

# TEI and ODD for LINGUISTICS

A solid basis for development?

[edrenth@fryske-akademy.nl](mailto:edrenth@fryske-akademy.nl)



# What is ODD

- One Document Does all
- It is a TEI document
  - Holding one schemaSpec element
- It is the mechanism to customize TEI
  - TEI is designed to be customized
- What can it do
  - Generate validation
  - Generate documentation
  - Describe processing model
- <https://tei-c.org/guidelines/customization/>
- <https://tei-c.org/release/doc/tei-p5-doc/en/html/USE.html#IM-unified>
- <https://tei-c.org/release/doc/tei-p5-doc/en/html/TD.html#TDmodules>
- <https://tei-c.org/release/doc/tei-p5-doc/en/html/TD.html#TDPMPM>

```

<schemaSpec ident="fa-tei-dictionaries" start="TEI teiCorpus">
  <desc> [6 lines]
  <moduleRef key="core" include="p label item list author title cit quote bibl biblScope gloss publisher
  <moduleRef key="dictionaries" include="def entry etym form gram hyph orth pron sense superEntry usg l:
  <moduleRef key="tei" />
  <moduleRef key="header" except="appInfo application"/>
  <moduleRef key="textstructure" include="TEI text body"/>
  <moduleRef key="namesdates" include="person listPerson placeName persName orgName country region set
  <moduleRef key="msdescription" include="repository institution msDesc msIdentifier msName"/>
  <elementSpec ident="TEI" mode="change">
    <desc>A TEI document contains an optional header and a text (container)</desc>
    <content>
      <sequence>
        <elementRef key="teiHeader" minOccurs="0"/>
        <elementRef key="text"/>
      </sequence>
    </content>
    <constraintSpec ident="idno-check" scheme="schematron">
      <constraint>
        <assert xmlns="http://purl.oclc.org/dsdl/schematron"
          test="tei:teiHeader/tei:fileDesc/tei:sourceDesc/tei:msDesc/tei:msIdentifier/tei:idno"
          >teiHeader/fileDesc/sourceDesc/msDesc/msIdentifier/idno is required within TEI.</assert>
        </constraint>
      </constraintSpec>
    </elementSpec>
    <elementSpec ident="msIdentifier" mode="change" module="msdescription">
      <constraintSpec ident="msId_minimal" scheme="schematron" mode="delete"/>
    </elementSpec>

```

source attribute on  
schemaSpec or moduleRef:  
where exactly do I get my  
module info!

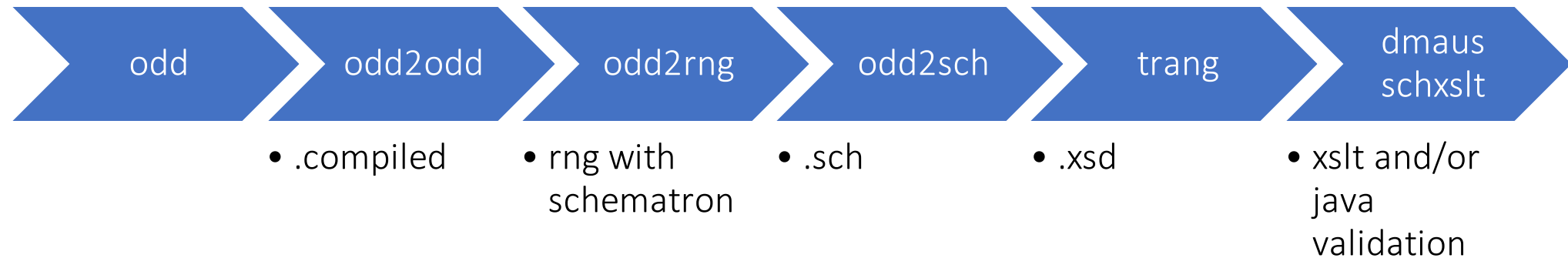
```
<classSpec type="atts" ident="att.features" module="analysis" mode="add">
  <desc> [4 lines]
  <attList>
    <attDef ident="islemma" ns="http://www.fryske-akademy.org/linguistics/2.0"> [8
    <attDef ident="abbr" ns="http://www.fryske-akademy.org/linguistics/2.0"> [8 li
    <attDef ident="poss" ns="http://www.fryske-akademy.org/linguistics/2.0"> [8 li
    <attDef ident="reflex" ns="http://www.fryske-akademy.org/linguistics/2.0"> [8
    <attDef ident="prontype" ns="http://www.fryske-akademy.org/linguistics/2.0"> [
    <attDef ident="case" ns="http://www.fryske-akademy.org/linguistics/2.0">
      <desc>http://universaldependencies.org/u/feat/Case.html
        Case is usually an inflectional feature of nouns.
      </desc>
      <datatype>
        <dataRef key="teidata.enumerated"/>
      </datatype>
      <valList type="closed">
        <valItem ident="nom">
          <desc>nominative</desc>
        </valItem>
        <valItem ident="acc">
```

```
<elementSpec ident="w" module="analysis"  
  mode="change">  
  <classes mode="change">  
    <memberOf key="att.features" />  
  </classes>  
</elementSpec>  
<elementSpec ident="m" module="analysis"  
  mode="change">  
  <classes mode="change">  
    <memberOf key="att.linguistic" />  
  </classes>  
</elementSpec>  
<elementSpec ident="join" module="linking"  
  mode="change">  
  <classes mode="change">  
    <memberOf key="att.linguistic" />  
    <memberOf key="att.features" />  
  </classes>  
</elementSpec>
```

# More possibilities

- specGrp – specRef: grouping specs
- macroSpec – macroRef: expanding spec content
- model: define behaviour of elements

# ODD, processing



1. Maven: <https://bitbucket.org/fryske-akademy/online-dictionaries/src/master/pom.xml>
2. Oxygen
3. <https://oxgarage.tei-c.org/>
4. <https://roma.tei-c.org/>
5. Command line / maven
  1. <https://github.com/TEIC/Stylesheets/tags>

*2 – 4 use a version you may not want!*

# ODD, processing, maven

```
<transformationSet>
  <stylesheet>src/main/Stylesheets-${stylesheetversion}/odds/odd2odd.xsl</stylesheet>
  <parameters>...</parameters>
  <outputDir>src/main/resources/odd</outputDir>
  <fileMappers>...</fileMappers>
</transformationSet>
<transformationSet>
  <stylesheet>src/main/Stylesheets-${stylesheetversion}/odds/odd2relax.xsl</stylesheet>
  <parameters>...</parameters>
</transformationSet>
<transformationSet>
  <stylesheet>src/main/Stylesheets-${stylesheetversion}/odds/extract-isosch.xsl</stylesheet>
  <outputDir>src/main/resources/schematron</outputDir>
  <fileMappers>...</fileMappers>
</transformationSet>
```

```
<plugin>
  <groupId>net.sigmalab.trang</groupId>
  <artifactId>trang-maven-plugin</artifactId>
  <version>1.2</version>
```

```
1. <dependency>
  <groupId>name.dmaus.schxslt</groupId>
  <artifactId>java</artifactId>
  <version>2.0.3</version>
```

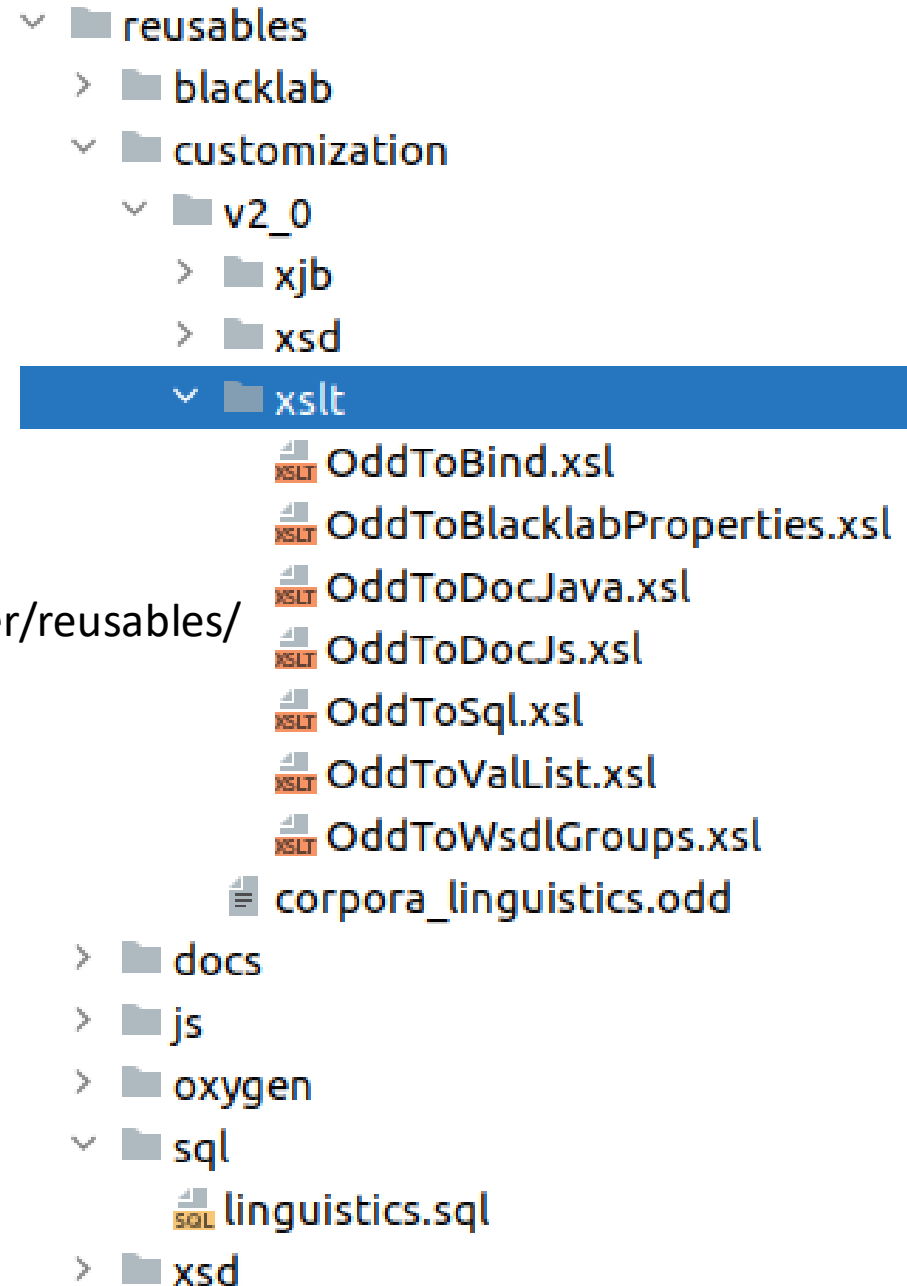
<https://bitbucket.org/fryske-akademy/online-dictionaries/src/master/pom.xml>



# ODD, generation

<https://search.maven.org/search?q=a:TeiLinguisticsFa>

<https://bitbucket.org/fryske-akademy/tei-encoding/src/master/reusables/>



# ODD, chaining

<http://teic.github.io/PDF/howtoChain.pdf>



- odd2odd.xsl
- @source=...

*Inherit from other odd's*

```
<schemaSpec ident="Bare-plus"
  source="tei_bare.compiled.odd" start="TEI">
  <moduleRef key="tei"/>
  <moduleRef key="header"/>
  <moduleRef key="gaiji"
    source="http://www.tei-c.org/release/xml/tei/odd/p5subset.xml"/>
  <moduleRef key="textstructure"/>
</schemaSpec>
```

# ODD, processing model

<https://tei-c.org/release/doc/tei-p5-doc/en/html/TD.html#TDPM>

<https://teipublisher.com>

<https://e-editiones.org/>

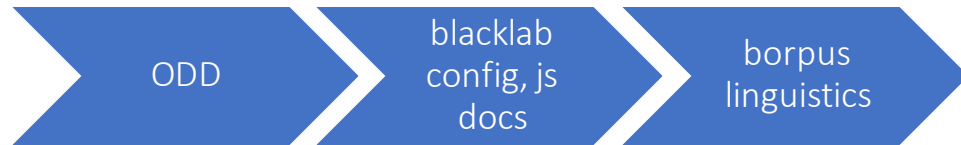
Very promising!

```
<elementSpec mode="change" ident="dateline">  
  <model behaviour="block">  
    <outputRendition>text-align: right;</outputRendition>  
  </model>  
</elementSpec>
```

```
<elementSpec ident="term" mode="add">  
  <model behaviour="inline" output="latex">  
    <param name="ref" value="substring-after(@ref, '#')"/>  
    <pb:template>\glslink{[[ref]]}{[[content]]}</pb:template>  
  </model>  
</elementSpec>
```

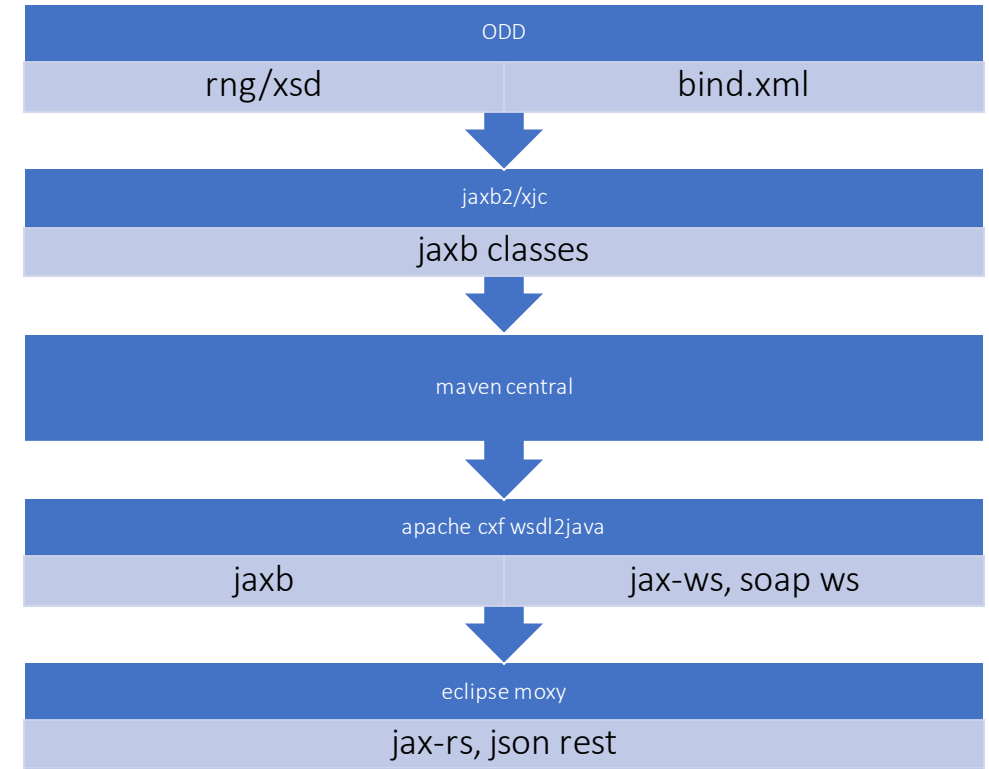
# Usage in applications

<https://web2.fa.knaw.nl/corpus-frontend>



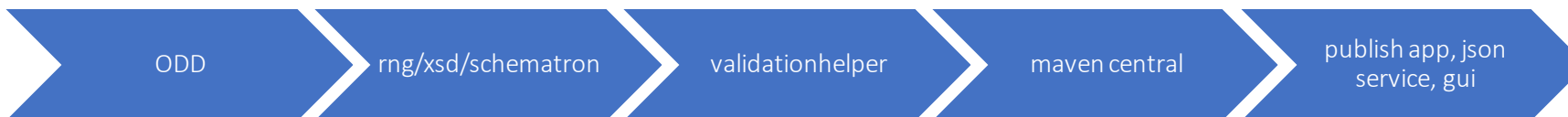
• corpora

<https://web2.fa.knaw.nl/foarkarswurdlist-ws/>



Frisian lexicon

<https://web2.fa.knaw.nl/exist/apps/onfw/index.html> (TEST!)



•dictionaries

# pros

- Reliable build processes that guarantee interoperability
- Maintain data logic in one place
- Generation of rng, schematron, xsd
- Generation using xslt
- Sticking close to TEI, benefit from updates and tools
- Limit knowledge and technologies to maintain

# cons

- Niche (complex) knowledge
- Stylesheets may not generate what you want
- Chaining (inheritance) can be confusing
- Hard to debug and test
- ODD change may cascade updates of libs and applications
- Xsd support (via trang) less stable than rng

# To ODD or not to ODD

- It is possible to maintain stable build processes based on ODD
  - With code generation
- Active community, active maintenance of stylesheets
- It is possible to build reusable libraries based on ODD
- Over the past 4 years little problems
- ODD syntax is rather simple
- ODD with teipublisher for digital editions and integration in blacklab

I would like odd to get a more prominent place in the TEI stack and community. It could be a well known goldmine

# Thanks

Eduard Drenth

[edrenth@fryske-akademy.nl](mailto:edrenth@fryske-akademy.nl)

