# TestXNG
## Testing XML Schemas and XSL Transformations

Filip van Laenen

`f.a.vanlaenen@ieee.org`

`@filipvanlaenen`

# TestXNG
## ~~Testing~~ Checking XML Schemas and XSL Transformations

Filip van Laenen

`f.a.vanlaenen@ieee.org`

`@filipvanlaenen`

# What's TestXNG?

- Test framework for:
  - XML Schema
  - XSL Transformations
- XML based
- Inspired by TestNG
- Java
- Open Source
  - http://sourceforge.net/projects/testxng

# Why TestXNG?

- No testing frameworks for XML Schemas
- No good testing frameworks for XSL Transformations
  - Jeni Tennison's XSLT Unit Testing Utilities
  - Juxy
  - UTF-X
  - XMLUnit
  - XSLTUnit

# Architecture

- TestXNG files define tests
- XML arguments
  - Sources and results
  - In-line or external
- Sensible default values and configurations
- Command line, Maven, plug-ins

# Example: Validation (1)

```xml
<Tests>
  ...
  <Test id="FooBarValidates">
    <Source>
      <Foo>
        <Bar>Lorem ipsum</Bar>
      </Foo>
    </Source>
    <Assert>
      <Validates>
        <Schema>FooBar.xsd</Schema>
      </Validates>
    </Assert>
  </Test>
  ...
</Tests>
```

# Example: Validation (2)

```
<Tests>

  ...

  <Test id="ExternalFooBarValidates">
    <SourceFile>FooBar.xml</SourceFile>
    <Assert>
      <Validates>
        <Schema>FooBar.xsd</Schema>
      </Validates>
    </Assert>
  </Test>

  ...

</Tests>
```

# Example: Non-validation

```xml
<Tests>
  ...
  <Test id="ExternalFooBazDoesNotValidate">
    <SourceFile>FooBaz.xml</SourceFile>
    <Assert>
      <DoesNotValidate>
        <Schema>FooBar.xsd</Schema>
      </DoesNotValidate>
    </Assert>
  </Test>
  ...
</Tests>
```

# Example: Transformation (1)

```
<Tests>
  ...
  <Test id="BasicTestOfAnXslTransformation">
    <Source><Foo><Bar>Lorem ipsum</Bar></Foo></Source>
    <Assert>
      <Transforms>
        <ExpectedResult>
          <Foo>A Bar with Lorem ipsum</Foo>
        </ExpectedResult>
        <Transformation>FooBar.xslt</Transformation>
      </Transforms>
    </Assert>
  </Test>
  ...
</Tests>
```

# Example: Transformation (2)

```xml
<Tests>
  ...
  <Test id="BasicTestOfAnXslTransformation">
    <SourceFile>FooBar.xml</SourceFile>
    <Assert>
      <Transforms>
        <ExpectedResultFile>
          FooBarTransformed.xml
        </ExpectedResultFile>
        <Transformation>FooBar.xslt</Transformation>
      </Transforms>
    </Assert>
  </Test>
  ...
</Tests>
```

# Root elements

```
<Tests>
  ...
  <Test id="BasicTestOfAnXslTransformation">
    <Source><Foo><Bar>Lorem ipsum</Bar></Foo></Source>
    <Assert>
      <Transforms>
        <AsRootElement/>
        <ExpectedResult>
          <Foo>A Bar with Lorem ipsum as root</Foo>
        </ExpectedResult>
        <Transformation>FooBar.xslt</Transformation>
      </Transforms>
    </Assert>
  </Test>
  ...
</Tests>
```

# Modes

```xml
<Tests>
  ...
  <Test id="BasicTestOfAnXslTransformation">
    <Source><Foo><Bar>Lorem ipsum</Bar></Foo></Source>
    <Assert>
      <Transforms>
        <Mode>Qux</Mode>
        <ExpectedResult>
          <Foo>A Bar with Lorem ipsum for Qux</Foo>
        </ExpectedResult>
        <Transformation>FooBar.xslt</Transformation>
      </Transforms>
    </Assert>
  </Test>
  ...
</Tests>
```

# Testing transformations

- Parameters and variables
- Direct calls to templates
- Keys
- Result types
  - XML, HTML, TXT, …

# When are two XML trees equal?

- XML elements and attributes
- Whitespace
  - **`<u>abc </u>def` ≠ `<u>abc</u> def`**
    - <u>abc </u>def ≠ <u>abc</u> def
- XML instructions
- XML comments

# Inspiration from TestNG

- Dependencies between tests
- Groups
  - Integration with TestNG?

# Effects of TestXNG (or TDD?)

- Modularization
  - Test hooks
- Readability
- Comprehensability

# Questions?
# Comments?

`http://sourceforge.net/projects/testxng`

`f.a.vanlaenen@ieee.org`
`@filipvanlaenen`