



A Data Usability Company
ANTENNA HOUSE

Validating XSL-FO With Relax NG and Schematron

Tony Graham
XML Division
Antenna House, Inc.
tgraham@antenna.co.jp
tony@antennahouse.com
[@tgraham_antenna](https://twitter.com/tgraham_antenna)

28 MB FO file

198,995 elements

711,185 attributes

Most attributes parsed as expressions

829 Relax NG patterns

611 Schematron <assert> and <report>

How long does it take to validate in the editor?

28 MB FO file

198,995 elements

711,185 attributes

829 Relax NG patterns

611 Schematron <assert> and <report>

Most attributes parsed as expressions

How long does it take to validate in the editor?

test.fo [C:\projects\oxygen\focheck\test\samples\test.fo] - <oXygen/> XML Editor

File Edit Find Project Options Tools Document Window Help

XPath 2.0 Execute XPath on 'Current File'

• test.fo x

```
<?xml version="1.0" encoding="UTF-8"?>
<fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format"
  margin-right="12mm" margin-bottom="20mm" margin-left="22mm" overflow="error-if-overflow"/><fo:region-before
  ePageBeforeFirstOdd" extent="45mm" display-align="after" precedence="false"/><fo:region-end
  ePageEndFirstOdd" extent="12mm" display-align="before"/><fo:region-after region-name="rgnTitlePageAfterFirstOdd" extent="20mm"
  " marks" not allowed here; expected attribute "allowed-height-scale", "allowed-width-scale", "auto-restore", "axf:abbreviation-character-count", "axf:alt-glyph", "axf:assumed-page-n...
```

テキスト

結果

情報 概要 - 48214 項目

リソース

- ▲ _pub_manual_AH.fo. スキーマ “[axfrng](#)”, シナリオ “[XSL 1.1 with Antenna House extensions](#)”, 文書タイプ “[XSL 1.1 with Antenna House extensions](#)” (48214 項目)
- E [Jing] attribute “axf:printer-marks” not allowed here; expected attribute “allowed-height-scale”, “allowed-width-scale”, “auto-restore”, “axf:abbreviation-character-count”, “axf:alt-glyph”, “axf:assumed-page-number”, “axf:crop-area-visibility”, “axf:crop-offset-bottom”, “axf:crop-offset-left”, “axf:crop-offset-right”, “axf:crop-offset-top”, “axf:font-size”, “axf:font-style”, “axf:font-weight”, “axf:height”, “axf:line-height”, “axf:page-height”, “axf:page-width”, “axf:width”, “axf:zero-margin”, “axf:zoom”, “axf:z-index”, “axf:z-space”
 - E [Jing] attribute “axf:printer-marks-line-color” not allowed here; expected attribute “allowed-height-scale”, “allowed-width-scale”, “auto-restore”, “axf:abbreviation-character-count”, “axf:alt-glyph”, “axf:assumed-page-number”, “axf:crop-area-visibility”, “axf:crop-offset-bottom”, “axf:crop-offset-left”, “axf:crop-offset-right”, “axf:crop-offset-top”, “axf:font-size”, “axf:font-style”, “axf:font-weight”, “axf:height”, “axf:line-height”, “axf:page-height”, “axf:page-width”, “axf:width”, “axf:zero-margin”, “axf:zoom”, “axf:z-index”, “axf:z-space”
 - E [Jing] attribute “axf:printer-marks-line-width” not allowed here; expected attribute “allowed-height-scale”, “allowed-width-scale”, “auto-restore”, “axf:abbreviation-character-count”, “axf:alt-glyph”, “axf:assumed-page-number”, “axf:crop-area-visibility”, “axf:crop-offset-bottom”, “axf:crop-offset-left”, “axf:crop-offset-right”, “axf:crop-offset-top”, “axf:font-size”, “axf:font-style”, “axf:font-weight”, “axf:height”, “axf:line-height”, “axf:page-height”, “axf:page-width”, “axf:width”, “axf:zero-margin”, “axf:zoom”, “axf:z-index”, “axf:z-space”
 - E [Jing] attribute “axf:crop-area-visibility” not allowed here; expected attribute “allowed-height-scale”, “allowed-width-scale”, “auto-restore”, “axf:abbreviation-character-count”, “axf:alt-glyph”, “axf:assumed-page-number”, “axf:crop-offset-bottom”, “axf:crop-offset-left”, “axf:crop-offset-right”, “axf:crop-offset-top”, “axf:font-size”, “axf:font-style”, “axf:font-weight”, “axf:height”, “axf:line-height”, “axf:page-height”, “axf:page-width”, “axf:width”, “axf:zero-margin”, “axf:zoom”, “axf:z-index”, “axf:z-space”
 - E [Jing] attribute “axf:bleed” not allowed here; expected attribute “allowed-height-scale”, “allowed-width-scale”, “auto-restore”, “axf:abbreviation-character-count”, “axf:alt-glyph”, “axf:assumed-page-number”, “axf:crop-area-visibility”, “axf:crop-offset-bottom”, “axf:crop-offset-left”, “axf:crop-offset-right”, “axf:crop-offset-top”, “axf:font-size”, “axf:font-style”, “axf:font-weight”, “axf:height”, “axf:line-height”, “axf:page-height”, “axf:page-width”, “axf:width”, “axf:zero-margin”, “axf:zoom”, “axf:z-index”, “axf:z-space”
 - E [Jing] attribute “axf:printer-marks-zero-margin” not allowed here; expected attribute “allowed-height-scale”, “allowed-width-scale”, “auto-restore”, “axf:abbreviation-character-count”, “axf:alt-glyph”, “axf:assumed-page-number”, “axf:crop-area-visibility”, “axf:crop-offset-bottom”, “axf:crop-offset-left”, “axf:crop-offset-right”, “axf:crop-offset-top”, “axf:font-size”, “axf:font-style”, “axf:font-weight”, “axf:height”, “axf:line-height”, “axf:page-height”, “axf:page-width”, “axf:width”, “axf:zero-margin”, “axf:zoom”, “axf:z-index”, “axf:z-space”
 - E [Jing] attribute “axf:crop-offset-top” not allowed here; expected attribute “allowed-height-scale”, “allowed-width-scale”, “auto-restore”, “axf:abbreviation-character-count”, “axf:alt-glyph”, “axf:assumed-page-number”, “axf:crop-area-visibility”, “axf:crop-offset-bottom”, “axf:crop-offset-left”, “axf:crop-offset-right”, “axf:font-size”, “axf:font-style”, “axf:font-weight”, “axf:height”, “axf:line-height”, “axf:page-height”, “axf:page-width”, “axf:width”, “axf:zero-margin”, “axf:zoom”, “axf:z-index”, “axf:z-space”

エラー x

```
17      </fo:flow>
18  </fo:page-sequence>
19 </fo:root>
20
```

E [ISO Schematron] 'font-size' should be EnumerationToken, Length, or Percent. '#aabbcc' is a Color.

Text Grid Author

28 MB FO file

198,995 elements

711,185 attributes

829 Relax NG patterns

611 Schematron <assert> and <report>

Most attributes parsed as expressions

How long does it take to validate in the editor?

Element = Formatting Object

“The vocabulary of formatting objects supported by XSL - the set of **fo:** element types - represents the set of typographic abstractions available to the designer.”

- <fo:block>
- <fo:footnote>
- <fo:wrapper>
- <fo:instream-foreign-object>
- <fo:change-bar-begin />

28 MB FO file

198,995 elements

711,185 attributes

Most attributes parsed as expressions

829 Relax NG patterns

611 Schematron <assert> and <report>

How long does it take to validate in the editor?

Attribute = Property

“The properties associated with an instance of a formatting object control the formatting of that object.”

- ‘color’
- ‘font-size’
- ‘master-name’
- ‘hyphenation-push-character-count’
- ‘relative-align’

28 MB FO file

198,995 elements

711,185 attributes

Most attributes parsed as expressions

829 Relax NG patterns

611 Schematron <assert> and <report>

How long does it take to validate in the editor?

Property value expressions

“All property value specifications in attributes within an XSL stylesheet can be expressions. ... The expression is first evaluated and then the resultant value is used to determine the value of the property.”

- font-size="12pt"
- start-indent=" 1em * 1.3 * (1 - 1)"
- end-indent="label-end()"

28 MB FO file

198,995 elements

711,185 attributes

Most attributes parsed as expressions

829 Relax NG patterns

611 Schematron <assert> and <report>

How long does it take to validate in the editor?

Relax NG schema

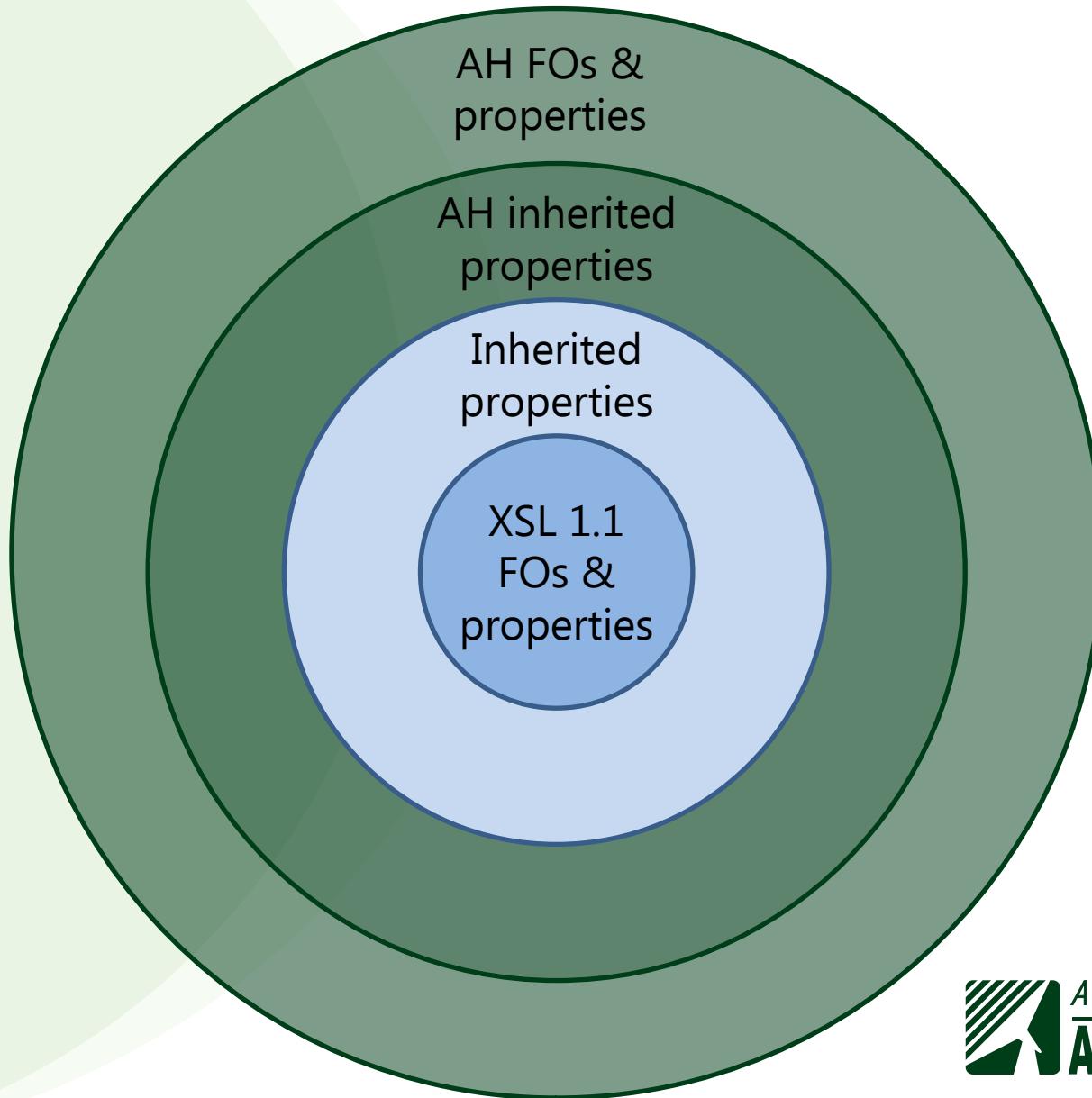
- Using compact syntax:
 - Similar to model syntax in XSL 1.1 spec
 - Easy to prototype
 - Easier to write AH extensions
- Translated to:
 - XML syntax for use with oXygen
 - W3C XML Schema

Validate unto Caesar...

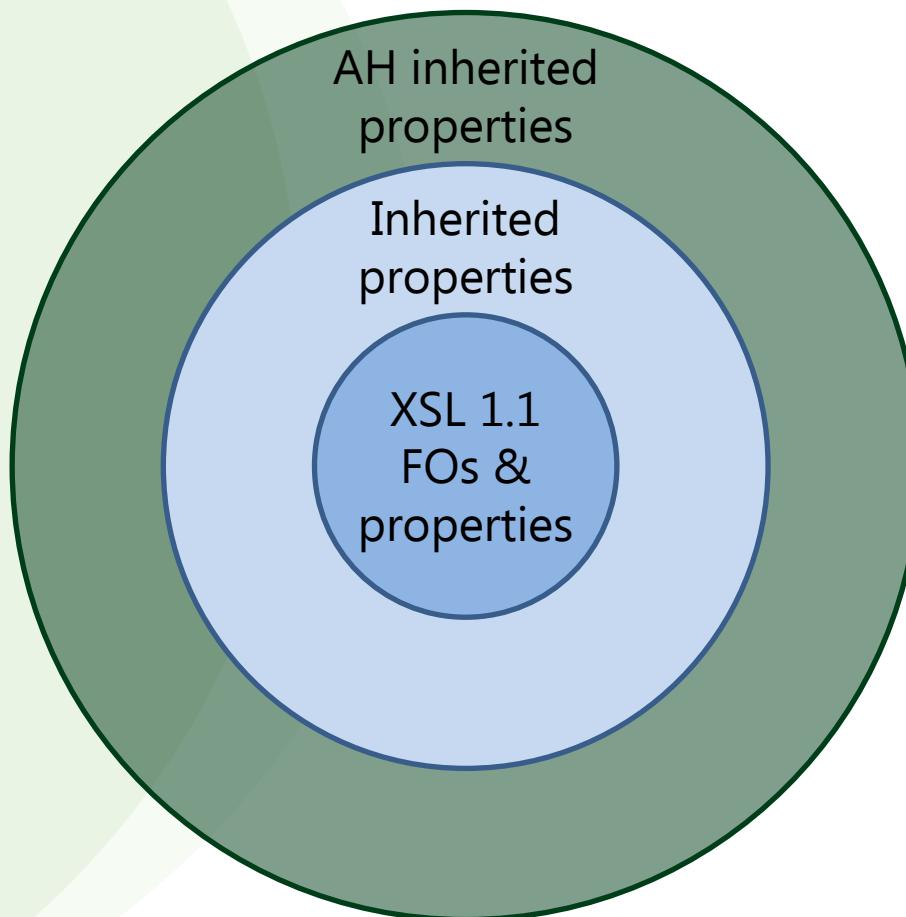
Relax NG used for checking:

- Element structure
 - Required attributes
 - <id> and <idref> FO datatypes
 - 'xml:lang' value
-
- All other attributes are '{ text }?'

Relax NG Structure



Generated from XSL 1.1 Spec XML



Relax NG from XSL 1.1 Spec XML

```
<div3 id="fo_block"><head>fo:block</head>

<p><emph>Common Usage:</emph></p>
<p>The fo:block formatting object is commonly
used for formatting paragraphs,
titles, headlines, figure and table captions,
etc.</p>
...
<p><emph>Contents:</emph></p>
<eg xml:space="preserve">
(#PCDATA|<loc
  href="#inline.fo.list">%inline;</loc>|<loc
  href="#block.fo.list">%block;</loc>)*
</eg>
```

Seems pretty simple...

```
fo_block.model =  
(text|inline.fo.list|block.fo.list)*
```

```
inline.fo.list =  
    fo_bidi-override  
    | fo_character  
    | fo_external-graphic  
    | fo_instream-foreign-object  
    | fo_inline  
    | fo_inline-container  
    | fo_leader  
    | fo_page-number  
    | fo_page-number-citation  
    | fo_page-number-citation-last  
    | fo_scaling-value-citation  
    | fo_basic-link  
    | fo_multi-toggle  
    | fo_index-page-citation-list
```

```
block.fo.list =  
    fo_block  
    | fo_block-container  
    | fo_table-and-caption  
    | fo_table  
    | fo_list-block
```

But...

- “The following formatting objects are "neutral" containers and may be used... anywhere where #PCDATA, %block;, or %inline; are allowed: ...”
- “The following formatting objects define "points" and may be used anywhere as a descendant of fo:flow or fo:static-content: ...”
- “The following "out-of-line" formatting objects may be used anywhere where #PCDATA, %block;, or %inline; are allowed (except as a descendant of any "out-of-line" formatting object): ...”

fo:block patterns

```
fo_block =  
## The fo:block formatting object is commonly  
used for formatting paragraphs, titles,  
headlines, figure and table captions, etc.  
element block {  
    fo_block.attlist,  
    fo_block.model  
}  
  
fo_block.model =  
    fo_marker*,  
    fo_initial-property-set?,  
    ( (text|inline.fo.list|block.fo.list |  
neutral.fo.list)* & (out-of-line.fo.list)* ) &  
(point.fo.list)* )
```

fo:block patterns

```
fo_block.attlist =  
    common-accessibility-properties,  
    ...  
    common-margin-properties-block,  
    break-after,  
    break-before,  
    clear,  
    ...  
    white-space-treatment,  
    widows,  
    wrap-option
```

fo:wrapper

“The fo:wrapper formatting object is used to specify inherited properties for a group of formatting objects.”

“An fo:wrapper is only permitted to have children that would be permitted to be children of the parent of the fo:wrapper”

fo:wrapper Relax NG

```
neutral.fo.list =
```

```
...
```

```
| fo_wrapper
```

```
neutral.fo.list.block =
```

```
...
```

```
| fo_wrapper.block
```

```
neutral.fo.list.inline =
```

```
...
```

```
| fo_wrapper.inline
```

fo:wrapper Relax NG

```
fo_wrapper.block =
## The fo:wrapper formatting object is used to specify inherited
properties for a group of formatting objects.
element wrapper {
  fo_wrapper.attlist,
  fo_wrapper.model.block
}

fo_wrapper.model.block =
  fo_marker*,
  ( ( ( (text|block.fo.list)* | neutral.fo.list.block)* ) &
(point.fo.list)* )

...
fo_wrapper.model.inline =
  fo_marker*,
  ( ( ( (text|inline.fo.list)* | neutral.fo.list.inline)* &
(inline.out-of-line.fo.list)* ) &
(point.fo.list)* )
```

28 MB FO file

198,995 elements

711,185 attributes

Most attributes parsed as expressions

829 Relax NG patterns

611 Schematron <assert> and <report>

How long does it take to validate in the editor?

Schematron

- Multiple phases
 - FO structural rules
 - Property values
 - Autogenerated
 - Antenna House extensions

Autogenerated property Schematron

7.27.2 "column-count"

XSL Definition:

Value: <number> | inherit

```
<!-- column-count -->
<!-- <number> | inherit -->
<!-- Inherited: no -->
<!-- Shorthand: no -->
<rule context="fo:*/@column-count">
  <let name="expression" value="ahf:parser-runner(.)"/>
  <assert test="local-name($expression) = ('Number',
    'EnumerationToken', 'ERROR', 'Object')">'column-count' should be
    Number or EnumerationToken. '<value-of select="."/>' is a <value-
    of select="local-name($expression)"/>.</assert>
  <report test="$expression instance of element(EnumerationToken) and
    not($expression/@token = ('inherit'))">'column-count' enumeration
    token is '<value-of select="$expression/@token"/>'. Token should
    be 'inherit'.</report>
  <report test="local-name($expression) = 'ERROR'">Syntax error:
    'column-count=<value-of select="."/>'"</report>
</rule>
```

EBNF used with REx

```
Expression      ::= Expr EOF
Expr           ::= AdditiveExpr
AdditiveExpr   ::= MultiplicativeExpr (( '+' | '-' ) MultiplicativeExpr)*
MultiplicativeExpr ::= UnaryExpr (( '*' | 'div' | 'mod' ) UnaryExpr)*
UnaryExpr       ::= PrimaryExpr
                  | '-' UnaryExpr
PrimaryExpr    ::= '(' Expr ')'
                  | Numeric
                  | Literal
                  | Color
                  | EnumerationToken
                  | FunctionCall
FunctionCall   ::= FunctionName '(' ( Expr ( ',' Expr)* )? ')'
Numeric         ::= AbsoluteNumeric
                  | RelativeNumeric
AbsoluteNumeric ::= AbsoluteLength
                  ::= Number AbsoluteUnitName?
AbsoluteLength  ::= Number AbsoluteUnitName?
AbsoluteUnitName ::= ('cm' | 'mm' | 'in' | 'pt' | 'pc' | 'px'
                     | 'ex' | 'jpt' | 'q' | 'dpi' | 'dd' | 'cc'
                     | 'rem' | 'ch' | 'wch' | 'lh' | 'rlh' | 'vw'
                     | 'vh' | 'vmin' | 'vmax' | 'pvw' | 'pvh'
                     | 'pvmin' | 'pvmax' | 'gr')
RelativeNumeric ::= Percent
                  | RelativeLength
Percent         ::= Number '%'
RelativeLength  ::= Number RelativeUnitName
RelativeUnitName ::= 'em'
Color           ::= '#' ('CMYK' | 'Grayscale' | 'Registration' | 'Separation' | AlphaOrDigits)
AlphaOrDigits   ::= NCName
EnumerationToken ::= NCName
...
```

$$-1 - -2 = ?$$

```
<Expression>
  <Expr>
    <AdditiveExpr>
      <MultiplicativeExpr>
        <UnaryExpr>
          <TOKEN>-</TOKEN>
        <UnaryExpr>
          <PrimaryExpr>
            <Numeric>
              <AbsoluteNumeric>
                <AbsoluteLength>
                  <Number>1</Number>
                </AbsoluteLength>
              </AbsoluteNumeric>
            </Numeric>
          </PrimaryExpr>
        </UnaryExpr>
      </UnaryExpr>
    </MultiplicativeExpr>
  <TOKEN>-</TOKEN>
```

```
→ <MultiplicativeExpr>
  <UnaryExpr>
    <TOKEN>-</TOKEN>
  <UnaryExpr>
    <PrimaryExpr>
      <Numeric>
        <AbsoluteNumeric>
          <AbsoluteLength>
            <Number>2</Number>
          </AbsoluteLength>
        </AbsoluteNumeric>
      </Numeric>
    </PrimaryExpr>
  </UnaryExpr>
  </UnaryExpr>
</MultiplicativeExpr>
</AdditiveExpr>
</Expr>
<EOF/>
</Expression>
```

$$-1 - -2 = ?$$

```
<Number value="1"  
       is-positive="yes"  
       is-zero="no"/>
```

- Parser result reduced to FO datatype
- `parser-runner.xsl` uses only `xs1:function`
 - Using `xs1:template` caused wrong location in oXygen error reporting

Handwritten structure checks

```
<rule context="fo:footnote">
  <!-- http://www.w3.org/TR/xsl/#d0e6532 -->
  <report test="(for $ancestor in ancestor::fo:* return local-
    name($ancestor)) = ('float', 'footnote')">An 'fo:footnote' is
    not allowed as a descendant of 'fo:float' or
    'fo:footnote'.</report>
  <!-- http://www.w3.org/TR/xsl/#fo_footnote -->
  <report test="exists(ancestor::fo:block-container[@absolute-
    position = ('absolute', 'fixed')])">An 'fo:footnote' is not
    permitted as a descendant of an 'fo:block-container' that
    generates an absolutely positioned area.</report>
  <report test="exists(descendant::fo:block-container[@absolute-
    position = ('absolute', 'fixed')])">An 'fo:footnote' is not
    permitted to have as a descendant an 'fo:block-container' that
    generates an absolutely positioned area.</report>
  <report test="exists(descendant::fo:[local-name() = ('float',
    'footnote', 'marker')])">An 'fo:footnote' is not permitted to
    have an 'fo:float', 'fo:footnote', or 'fo:marker' as a
    descendant.</report>
</rule>
```

Handwritten property check

“If a non-positive or non-integer value is provided, the value will be rounded to the nearest integer value greater than or equal to 1.”

```
<rule context="fo:*/@column-count | fo:*/@number-columns-  
spanned">  
  <let name="expression" value="ahf:parser-runner(.)"/>  
  <report test="local-name($expression) = 'Number' and  
            ($expression/@is-positive) and  
            ($expression/@is-positive eq 'no' or  
             $expression/@is-zero = 'yes' or  
             exists($expression/@value) and  
             not($expression/@value castable as xs:integer))"  
        role="column-count">Warning: '<value-of select="local-  
name()" />' should be a positive integer. The FO  
formatter will round a non-positive or non-integer  
value to the nearest integer value greater than or  
equal to 1.</report>  
</rule>
```

When you've seen one “12pt”, you've seen them all

- Memorise XSL datatype result from every expression

```
<!-- ahf:parser-runner($input as xs:string) as element()+ -->
<!-- Runs the REx-generated parser on $input then reduces the parse
tree to a XSL 1.1 datatype. Uses @saxon:memo-function extension
to memorize return values (when used with Saxon PE or Saxon EE)
to avoid reparsing the same strings again and again when this is
used as part of validating an entire XSL-FO document. -->
<xsl:function name="ahf:parser-runner" as="element()+
    saxon:memo-function="yes"
    xmlns:saxon="http://saxon.sf.net/" >
    <xsl:param name="input" as="xs:string" />

    <xsl:sequence
        select="ahf:parser-runner2($input, false())"/>
</xsl:function>
```

Testing

- **XSpec** for parser-runner.xsl
- **stf** for Schematron
 - <https://github.com/MenteaXML/stf>

```
<?stf column-count:1 ?>  
<fo:region-body column-count="-1"  
xmlns:fo="http://www.w3.org/1999/XSL/  
Format" />
```

28 MB FO file

198,995 elements

711,185 attributes

Most attributes parsed as expressions

829 Relax NG patterns

611 Schematron <assert> and <report>

How long does it take to validate in the editor?



XPath 2.0 Execute XPath on 'Current File'



[08:36:11] start scanning file:/C:/projects/oxygen/focheck/test/samples/_pub_manual_AH.fo with schema(s): file:/C:/projects/oxygen/focheck/schema/rng/axf.rng
[08:36:12] - Found 0 problem(s)
[08:36:12] - Schematron Error Scanner - start scanning
file:/C:/projects/oxygen/focheck/test/samples/_pub_manual_AH.fo with schema(s): file:/C:/projects/oxygen/focheck/schematron/axf.sch
[08:36:17] Found 1 problem(s)

```

<x?xml version="1.0" encoding="UTF-8"?><fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:axf="http://www.antennahouse.com/names/XSL/Extensions" font-size="11pt"
  line-height="normal" xml:lang="en" font-family="Times New Roman"
  line-stacking-strategy="line-height" text-align="start" axf:line-break="normal"
  color="black" hyphenate="false" axf:hyphenation-minimum-character-count="1"
  hyphenation-keep="column"><fo:layout-master-set><fo:simple-page-master
  master-name="pmsCover" page-width="210mm" page-height="297mm" margin-top="0mm"
  margin-right="0mm" margin-bottom="0mm" margin-left="0mm" axf:printer-marks="crop"
  axf:printer-marks-line-color="#000000" axf:printer-marks-line-width="0.25pt"
  axf:crop-area-visibility="visible" axf:bleed="5mm" axf:printer-marks-zero-margin="0mm"
  axf:crop-offset-top="33.5mm" axf:crop-offset-bottom="33.5mm" axf:crop-offset-left="23.5mm"
  axf:crop-offset-right="23.5mm"><fo:region-body margin-top="0mm" margin-right="0mm"
  margin-bottom="0mm" margin-left="0mm" overflow="error-if-overflow"/><fo:region-after
  extent="0mm" display-align="before" precedence="false"/><fo:region-start
  region-name="rgnCoverPageStart" extent="40mm" display-align="after"/><fo:region-end
  region-name="rgnCoverPageEnd" extent="29mm + 10mm + 5mm"
  display-align="after"/></fo:simple-page-master><fo:simple-page-master
  master-name="pmsTitlePageFirstOdd" page-width="210mm" page-height="297mm" margin-top="0mm"
  margin-right="0mm" margin-bottom="0mm" margin-left="0mm" axf:crop-area-visibility="visible"
  axf:bleed="5mm" axf:printer-marks-zero-margin="0mm" axf:crop-offset-top="33.5mm"
  axf:crop-offset-bottom="33.5mm" axf:crop-offset-left="23.5mm" axf:crop-offset-right="23.5mm"
  display-align="after"/>
```

E [ISO Schematron] Syntax error: 'content-width=""'

Text Grid Author

Info Description - 1 item

▲ _pub_manual_AH.fo, scenario "XSL 1.1 with Antenna House extensions", document type "XSL 1.1 with Antenna House extensions" (1 item)

E [ISO Schematron] Syntax error: 'content-width=""'

Summary

<https://github.com/AntennaHouse/focheck>

focheck validates:

- XSL 1.1 Formatting Objects and properties
- Antenna House AH Formatter extensions
- Property value expressions
- Quickly!

Pull requests and issues are welcome

Questions?



A Data Usability Company
ANTENNA HOUSE

Moore's Law is your friend

- 2004
 - RenderX paper at XML Europe
 - “XSLT is powerful but requires more resources than DTD validation”
 - Relax NG schema usable in editor
- 2015
 - Relax NG, Schematron, and XSLT in editor
- The difference?
 - Faster processors