

TEI Members Meeting
11 Oct 2002, Chicago

What is a feature structure?

- A device for the linguistic analysis of text
- A recursive bundle of feature-value pairs
-

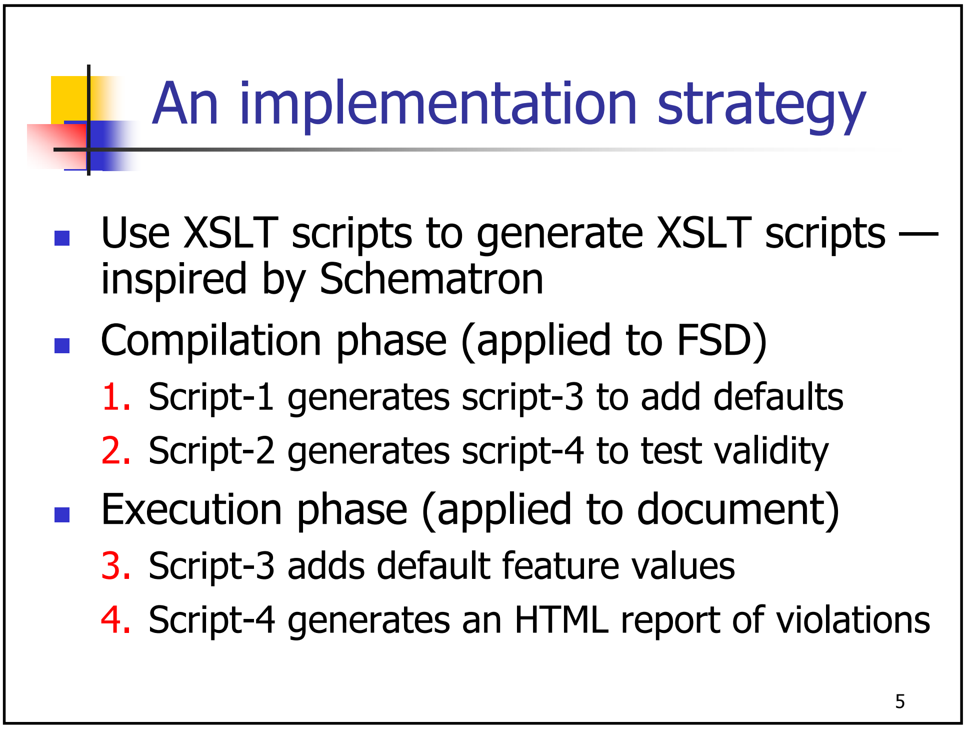
```
┌                                            ┐
  category   = noun
  wordForm   = Kind
  proper     = -
                ┌                 ┐
  agreement  =   gender = neut
                 number = sg
                 case   = nom
                └                 ┘
└                                            ┘
```

In TEI markup

```
<fs>
  <f name="category"><sym value="noun"></f>
  <f name="wordForm"><str>Kind</str></f>
  <f name="proper"><minus/></f>
  <f name="agreement">
    <fs><f name="gender">
                <sym value="neut"></f>
            <f name="number">
                <sym value="sg"></f>
            <f name="case"><sym value="nom"></f>
    </fs></f>
</fs>
```

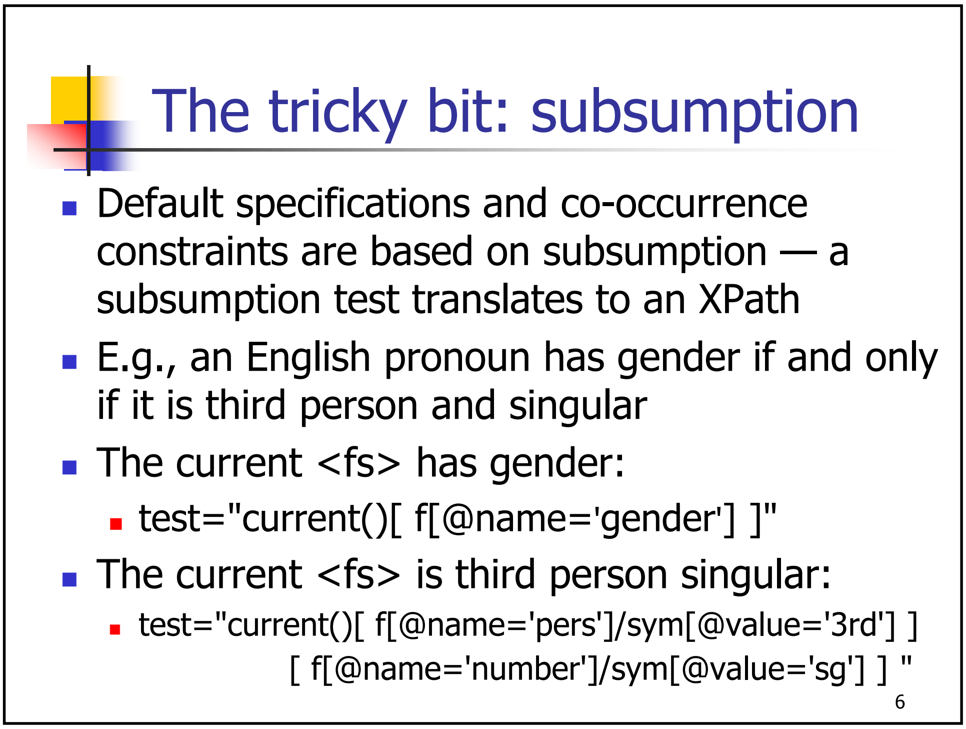
What is an FSD?

- An auxiliary document type used in conjunction with <fs> markup to:
  - Document the allowed features
  - Document their allowed values
  - Specify default values for underspecified features
  - Specify constraints on feature co-occurrence
- In short: “It’s an *XML schema language* for <fs> markup.”



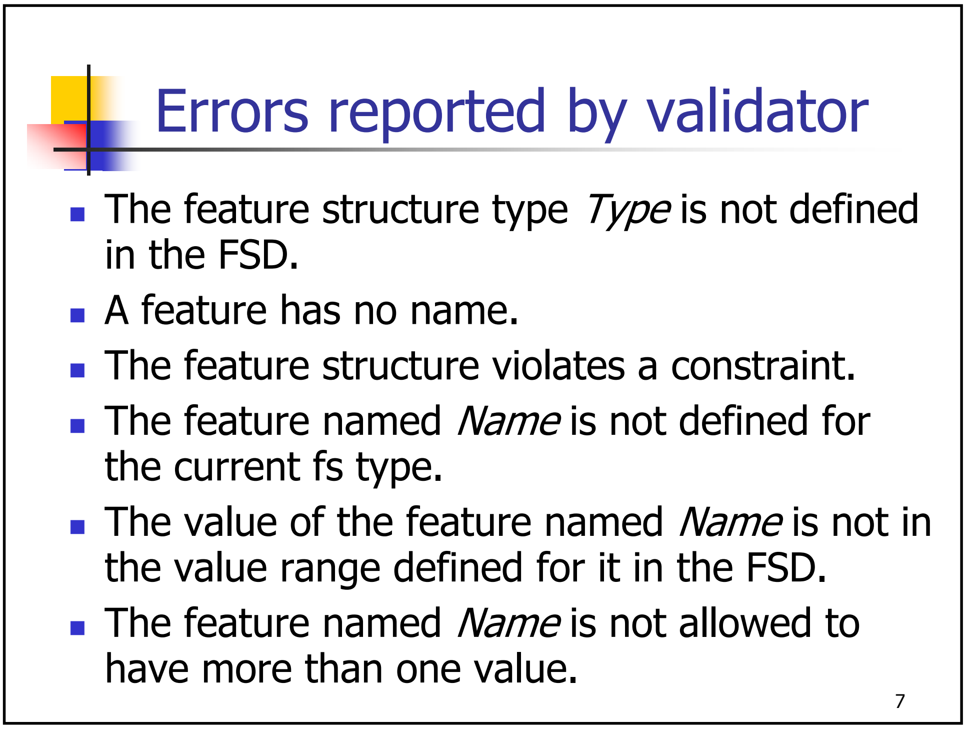
An implementation strategy

- Use XSLT scripts to generate XSLT scripts — inspired by Schematron
- Compilation phase (applied to FSD)
  1. Script-1 generates script-3 to add defaults
  2. Script-2 generates script-4 to test validity
- Execution phase (applied to document)
  3. Script-3 adds default feature values
  4. Script-4 generates an HTML report of violations



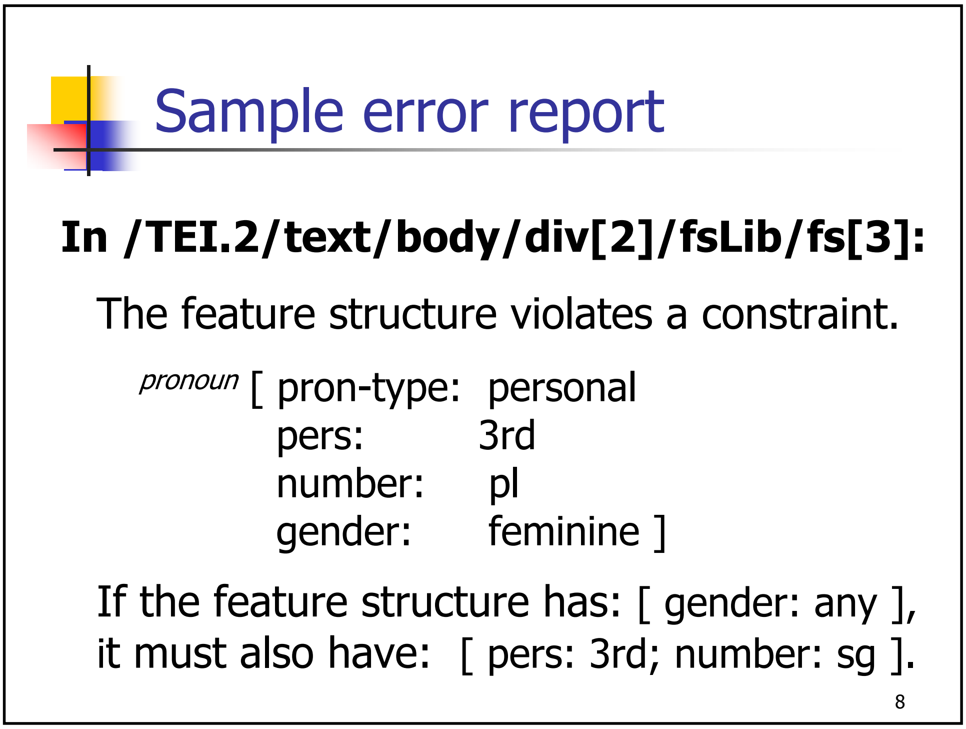
The tricky bit: subsumption

- Default specifications and co-occurrence constraints are based on subsumption — a subsumption test translates to an XPath
- E.g., an English pronoun has gender if and only if it is third person and singular
- The current <fs> has gender:
  - test="current()[f[@name='gender']]"
- The current <fs> is third person singular:
  - test="current()[f[@name='pers']/sym[@value='3rd']]
    [f[@name='number']/sym[@value='sg']] "



Errors reported by validator

- The feature structure type *Type* is not defined in the FSD.
- A feature has no name.
- The feature structure violates a constraint.
- The feature named *Name* is not defined for the current fs type.
- The value of the feature named *Name* is not in the value range defined for it in the FSD.
- The feature named *Name* is not allowed to have more than one value.



Sample error report

In /TEI.2/text/body/div[2]/fsLib/fs[3]:

The feature structure violates a constraint.

pronoun [pron-type: personal
pers: 3rd
number: pl
gender: feminine]

If the feature structure has: [gender: any],
it must also have: [pers: 3rd; number: sg].

Toward an ISO standard?

- It has been proposed that TEI feature structure markup be put forward to the new ISO TC37/SC4 as a proposed standard
- TC 37 — “Terminology and other language resources”
  - SC 4 — “Language resources”
  - Chair: Laurent Romary

Some issues

- Current DTDs for <fs> and FSD are intertwined with the TEI DTD:
  - An ISO standard would need to stand on its own.
- Current scheme has bells and whistles that have never been implemented:
  - An ISO standard should be simplified and be backed by a working implementation.

Making it stand on its own

- Drop TEI extension mechanisms in favor of fixed names and content models.
- In the DTD for the FSD:
  - Drop dependency on TEI header in favor of a header with a content model of ANY.
  - Drop dependency on TEI %paraContent in favor of a documentation element with a content model of ANY.

Making it simpler

- Drop most global attributes.
- Drop <alt>; <fAlt> is adequate.
- Drop value types motivated by general data representation (e.g. <nbr>, <msr>, <rate>)
- Rethink special values in light of implementation (e.g. <uncertain>, <dft>, <none>, <any>)
- Rethink relation attribute in light of implementation (e.g. eq, ne, sb, ns)