

# TEI Processing Model

TEI Annual Conference and Members' Meeting 2025

# Workshop outline

- Rationale behind TEI Processing Model
  - XSLT: duplicate code for different output formats, e.g. TEI Stylesheets
  - Case study: FRUS (3.5k lines of code per output format vs 350 lines of ODD)
- Data vs Presentation or Encoding vs visualization
- Case studies
  - dateline, choice with abbr/expan, persName & person note
- Very quick introduction to XPath
  - XML as a tree; element and attribute nodes; parent/child, ancestor/descendant, root
  - filters
  - Sequences
- Main concepts of the TEI Processing Model
  - Typical behaviours and content parameter adjustments
  - Styling via cssClass or tei-\* classes
- Hands-on session: the ODD editor

<https://github.com/eeditiones/workshop>

Parallel segmentation: aligned reading views

```

<div type="poem">
  <head>Cuarto solo</head>
  <app>
    <rdg wit="#ms">
      <pb facs="resources/images/cuarto-solo.png"/>
      <del rend="overstrike">Si alguna vez</del>
    </rdg>
  </app>
</lg>
<l n="1">
  <app>
    <rdg wit="#ms">
      <add place="margin">Si</add>
    </rdg>
    <rdg wit="#pub">Si</rdg>
  </app> te atreves a <app>
    <rdg wit="#ms">
      <del rend="overstrike">
        <gap reason="illegible" extent="7" unit="chars"/>der</del>
      </rdg>
    </app> sorprender</l>
  <l n="2">
    <app>
      <rdg wit="#ms">el sentido</rdg>
      <rdg wit="#pub">la verdad</rdg>
    </app> de esta vieja pared;</l>
  </l>
</div>

```

Encoding

```

<div type="poem">
  <head>Cuarto solo</head>
  <app>
    <rdg wit="#ms">
      <pb facs="resources/images/cuarto-solo.png"/>
      <del rend="overstrike">Si alguna vez</del>
    </rdg>
  </app>
  <lg>
    <l n="1">
      <app>
        <rdg wit="#ms">
          <add place="margin">Si</add>
        </rdg>
        <rdg wit="#pub">Si</rdg>
      </app> te atreves a <app>
        <rdg wit="#ms">
          <del rend="overstrike">
            <gap reason="illegible" extent="7" unit="chars"/>der</del>
          </rdg>
        </app> sorprender</l>
    <l n="2">
      <app>
        <rdg wit="#ms">el sentido</rdg>
        <rdg wit="#pub">la verdad</rdg>
      </app> de esta vieja pared;</l>
    </l>
  </lg>
</div>

```

Encoding of variants: two alternative readings

```

<div type="poem">
  <head>Cuarto solo</head>
  <app>
    <rdg wit="#ms">
      <pb facs="resources/images/cuarto-solo.png"/>
      <del rend="overstrike">Si alguna vez</del>
    </rdg>
  </app>
</lg>
  <l n="1">
    <app>
      <rdg wit="#ms">
        <add place="margin">Si</add>
      </rdg>
      <rdg wit="#pub">Si</rdg>
    </app> te atreves a <app>
      <rdg wit="#ms">
        <del rend="overstrike">
          <gap reason="illegible" extent="7" unit="chars"/>der</del>
        </rdg>
      </app> sorprender</l>
  <l n="2">
    <app>
      <rdg wit="#ms">el sentido</rdg>
      <rdg wit="#pub">la verdad</rdg>
    </app> de esta vieja pared;</l>

```

Encoding of variants: only one reading

```

<div type="poem">
  <head>Cuarto solo</head>
  <app>
    <rdg wit="#ms">
      <pb facs="resources/images/cuarto-solo.png"/>
      <del rend="overstrike">Si alguna vez</del>
    </rdg>
  </app>
</lg>
<l n="1">
  <app>
    <rdg wit="#ms">
      <add place="margin">Si</add>
    </rdg>
    <rdg wit="#pub">Si</rdg>
  </app> te atreves a <app>
    <rdg wit="#ms">
      <del rend="overstrike">
        <gap reason="illegible" extent="7" unit="chars"/>der</del>
      </rdg>
    </app> sorprender</l>
<l n="2">
  <app>
    <rdg wit="#ms">el sentido</rdg>
    <rdg wit="#pub">la verdad</rdg>
  </app> de esta vieja pared;</l>

```

Encoding of common text





Cuarto solo

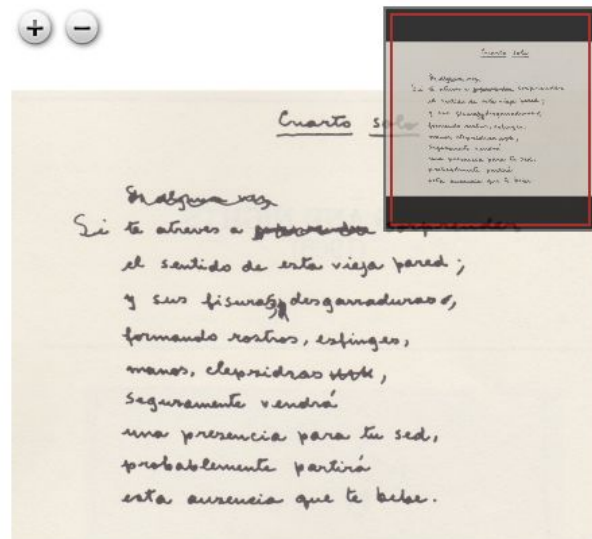
## Cuarto solo

Si alguna vez

Si te atreves a [..7..] der sorprender  
el sentido de esta vieja pared;  
y sus fisuras, [..1..] desgarraduras [..1..],  
formando rostros, esfinges,  
manos, clepsidras [..3..],  
seguramente vendrá  
una presencia para tu sed,  
probablemente partirá  
esta ausencia que te bebe.

## Cuarto solo

Si te atreves a sorprender  
la verdad de esta vieja pared;  
y sus fisuras, desgarraduras,  
formando rostros, esfinges,  
manos, clepsidras,  
seguramente vendrá  
una presencia para tu sed,  
probablemente partirá  
esta ausencia que te bebe.



Reading view of both witnesses



Cuarto solo

## Cuarto solo

Si alguna vez

Si te atreves a [..7..] der sorprend

el sentido de esta vieja pared;

y sus fisuras; s, [..1..] desgarraduras [..1..],

formando rostros, esfinges,

manos, clepsidras [..3..],

seguramente vendrá

una presencia para tu sed,

probablemente partirá

esta ausencia que te bebe.

## Cuarto solo

Si te atreves a sorprender

la verdad de esta vieja pared;

y sus fisuras, desgarraduras,

formando rostros, esfinges,

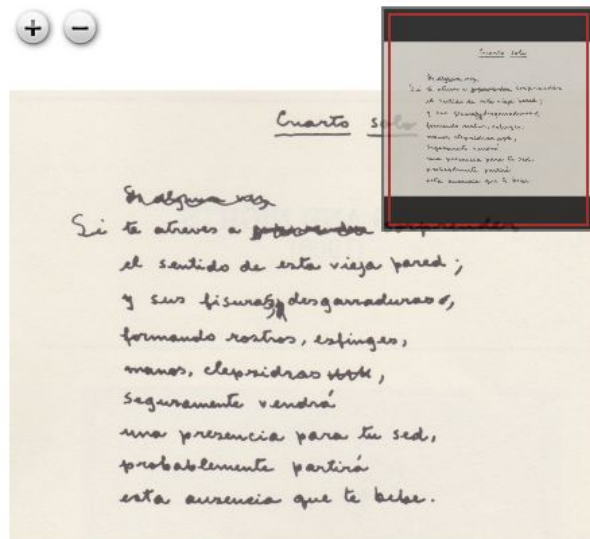
manos, clepsidras,

seguramente vendrá

una presencia para tu sed,

probablemente partirá

esta ausencia que te bebe.



Alignment at the verse line level

# Setting-up eXist-DB and package installation

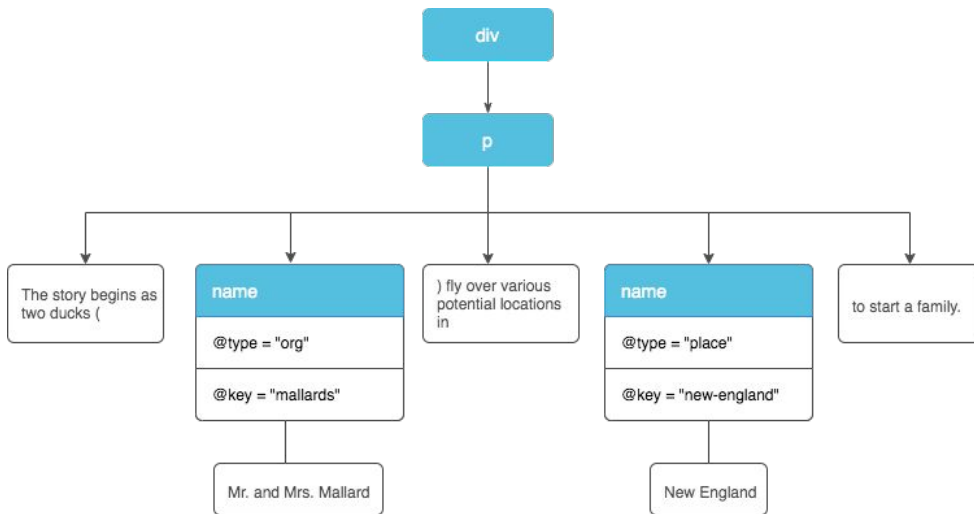
# Package installation

- Run eXist-DB
- Open <http://localhost:8080/exist/apps/dashboard/index.html>
- Log in as **admin** and click on package manager.
- Click on the add package symbol in the upper left corner and select the **.xar** file contained in **workshop/data/xars**

# Introduction to XPath

# XML as a tree

```
<?xml version="1.0" encoding="UTF-8"?>
<div xmlns="http://my.fantasy.namespace">
  <!-- This whole document is in a made-up
namespace -->
  <p>The story begins as two ducks
  (<name type="org" key="mallards">Mr.
  and Mrs. Mallard</name>)fly over
  various potential locations in <name
  type="place" key="new-england">New
  England</name> to start a family.</p>
</div>
```





# Node types in XML document

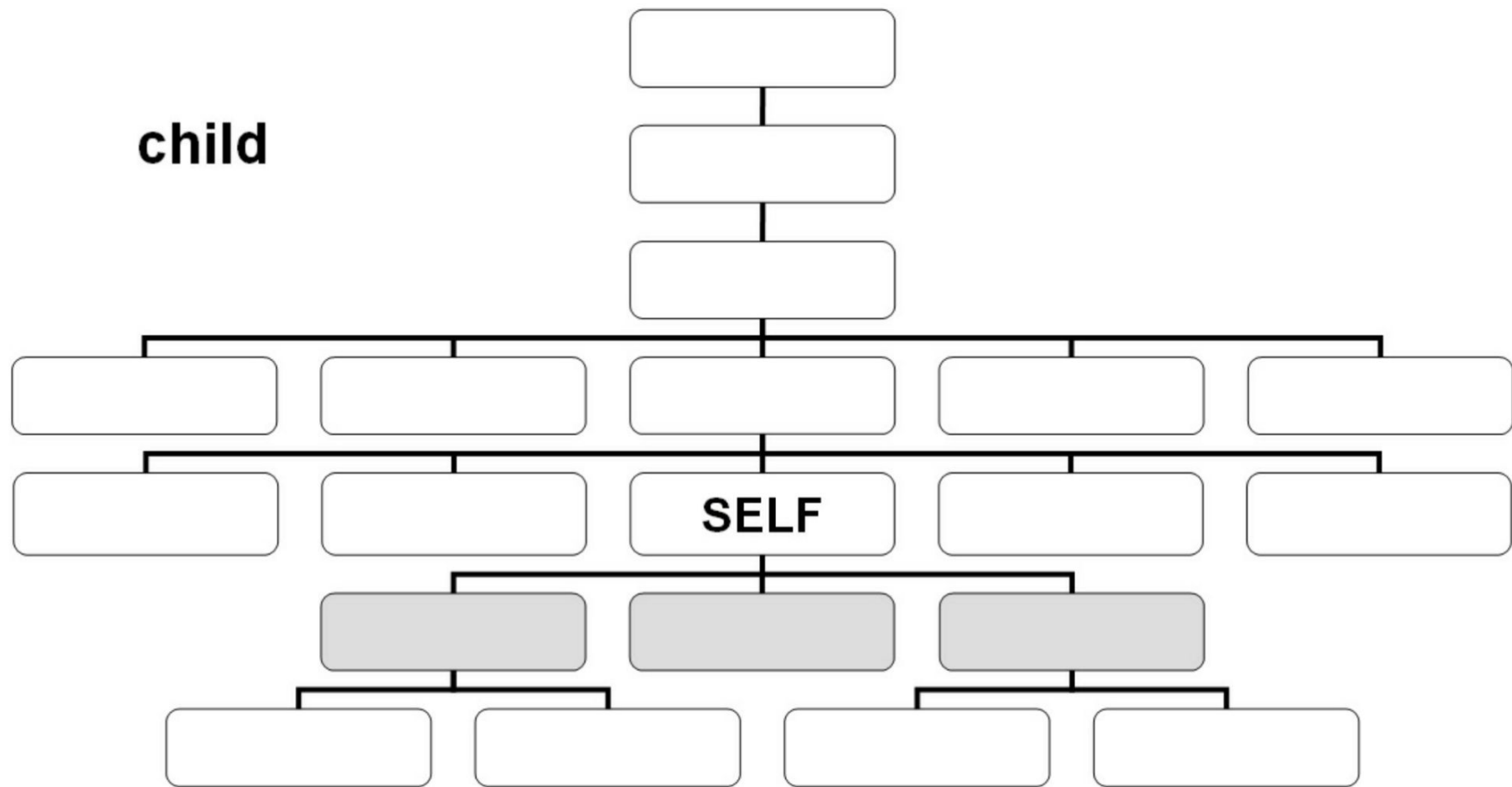
- element
- attribute
- text
- namespace
- processing-instruction
- comment
- document node



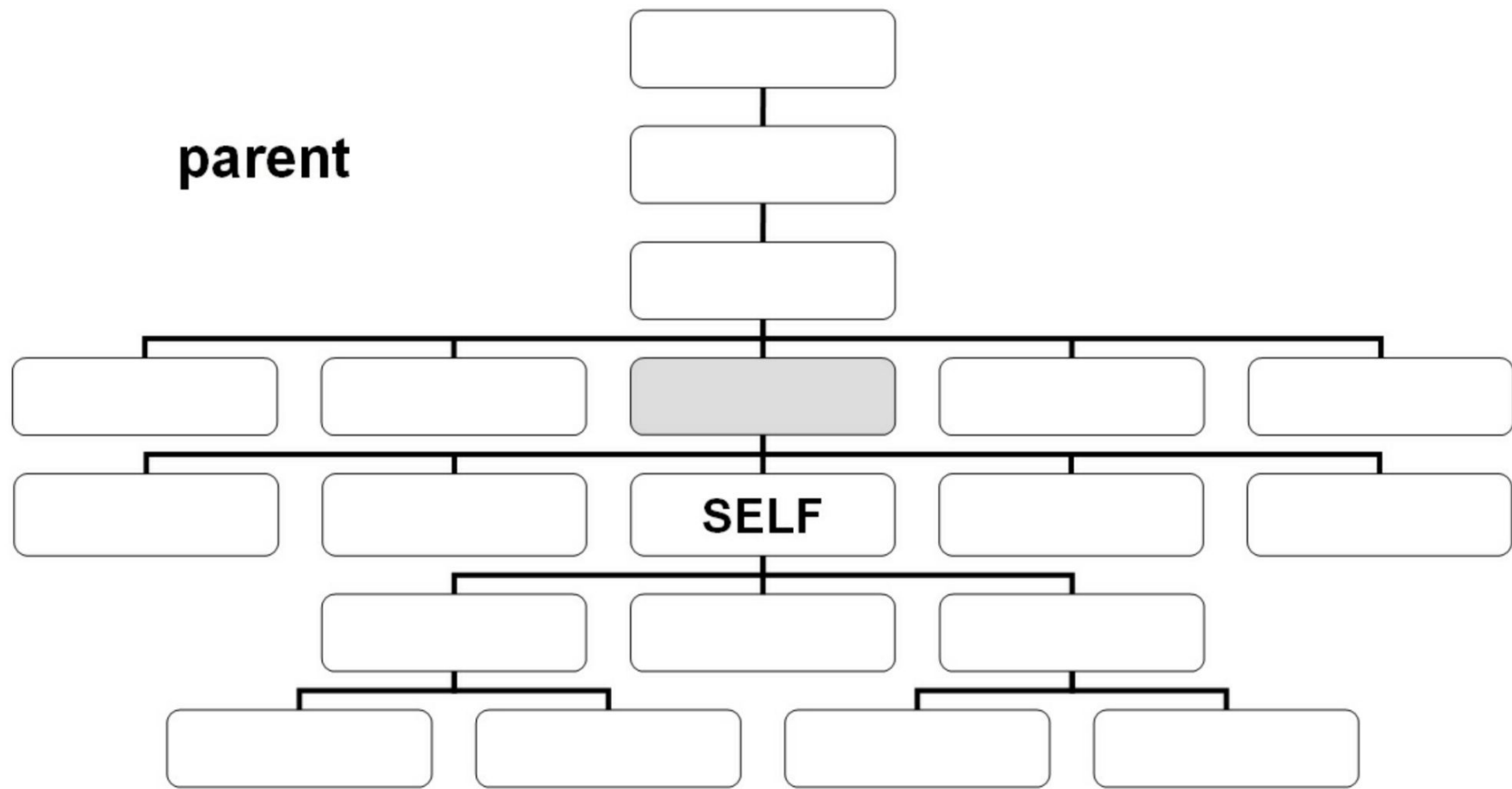
# Family relationships between nodes

- parent
- child
- ancestor
- descendant
- sibling
- self

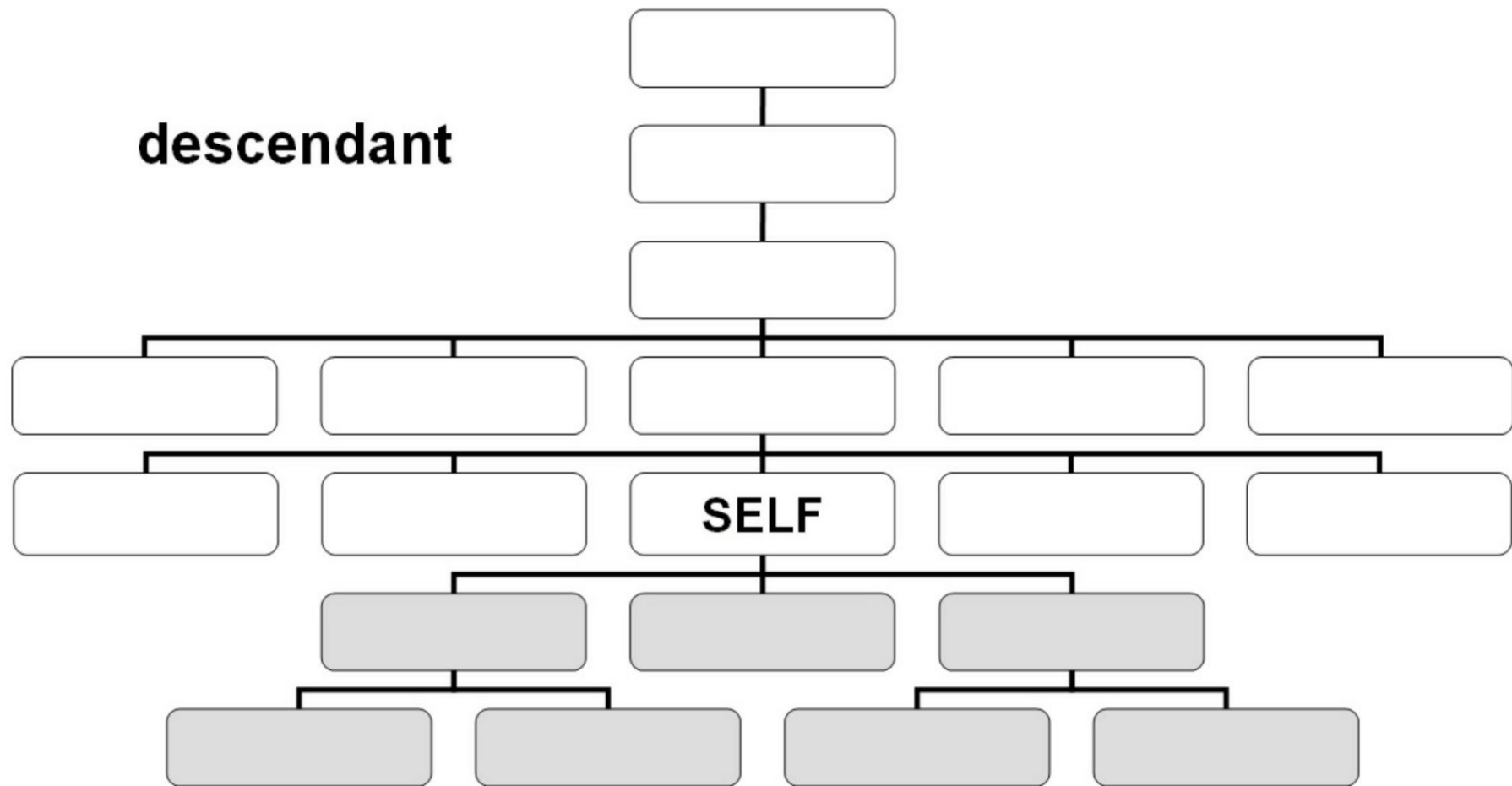
**child**



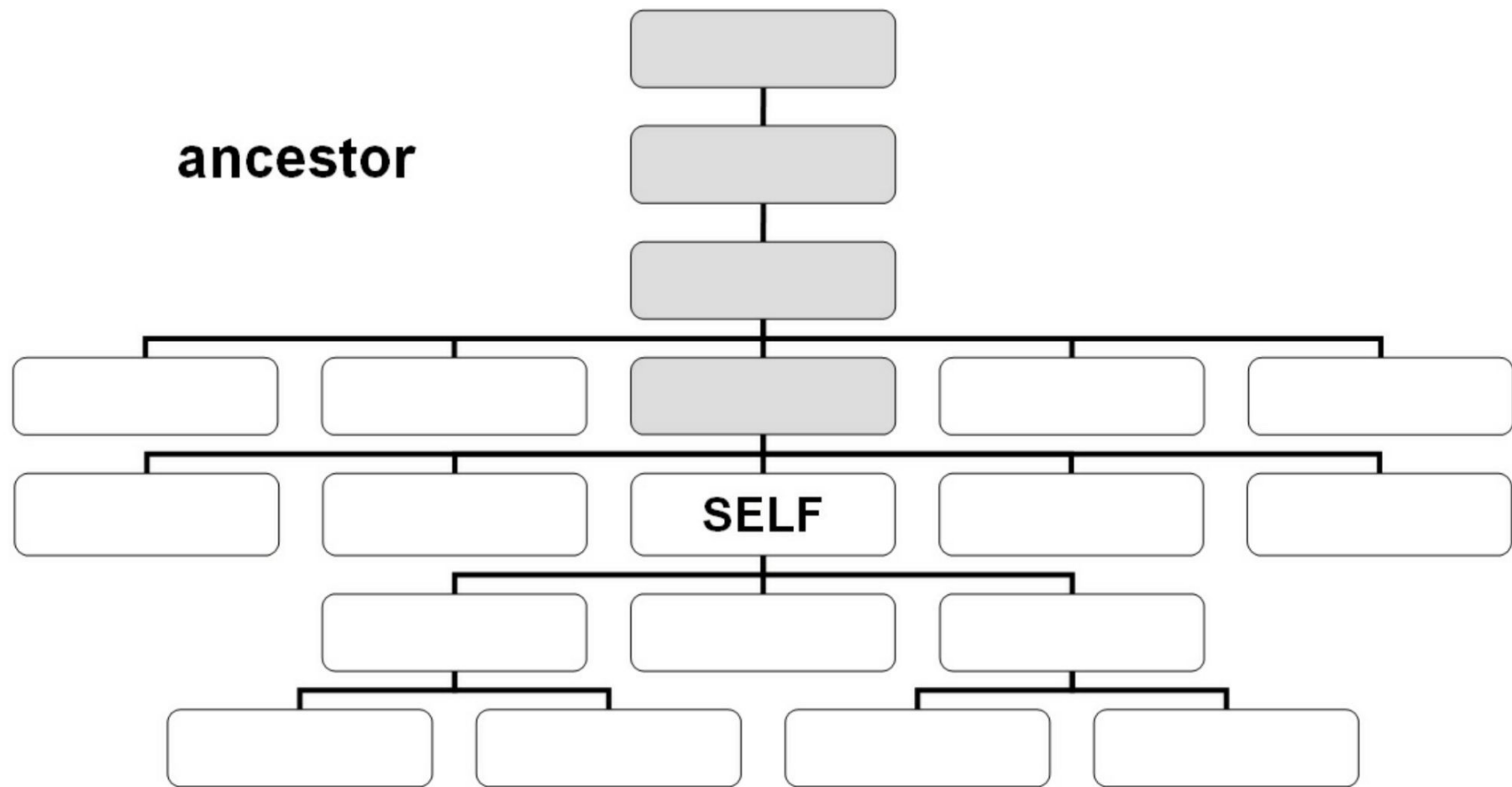
**parent**



**descendant**



**ancestor**



# XPath

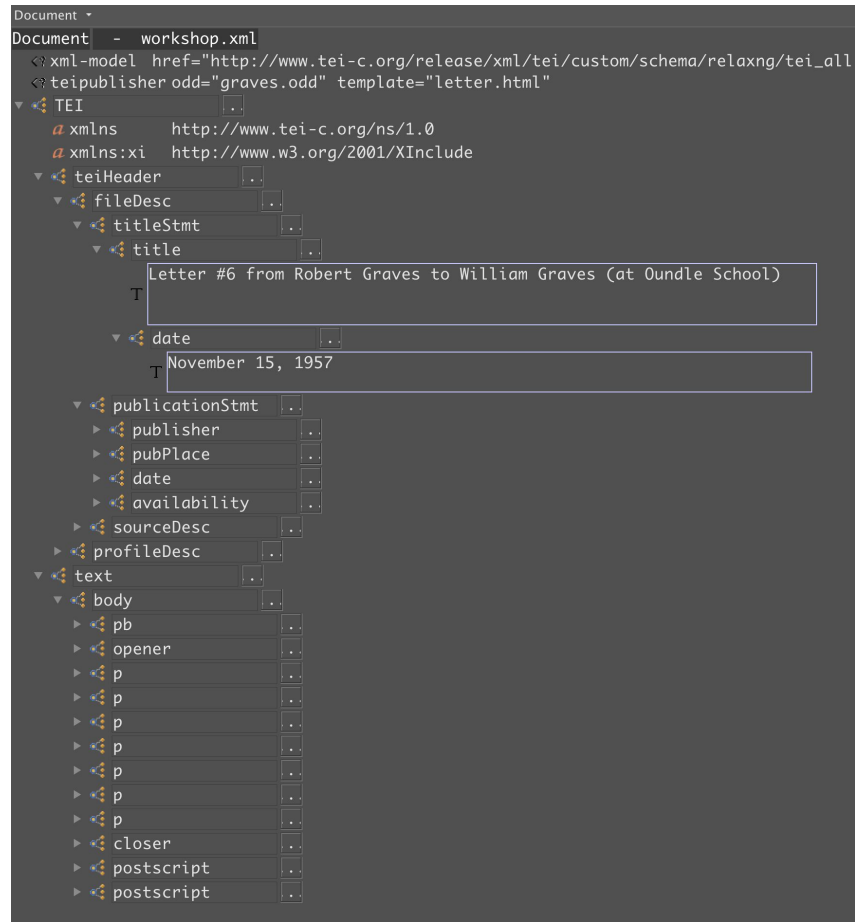
- XPath is a language for selecting sets of nodes
- Main thing an XPath expression does is to describe, in a formal way that a computer can process easily, certain sets of nodes, e.g.:
  - title of the document
  - all first paragraphs in a section
  - all references to a name of a person
  - all sections that are not of type *appendix*

```

4 <TEI xmlns="http://www.tei-c.org/ns/1.0" xmlns:xi="http://www.w3.org/2001/XInclude">
5   <teiHeader>
6     <fileDesc>
7       <titleStmt>
8         <title>Letter #6 from Robert Graves to William Graves (at Oundle School)
9         <date>November 15, 1957</date>
10      </title>
11    </titleStmt>
12    <publicationStmt>
13      <publisher>Digital Humanities at Oxford Summer School</publisher>
14      <pubPlace>Oxford</pubPlace>
15      <date when="2015-07">July 2015</date>
16      <availability> [30 lines]
17    </publicationStmt>
18    <sourceDesc>
19      <p>Letters from Robert and Beryl Graves to William Graves 1957-1967. 41 ALS and TLS,
20      56pp. , St John's College (Oxford) Library, Special Collections (Holmes 4) , Box
21      4 Gp17 </p>
22      <p>ST JOHN'S COLLEGE LIBRARY COLLECTION OWNED LETTERS FROM ROBERT (1895-1985) & BERYL
23      (1915-2003) GRAVES TO WILLIAM GRAVES (1940- ) </p>
24    </sourceDesc>
25  </fileDesc>
26  <profileDesc>
27    <textClass>
28      <catRef scheme="#genre" target="#correspondence"/>
29    </textClass>
30    <langUsage>
31      <language ident="en">English</language>
32    </langUsage>
33    <particDesc> [240 lines]
34    <settingDesc> [299 lines]
35    <correspDesc> [10 lines]
36  </profileDesc>
37 </teiHeader>

```

TEI/XML document viewed in text mode



TEI/XML document viewed in a tree editor



# Testing XPath expressions

- **Editor:** Go to <http://localhost:8080/exist/eXide/index.html>
- **Namespace:**
  - To query TEI documents, you need to declare the TEI namespace

**declare namespace tei="http://www.tei-c.org/ns/1.0";**

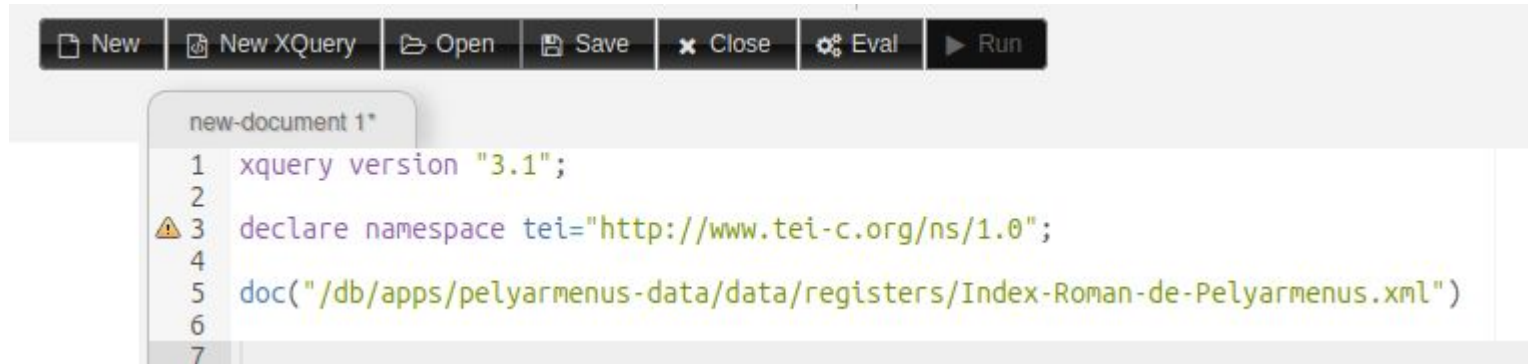


# Testing XPath expressions

- **Context:**

- specify the context with **doc()** or **collection()** functions, otherwise the entire database is searched

`doc("/db/apps/pelyarmenus-data/data/registers/Index-Roman-de-Pelyarmenus.xml")`



# Testing XPath expressions

- Get all person names (persName element) of the document
- After writing the expression, click on the **Eval** button

# Testing XPath expressions

- Get all person names (persName element) of the document
- After writing the expression, click on the **Eval** button

```
doc("/db/apps/pelyarmenus-data/data/registers/Index-Roman-de-Pelyarmenus.xml")//tei:persName
```

# Testing XPath expressions

1. Get the `<persName>` elements that are a direct child of `<person>`
2. Get the `<persName>` elements that have a `@ref` attribute
3. Count how many `<persName>` elements have a `@ref` attribute and
4. How many `<persName>` elements do not have a `@ref` attribute
5. Get the `<persName>` elements whose `@ref` attribute is equal to `"#Edypus2"`
6. Get the `<persName>` element whose `@ref` attribute starts with `"#Edypus"`

# Solutions

1. `doc("/db/apps/pelyarmenus-data/data/registers/Index-Roman-de-Pelyarmenus.xml")//tei:person/tei:persName`
2. `doc("/db/apps/pelyarmenus-data/data/registers/Index-Roman-de-Pelyarmenus.xml")//tei:persName[@ref]`
3. `count(doc("/db/apps/pelyarmenus-data/data/registers/Index-Roman-de-Pelyarmenus.xml")//tei:persName[@ref])`
4. `count(doc("/db/apps/pelyarmenus-data/data/registers/Index-Roman-de-Pelyarmenus.xml")//tei:persName[not(@ref)])`
5. `doc("/db/apps/pelyarmenus-data/data/registers/Index-Roman-de-Pelyarmenus.xml")//tei:persName[@ref = "#Edypus2"]`
6. `doc("/db/apps/pelyarmenus-data/data/registers/Index-Roman-de-Pelyarmenus.xml")//tei:persName[starts-with(@ref, "#Edypus")]`

# TEI Processing model

# TEI Processing Model

- The keystone of TEI Publisher: the implementation of the TEI Processing Model
- It formalizes the description of transformation rules in TEI (implementation independent)
- Officially part of the TEI Guidelines since 2016
- More sustainable and usually faster than customized, project-specific transformations
- It is the source of all TEI Publisher textual transformations



# TEI Processing Model

How editors and developers communicate?

*"this is a block", "this is a footnote", "a note on the margin", "an inline fragment set out in italic and highlighted", "fragment marked in <>", "a first-level heading"*

Note this quite limited vocabulary - gave rise to the concept of "**behaviours**"

## TEI Source

```
<TEI>
<text>
  <body>
```

TEI Fragment

```
<div type="chapter">
  <head>
    <p>
```

```
<div type="chapter">
  <head>
    <list>
      <item>
```



## ODD

```
<elementSpec ident="div">
  <model predicate="@type='chapter'"
    behaviour="section">
    <model behaviour="block">
```

```
<elementSpec ident="head">
  <model behaviour="heading">
```

```
<elementSpec ident="list">
  <model behaviour="list">
```



HTML

FO

LaTeX

ePub

# Customization process

- Write the TEI ODD XML by hand
- Easier and safer: using the ODD editor (accessible through a URL like: <https://workshop.jinntec.de/exist/apps/tei-publisher/odd-editor.html>)

# TEI representation

elementSpec

modelSequence

model

@predicate

@behaviour

@cssClass

outputRendition

```
<elementSpec ident="name" mode="change">
  <model predicate="@type='place'" behaviour="inline">
    <outputRendition>
      font-variant: small-caps;
    </outputRendition>
  </model>
  <model behaviour="inline">
    <outputRendition>
      color: #CC6600;
    </outputRendition>
  </model>
</elementSpec>
```

edep.odd <>   

Add Element



Jump to element ...

## Element Specs

TEI

ab

abbr

add

choice

del

ex

expan

gap

lb

note

persName

add

MODE: CHANGE



^ model [inline]

@place="overstrike"



Output

Mode

[Processing mode - passed to subsequer

Description

[Document the model]

Predicate

1 @place="overstrike"

behaviour

inline



or [Custom Behaviour]

CSS Class

[Define CSS class name (for external CSS)]

Template

<|> <...> <X> [[...]]

1 [Define code template to apply to content]

Parameters +

Name

content



Parameter

1

('«', '..', '»')

☐ set



Renditions +

☐ Use source rendition

# Behaviours library

- **alternate**
- anchor
- **block**
- body
- break
- cell
- cit
- document
- figure
- graphic
- heading
- **inline**
- list
- listItem
- metadata
- note
- **omit**
- paragraph
- **pass-through**
- row
- section
- table
- text
- title
- webcomponent

# Predicate

- In the `@predicate` attribute we establish the condition that must be met for that model to be applied
- The value of `@predicate` is an XPath expression

**XPath:** expression language for XML

# Applying models

Only the **first** matching model will be applied. **Simple and predictable!**

Matching means without any predicate or with a predicate for which its condition evaluates to true

**Nota bene:** Any following models will be **ignored** even if they match



# How is it going to look like?

## **cssClass** vs outputRendition

Normally you should assign a CSS class through the `cssClass` attribute to be used in external CSS stylesheets

`outputRendition` recommended only for strictly editorial aspects, e.g. convention to add [ ] to mark conjectures

general appearance (like margins, typographical elements etc) should be controlled by the external CSS theme for the application

# Model parameters

- All behaviors take at least one implicit parameter which corresponds to the content to be processed
  - If the parameter `content` is not specified, the current node will be processed
- Some behaviours need additional parameters, e.g.:
  - `link: uri`
- Alternate behaviour has a different name for the content to be processed:
  - `alternate: default` and `alternate`

# Model parameters

```
<elementSpec ident="date" mode="change">
  <model output="web" predicate="@when" behaviour="alternate">
    <param name="default" value="."/>
    <param name="alternate" value="format-date(@when, '[FNn]', [D1o] [MNn], [Y]')"/>
  </model>
  <model predicate="text()" behaviour="inline"/>
</elementSpec>
```

Control and adjust the transformation process through parameters.

# modelSequence

```
<elementSpec id="title" mode="change">
  <modelSequence predicate="parent::titleStmt/parent::fileDesc">
    <model predicate="preceding-sibling::title" behaviour="text">
      <param name="content" value="' - '"/>
    </model>
    <model behaviour="inline"/>
  </modelSequence>
</elementSpec>
```

For cases when we want more than one model to be applied, e.g. to display the verse number and the verse content

# Hands on session

Check the [assignment](#) on handling the apparatus