XSpec v0.5.0

Sandro Cirulli
Oxford University Press and XSpec

XML London 2017
Content

1 Introduction

2 XSpec 101

3 XSpec v0.5.0

4 Contribute to XSpec

5 Conclusion
About Me

- Lead Language Technologist at Oxford University Press
- Co-maintainer of XSpec
- Certified Jenkins Engineer
- Co-organizer of DevOps Oxford Meetup
What is XSpec?

- XSpec is an open source **unit test** and behaviour-driven development (BDD) framework for XSLT, XQuery, and Schematron

- **XSpec v0.5.0** was released in **January 2017**

- **XSpec v0.5.0** is **included in oXygen 19**

---

**XSpec**

Unit Test and Behaviour Driven Development (BDD) Framework for XSLT, XQuery, and Schematron

[https://github.com/xspec/xspec](https://github.com/xspec/xspec)
The Importance of Testing

- Testing forces developers to write smaller units of code
- Testing acts as self-documentation
- Testing reduces the risk of regression bugs in software releases
- Unit testing contributes to removing 30% of bugs [McConnell]
The Importance of Testing

From xkcd.com, revisited by Daniel Ballinger
OK, I’m convinced. But where to start testing?

- Functions
- Named Templates
- Any piece of code that get called more than once
XSpec 101

- Testing an XSLT template
- Testing an XQuery function
<x:description
    xmlns:x="http://www.jenitennison.com/xslt/xspec"
    stylesheet="title.xsl">

    <x:scenario label="When converting a title element">
        <x:context>
            <title>My Title</title>
        </x:context>

        <x:expect label="it should return a h1 element">
            <h1>My Title</h1>
        </x:expect>

    </x:scenario>

</x:description>
<xsl:stylesheet
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    exclude-result-prefixes="xs" version="2.0">

    <xsl:template match="title">
        <h2>
            <xsl:value-of select="."/>
        </h2>
    </xsl:template>

</xsl:stylesheet>
XSLT UNIT TEST REPORT

Stylesheet: home/sandro/Documents/programming/xml.com/code/title.xsl
Tested: 12 March 2017 at 16:25

Contents

- When converting a title element
  - it should return a h1 element

When Converting A Title Element

When converting a title element
- it should return a h1 element

Result

<table>
<thead>
<tr>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;h2&gt;My Title&lt;/h2&gt;</code></td>
</tr>
</tbody>
</table>

Expected Result

<table>
<thead>
<tr>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;h1&gt;My Title&lt;/h1&gt;</code></td>
</tr>
</tbody>
</table>
XSLT UNIT TEST REPORT

Stylesheet: home/sandro/Documents/programming/xml.com/title.xsl
Tested: 12 March 2017 at 14:49

Contents

When converting a title element

When Converting A Title Element

When converting a title element
✓ it should return a h1 element

Passed:1 / Pending:0 / Failed:0 / Total:1

passed:1 pending:0 failed:0 total:1

1 0 0 1

Success
module namespace functx = "http://www.functx.com";

declare function functx:capitalize-first ($arg as xs:string?) as xs:string? { 
    concat(upper-case(substring($arg, 1, 1)), substring($arg, 2))
};
<x:description xmlns:x="http://www.jenitennison.com/xslt/xspec"
xmlns:functx="http://www.functx.com"
query="http://www.functx.com"
query-at="xquery-tutorial.xq">

<x:scenario label="Calling function capitalize-first">

<x:call function="functx:capitalize-first">
  <x:param select="'hello'"/>
</x:call>

<x:expect label="should capitalize the first character of the string" select="'Hello'"/>

</x:scenario>
</x:description>
A Brief History of XSpec

- Jeni Tennison creates XSpec
- Jeni presents XSpec at XML Prague
- XSpec released as open source on Google Code
- Florent Georges releases v0.2
- Florent releases v0.3
- oXygen 14 ships with XSpec
- Florent releases v0.4.0-RC
- XSpec migrates to GitHub
- Active contributions start on GitHub
- XSpec v0.5.0
This release is a great news.

I don't even understand how a serious XSLT coder might develop without using your tool in 2017!

Best wishes for this project

In reply to Erik Wilde

Johan Mörén @hutchkintoot · 10 Feb 2016
@dret @anarchivist don't be fooled about the low version number. It's a wonderful tool. Every xsl should be backed by an xspec test.
New features:
- XSLT 3.0 support
- JUnit support
- Feature parity between Windows & MacOS/Linux
- Test suite for testing XSpec itself

Bug fixes:
- Code coverage
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    exclude-result-prefixes="xs" version="3.0">

<xsl:template name="supportXPath3">
    <root>
        <question>
            <xsl:text>Does XSpec support XPath 3.0?</xsl:text>
        </question>
        <answer>
            <xsl:value-of select="let $answer := 'Yes it does' return $answer"/>
        </answer>
    </root>
</xsl:template>
</xsl:stylesheet>
<x:description
xmlns:x="http://www.jenitennison.com/xslt/xspec"
stylesheet="xspec-xslt3.xsl"
xslt-version="3.0">
<x:scenario label="When testing the inline function expression in XPath 3">
<x:call template="supportXPath3"/>

<x:expect label="it returns the expected answer">
<root>
  <question>Does XSpec support XPath 3.0?</question>
  <answer>Yes it does</answer>
</root>
</x:expect>
</x:scenario>
</x:description>
JUnit Support

- JUnit is a **popular unit testing framework for Java**
- JUnit is **understood natively by CI servers** like Jenkins
- Generate XSpec reports in JUnit with:

  /bin/xspec.sh -j tutorial/escape-for-regex.xspec
JUnit Support
Jenkins CI
XSpec is **tested itself** with a mix of shell and batch scripts and XSpec tests.

The **test suite runs automatically** at each pull request and code merge.

Tests runs on **Travis CI** and **AppVeyor**.

Documentation available on the wiki to **integrate automatic XSpec test suites in GitHub projects**.
Bonus Feature: Schematron Support

- Vincent Lizzi added **Schematron support in XSpec**

- The feature is **already available in the master branch** and will be included in XSpec 0.6.0

- **Documentation and tutorials available** on the wiki

- **Try it out!**
<?xml version="1.0"?>
<article>
  <title>Example</title>
</article>
<sch:schema
    xmlns:sch="http://purl.oclc.org/dsdl/schematron"
    queryBinding="xslt2">

    <sch:pattern>
        <sch:rule context="article">
            <sch:assert test="title" id="a001">
                article should have a title
            </sch:assert>
        </sch:rule>
    </sch:pattern>

</sch:schema>
<x:description
  xmlns:x="http://www.jenitennison.com/xslt/xspec"
schematron="demo-01.sch">
  <x:scenario label="Given an XML file with an article">
    <x:context href="demo-01.xml"/>
    <x:scenario label="article should have a title">
      <x:expect-not-assert id="a001"/>
    </x:scenario>
  </x:scenario>
</x:description>
Contribute to XSpec

- Checkout out [github.com/xspec/xspec](https://github.com/xspec/xspec) and star it
- Raise **issues**, send **pull requests**, contribute to tests, documentation, and **code reviews**
- **Integrate XSpec** test suites in your projects
- **Spread the word!**
XSpec is an open source unit test and behaviour-driven development (BDD) framework for XSLT, XQuery, and Schematron.

Check out new features in XSpec v0.5.0.

Use and contribute to XSpec on github.com/xspec/xspec.
Acknowledgements

Jeni Tennison

Tony Graham

Florent Georges

XSpec Community
Thank you for your attention!

**Contact:**
sandro.cirulli@oup.com
sandrocirulli.net/contact

**Slides:**
sandrocirulli.net/xml-london-2017

**XSpec:**
github.com/xspec/xspec
github.com/xspec/xspec/wiki