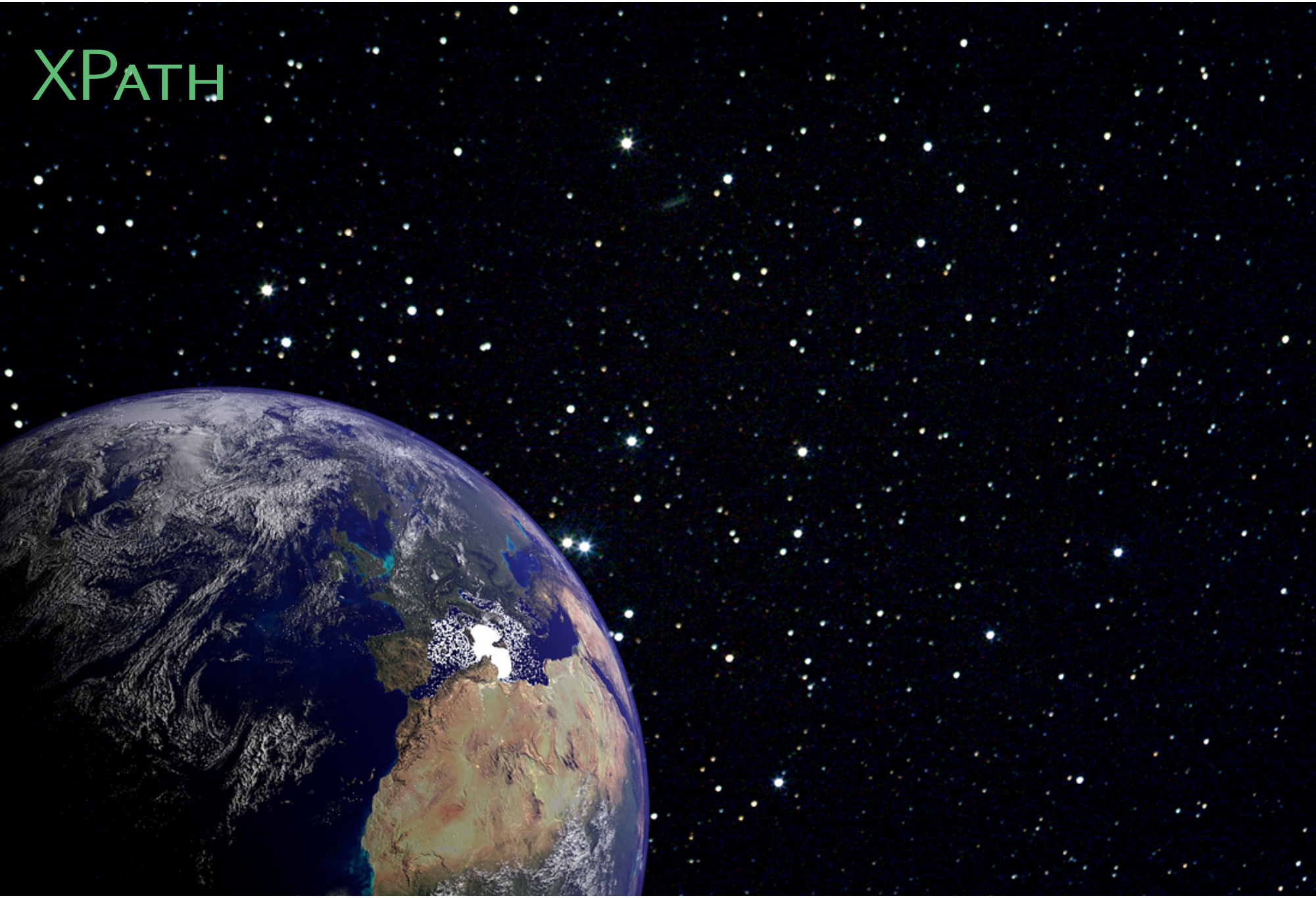
XPath Queries in the Real World

David Baelde Anthony Lick Sylvain Schmitz

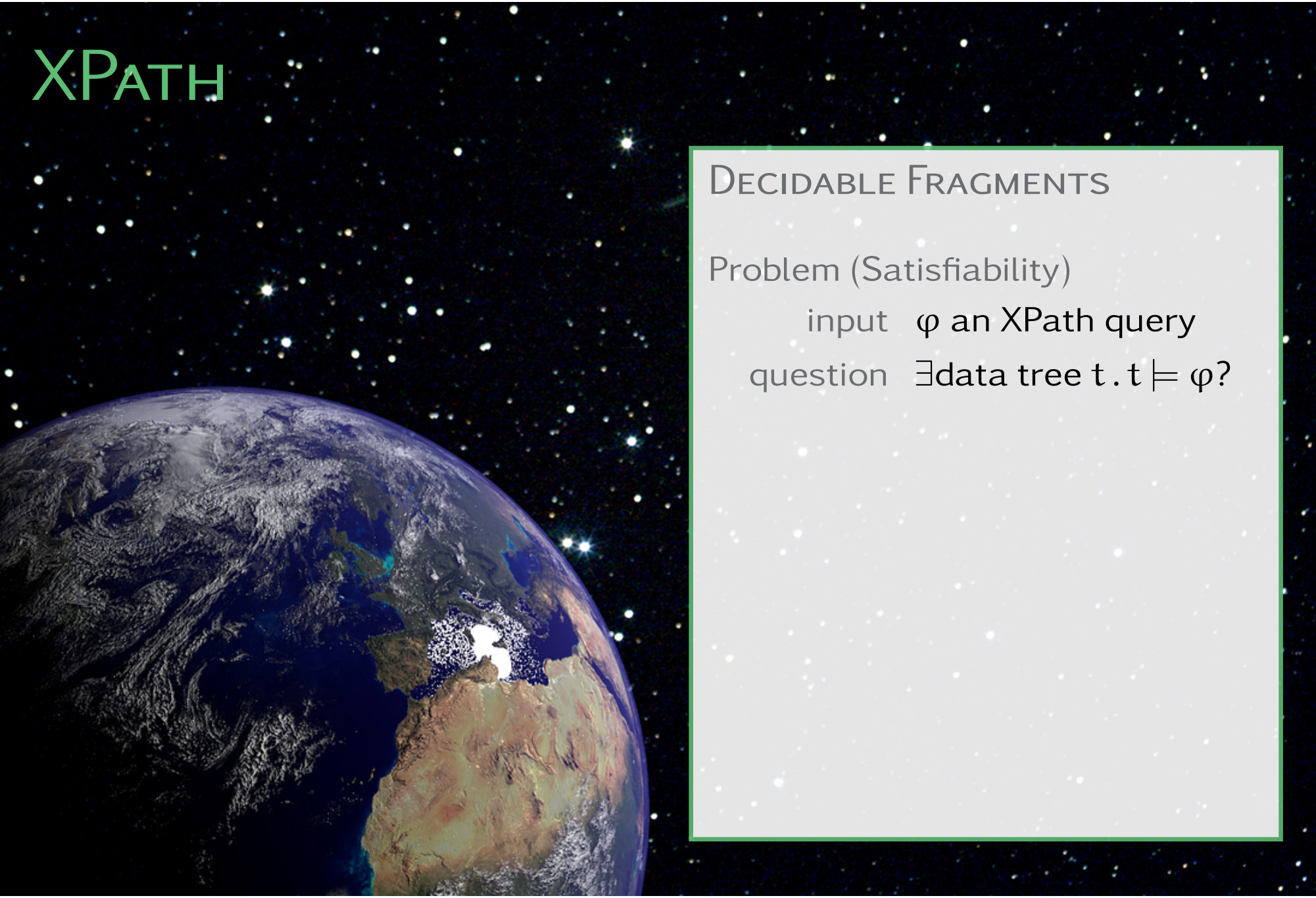
lsv
école normale supérieure paris–saclay
cnrs advancing the frontiers
université PARIS-SACLAY
IUF institut universitaire de France

PODS, July 2, 2019, Amsterdam

XPath



XPath



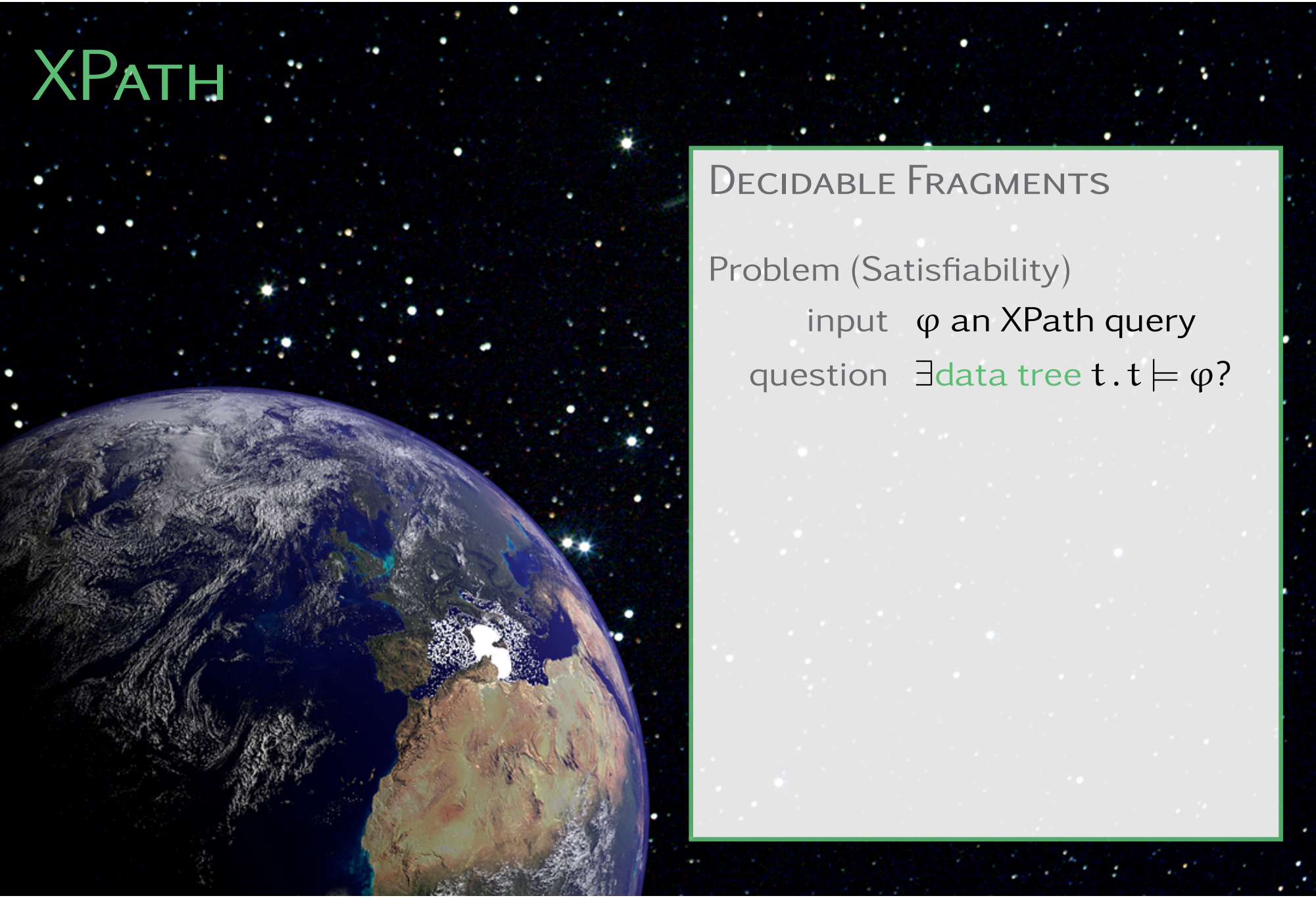
Decidable Fragments

Problem (Satisfiability)

input φ an XPath query

question $\exists$data tree t . $t \models \varphi$?

XPath



Decidable Fragments

Problem (Satisfiability)

input φ an XPath query

question $\exists$data tree t . $t \models \varphi$?

XPath

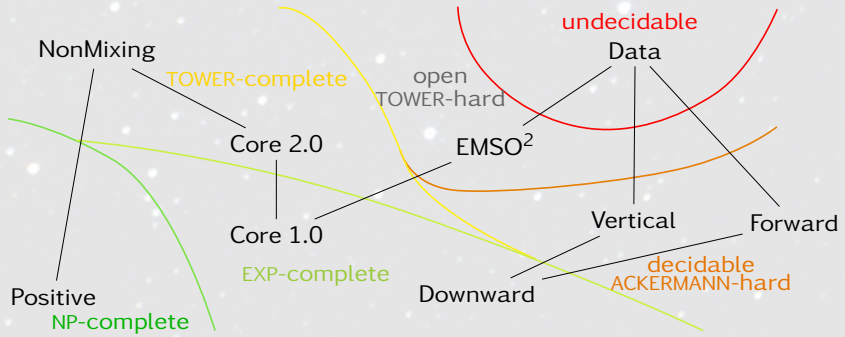
Decidable Fragments

Problem (Satisfiability)

input φ an XPath query

question $\exists$data tree t . $t \models \varphi$?

Expressiveness/Complexity



XPath

XML Path Language (XPath) 3.1

W3C Recommendation 21 March 2017

This version:
https://www.w3.org/TR/2017/REC-xpath-31-20170321/

Latest version of XPath 3.1:
https://www.w3.org/TR/xpath-31/

Previous versions of XPath 3.1:
https://www.w3.org/TR/2017/PR-xpath-31-20170117/
https://www.w3.org/TR/2016/CR-xpath-31-20161213/
https://www.w3.org/TR/2015/CR-xpath-31-20151217/
https://www.w3.org/TR/2014/CR-xpath-31-20141218/
https://www.w3.org/TR/2014/WD-xpath-31-20141007/
https://www.w3.org/TR/2014/WD-xpath-31-20140424/

Most recent version of XPath 3:
https://www.w3.org/TR/xpath-3/

Most recent version of XPath:
https://www.w3.org/TR/xpath/

Most recent Recommendation of XPath:
https://www.w3.org/TR/2014/REC-xpath-30-20140408/

Editors:
Jonathan Robie, biblicalhumanities.org <jonathan.robie@biblicalhumanities.org>
Michael Dyck, Invited Expert <jmdyck@ibiblio.org>
Josh Spiegel, Oracle Corporation <josh.spiegel@oracle.com>

Please check the **errata** for any errors or issues reported since publication.

See also **translations**.

This document is also available in these non-normative formats: XML and Change markings relative to previous edition.

Abstract

XPath 3.1 is an expression language that allows the processing of values conforming to the data model defined in [XQuery and XPath Data Model (XDM) 3.1]. The name of the language derives from its most distinctive feature, the path expression, which provides a means of hierarchic addressing of the nodes in an XML tree. As well as modeling the tree structure of XML, the data model also includes atomic values, function items, and sequences. This version of XPath supports JSON as well as XML, adding maps and arrays to the data model and supporting them with new expressions in the language and new functions in [XQuery and XPath Functions and Operators 3.1]. These are the most important new features in XPath 3.1:

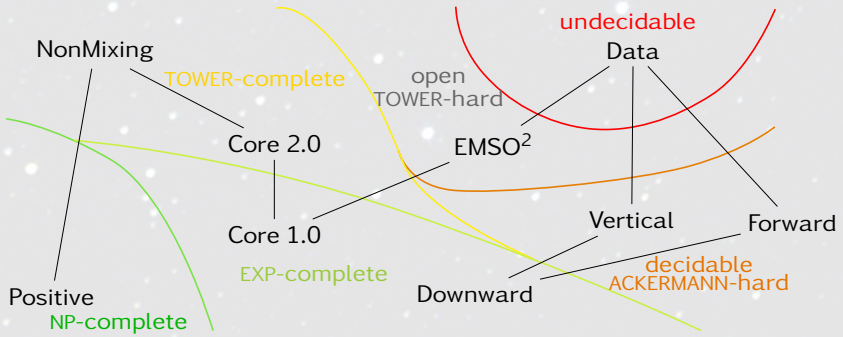
Decidable Fragments

Problem (Satisfiability)

input φ an XPath query

question $\exists$data tree t . $t \models \varphi$?

Expressiveness/Complexity



Overview

benchmark

- open source
- 21,141 XPath queries

coverage of decidable XPath fragments

- “vanilla”
- simple extensions

analysis

Overview

benchmark

- open source
- 21,141 XPath queries

coverage of decidable XPath fragments

- “vanilla”
- simple extensions

analysis

Overview

benchmark

- open source
- 21,141 XPath queries

coverage of decidable XPath fragments

- "vanilla"
- simple extensions

analysis

Overview

benchmark

- open source
- 21,141 XPath queries

coverage of decidable XPath fragments

- “vanilla”
- simple extensions

analysis

Overview

benchmark

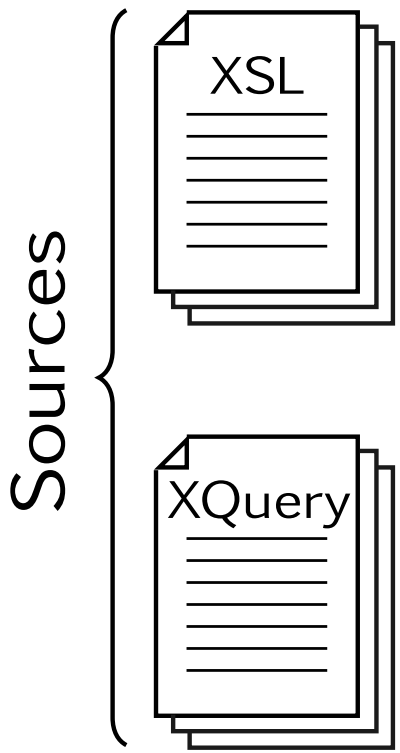
- open source
- 21,141 XPath queries

coverage of decidable XPath fragments

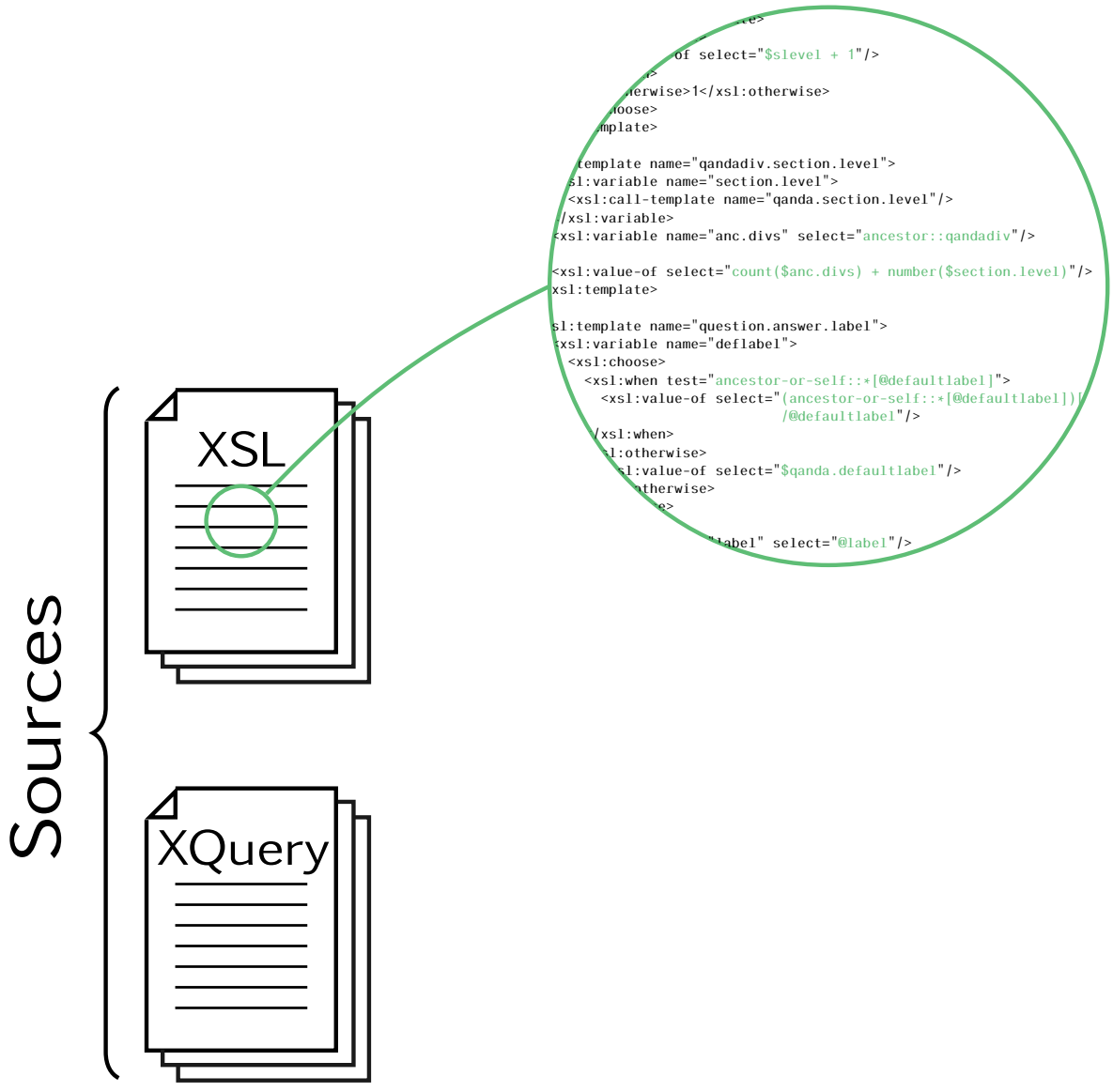
- “vanilla”
- simple extensions

analysis

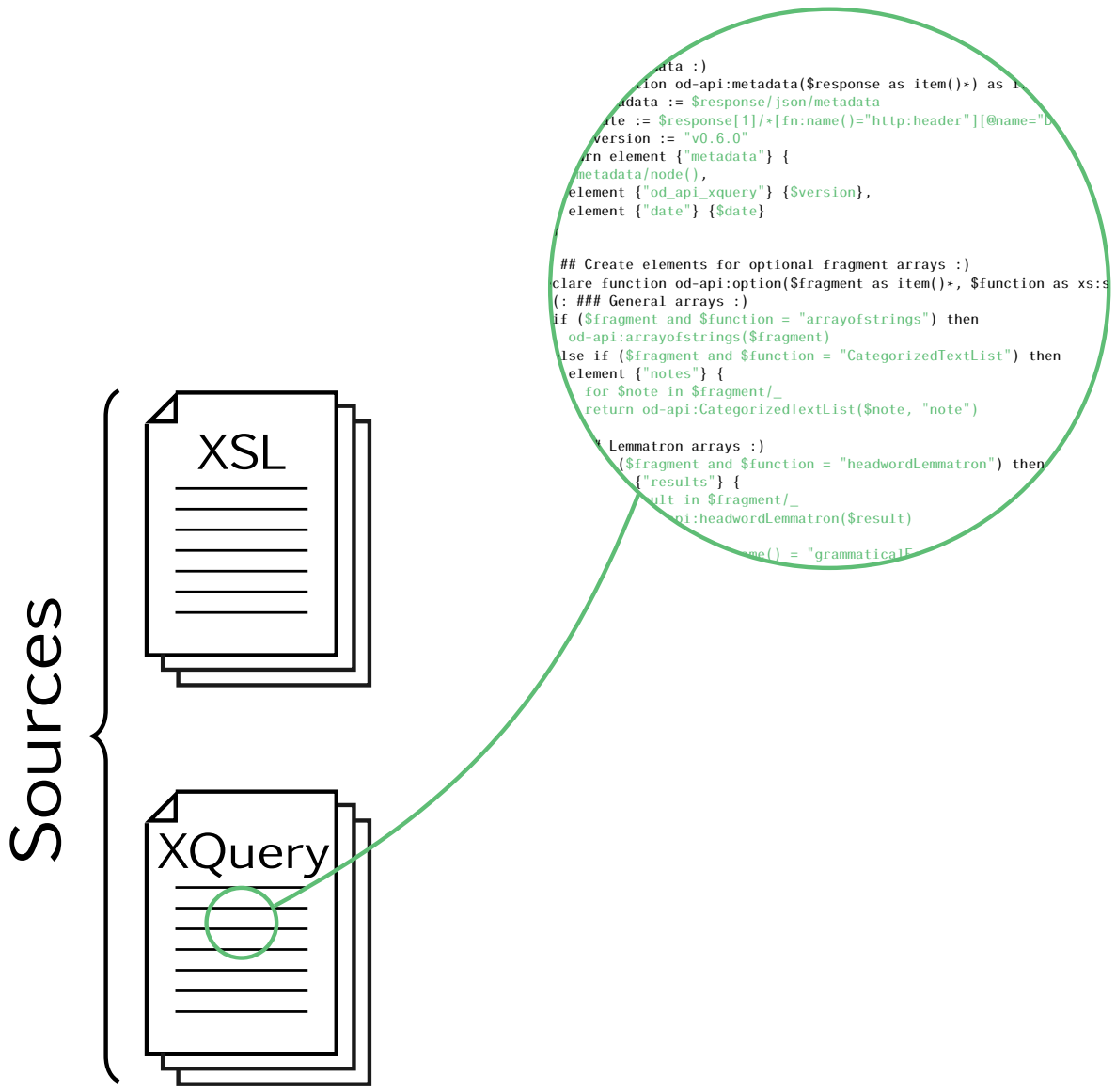
Benchmark Construction



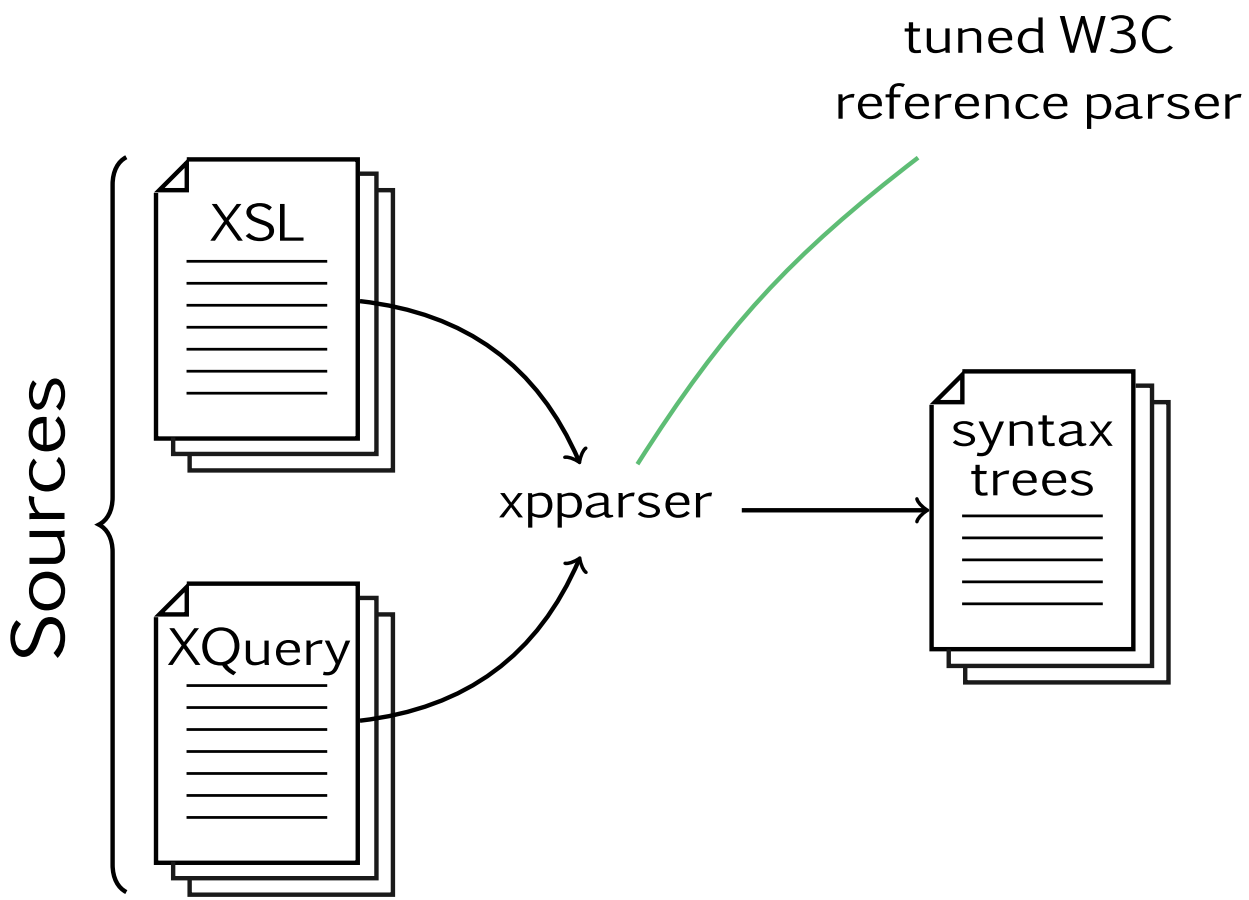
Benchmark Construction



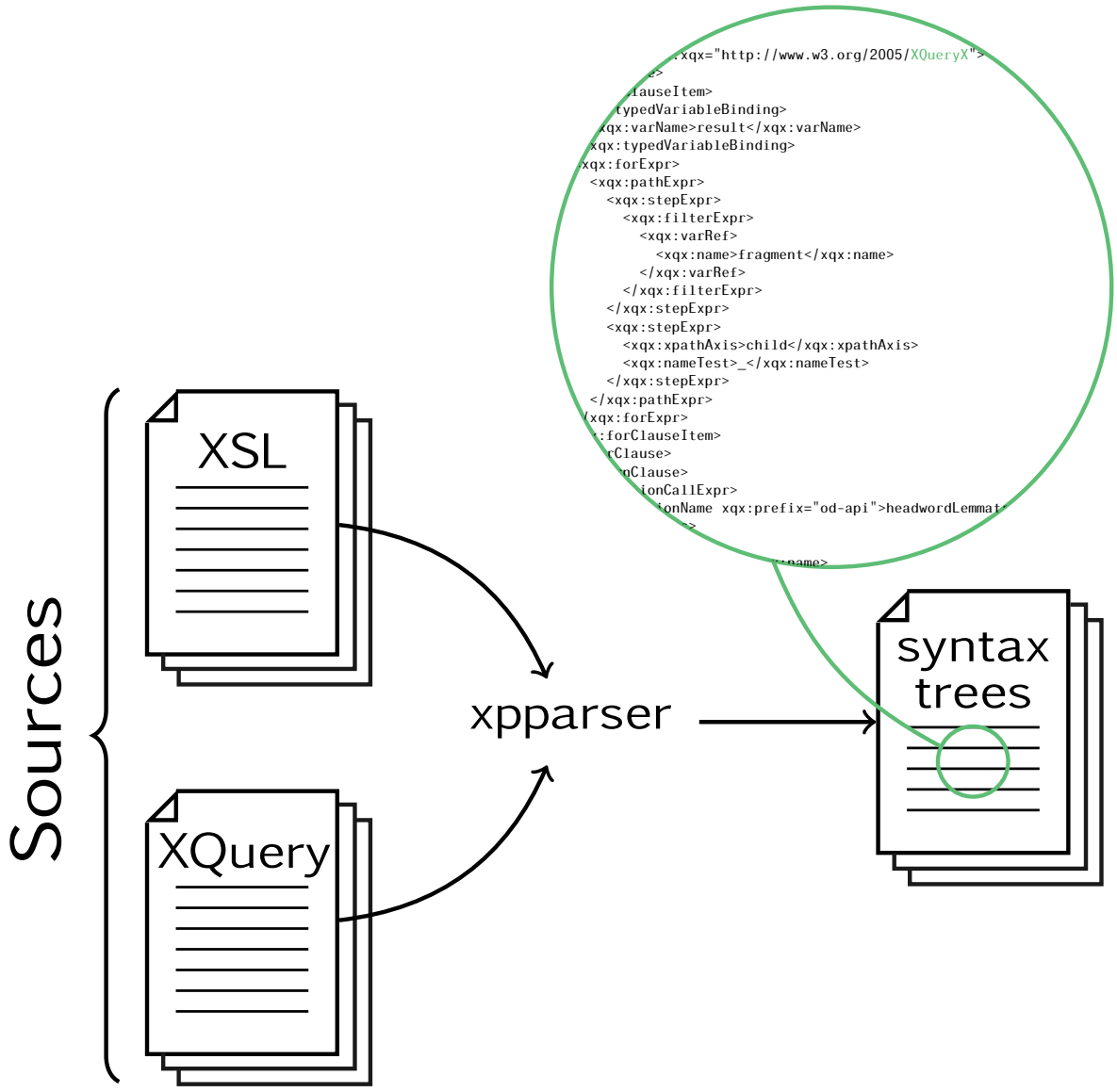
Benchmark Construction



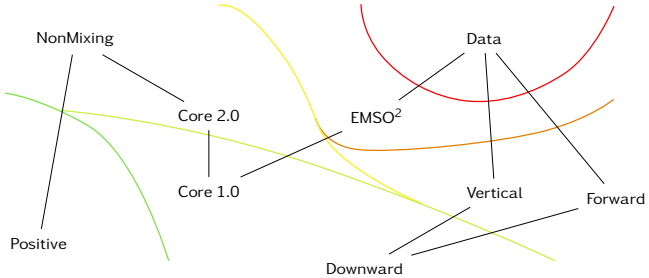
Benchmark Construction

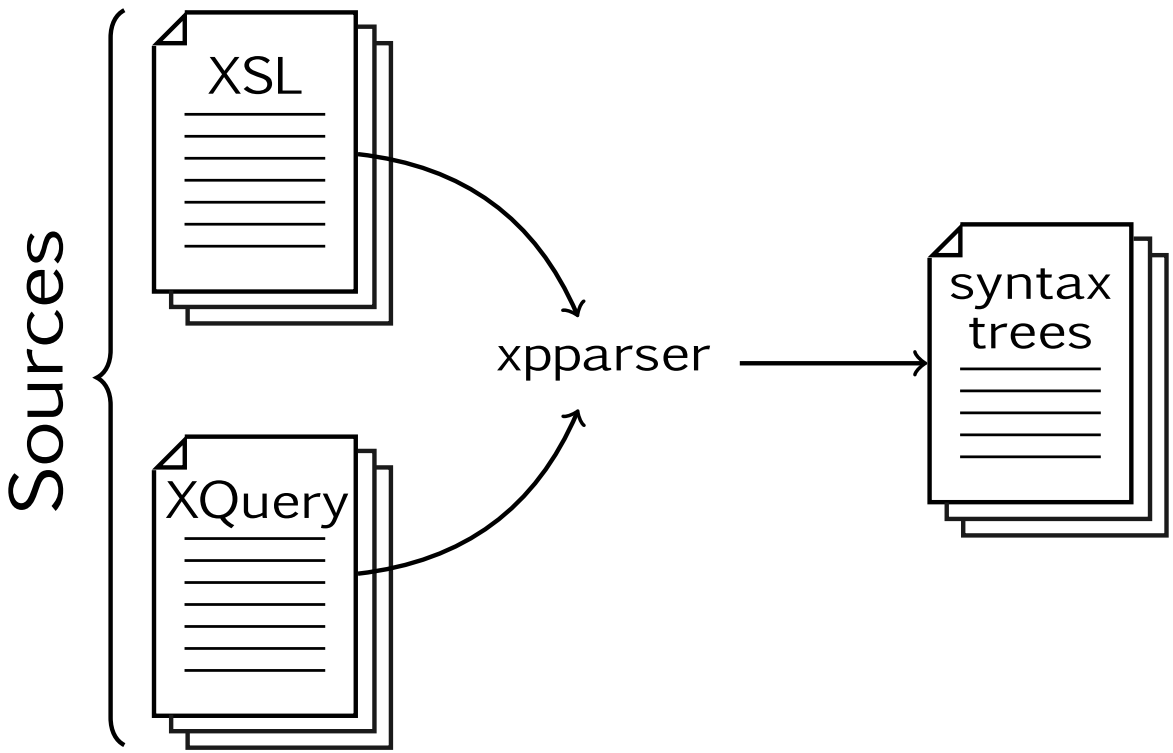


Benchmark Construction

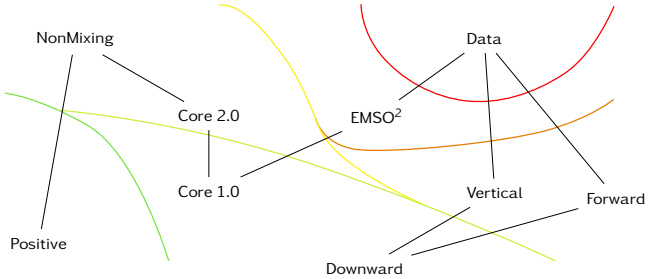


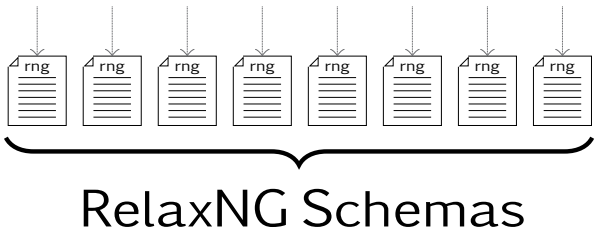
Benchmark Construction



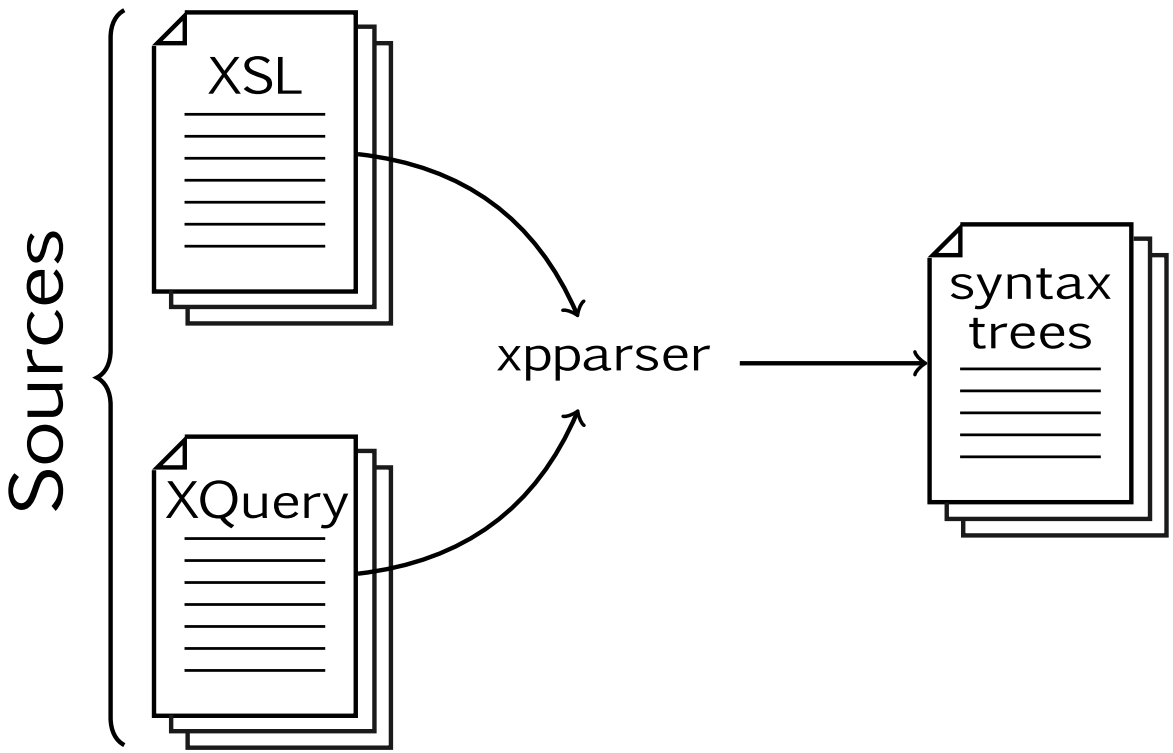


Benchmark Construction

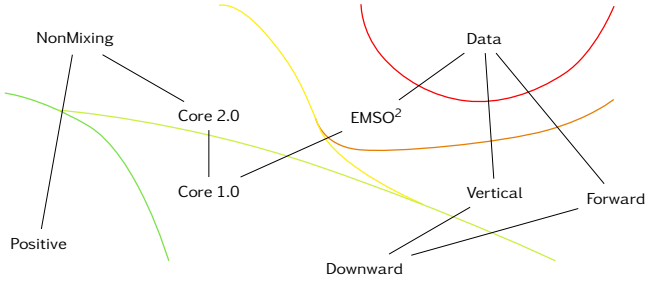


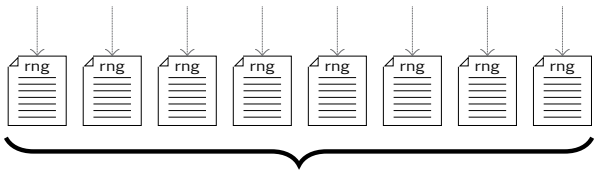


RelaxNG Schemas

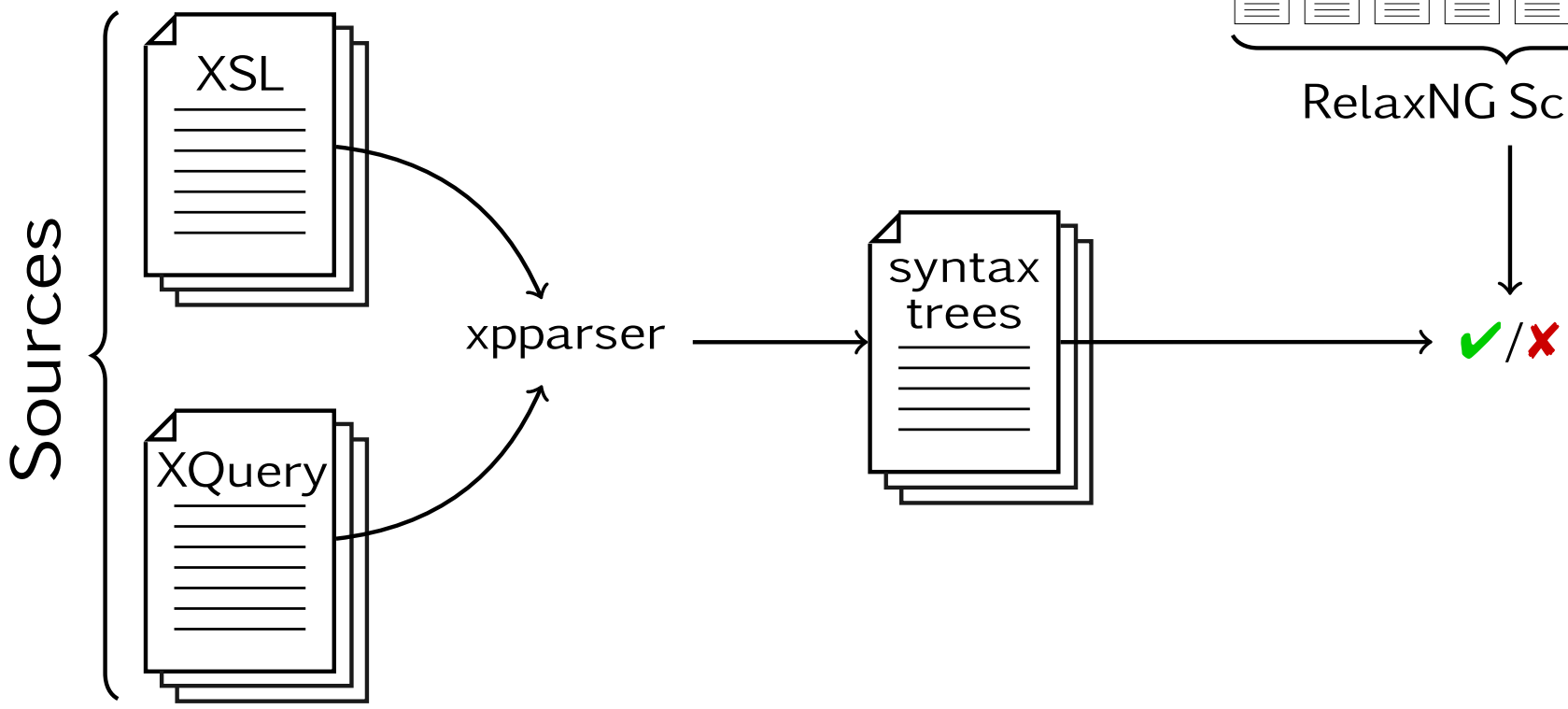


Benchmark Construction





RelaxNG Schemas



Benchmark Composition

Sources	Queries	Coverage		
		XPath 1.0	XPath 2.0	XPath 3.0
XSLT	14,675	98.4%	100.0%	100.0%
XQuery	6,466	76.1%	87.4%	99.8%
Total	21,141	91.6%	96.1%	99.9%

Benchmark: Functions

XPath and XQuery Functions and Operators 3.1

W3C Recommendation 21 March 2017

This version:
https://www.w3.org/TR/2017/REC-xpath-functions-31-20170321/

Latest version of XPath and XQuery Functions and Operators 3.1:
https://www.w3.org/TR/xpath-functions-31/

Previous versions of XPath and XQuery Functions and Operators 3.1:
https://www.w3.org/TR/2017/PR-xpath-functions-31-20170117/
https://www.w3.org/TR/2016/CR-xpath-functions-31-20161213/
https://www.w3.org/TR/2014/CR-xpath-functions-31-20141218/
https://www.w3.org/TR/2014/WD-xpath-functions-31-20141007/
https://www.w3.org/TR/2014/WD-xpath-functions-31-20140424/

Most recent version of XPath and XQuery Functions and Operators 3:
https://www.w3.org/TR/xpath-functions-3/

Most recent version of XPath and XQuery Functions and Operators:
https://www.w3.org/TR/xpath-functions/

Most recent Recommendation of XPath and XQuery Functions and Operators:
https://www.w3.org/TR/2014/REC-xpath-functions-30-20140408/

Editor:
Michael Kay (XSLT WG), Saxonica <http://www.saxonica.com/>

Please check the **errata** for any errors or issues reported since publication.

See also **translations**.

This document is also available in these non-normative formats: Specification in XML format using HTML5 vocabulary, XML function catalog, and HTML with change markings relative to version 3.0.

Abstract

This document defines constructor functions, operators, and functions on the datatypes defined in [XML Schema Part 2: Datatypes Second Edition] and the datatypes defined in [XQuery and XPath Data Model (XDM) 3.1]. It also defines functions and operators on nodes and node sequences as defined in the [XQuery and XPath Data Model (XDM) 3.1]. These functions and operators are defined for use in [XML Path Language (XPath) 3.1] and [XQuery 3.1: An XML Query Language] and [XSL Transformations (XSLT) Version 3.0] and other related XML standards. The signatures and summaries of functions defined in this document are available at: http://www.w3.org/2005/xpath-functions/.

Benchmark: Functions

XPath and XQuery Functions and Operators 3.1

W3C Recommendation 21 March 2017

This version:
https://www.w3.org/TR/2017/REC-xpath-functions-31-20170321/

Latest version of XPath and XQuery Functions and Operators 3.1:
https://www.w3.org/TR/xpath-functions-31/

Previous versions of XPath and XQuery Functions and Operators 3.1:
https://www.w3.org/TR/2017/PR-xpath-functions-31-20170117/
https://www.w3.org/TR/2016/CR-xpath-functions-31-20161213/
https://www.w3.org/TR/2014/CR-xpath-functions-31-20141218/
https://www.w3.org/TR/2014/WD-xpath-functions-31-20141007/
https://www.w3.org/TR/2014/WD-xpath-functions-31-20140424/

Most recent version of XPath and XQuery Functions and Operators 3:
https://www.w3.org/TR/xpath-functions-3/

Most recent version of XPath and XQuery Functions and Operators:
https://www.w3.org/TR/xpath-functions/

Most recent Recommendation of XPath and XQuery Functions and Operators:
https://www.w3.org/TR/2014/REC-xpath-functions-30-20140408/

Editor:
Michael Kay (XSLT WG), Saxonica <http://www.saxonica.com/>

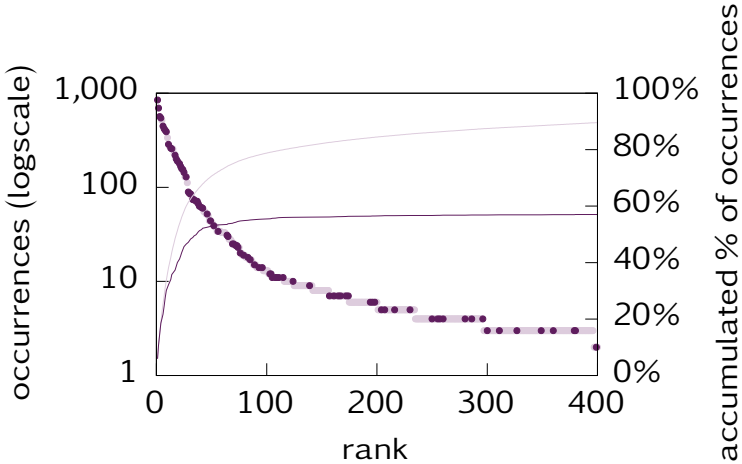
Please check the **errata** for any errors or issues reported since publication.

See also **translations**.

This document is also available in these non-normative formats: Specification in XML format using HTML5 vocabulary, XML function catalog, and HTML with change markings relative to version 3.0.

Abstract

This document defines constructor functions, operators, and functions on the datatypes defined in [XML Schema Part 2: Datatypes Second Edition] and the datatypes defined in [XQuery and XPath Data Model (XDM) 3.1]. It also defines functions and operators on nodes and node sequences as defined in the [XQuery and XPath Data Model (XDM) 3.1]. These functions and operators are defined for use in [XML Path Language (XPath) 3.1] and [XQuery 3.1: An XML Query Language] and [XSL Transformations (XSLT) Version 3.0] and other related XML standards. The signatures and summaries of functions defined in this document are available at: http://www.w3.org/2005/xpath-functions/.



- standard functions (dark violet)
  57.23% of function calls
  76.32% in XSLT
  42.93% in XQuery
- non-standard ones (light violet)
  typically user-defined

Benchmark: Functions

XPath and XQuery Functions and Operators 3.1

W3C Recommendation 21 March 2017

This version:
https://www.w3.org/TR/2017/REC-xpath-functions-31-20170321/

Latest version of XPath and XQuery Functions and Operators 3.1:
https://www.w3.org/TR/xpath-functions-31/

Previous versions of XPath and XQuery Functions and Operators 3.1:
https://www.w3.org/TR/2017/PR-xpath-functions-31-20170117/
https://www.w3.org/TR/2016/CR-xpath-functions-31-20161213/
https://www.w3.org/TR/2014/CR-xpath-functions-31-20141218/
https://www.w3.org/TR/2014/WD-xpath-functions-31-20141007/
https://www.w3.org/TR/2014/WD-xpath-functions-31-20140424/

Most recent version of XPath and XQuery Functions and Operators 3:
https://www.w3.org/TR/xpath-functions-3/

Most recent version of XPath and XQuery Functions and Operators:
https://www.w3.org/TR/xpath-functions/

Most recent Recommendation of XPath and XQuery Functions and Operators:
https://www.w3.org/TR/2014/REC-xpath-functions-30-20140408/

Editor:
Michael Kay (XSLT WG), Saxonica <http://www.saxonica.com/>

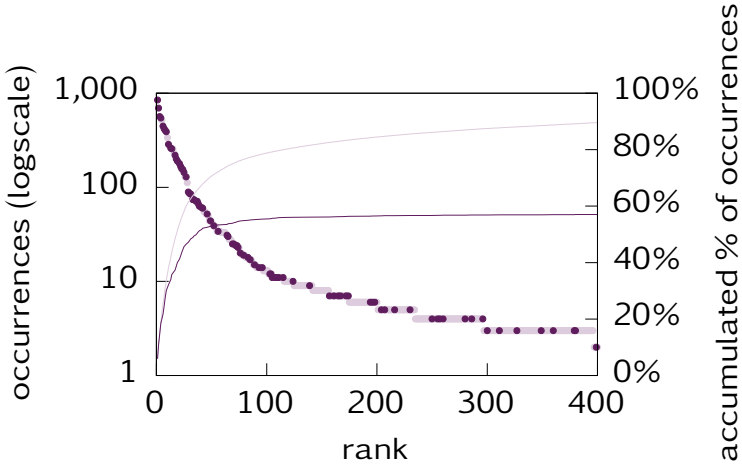
Please check the **errata** for any errors or issues reported since publication.

See also **translations**.

This document is also available in these non-normative formats: Specification in XML format using HTML5 vocabulary, XML function catalog, and HTML with change markings relative to version 3.0.

Abstract

This document defines constructor functions, operators, and functions on the datatypes defined in [XML Schema Part 2: Datatypes Second Edition] and the datatypes defined in [XQuery and XPath Data Model (XDM) 3.1]. It also defines functions and operators on nodes and node sequences as defined in the [XQuery and XPath Data Model (XDM) 3.1]. These functions and operators are defined for use in [XML Path Language (XPath) 3.1] and [XQuery 3.1: An XML Query Language] and [XSL Transformations (XSLT) Version 3.0] and other related XML standards. The signatures and summaries of functions defined in this document are available at: http://www.w3.org/2005/xpath-functions/.

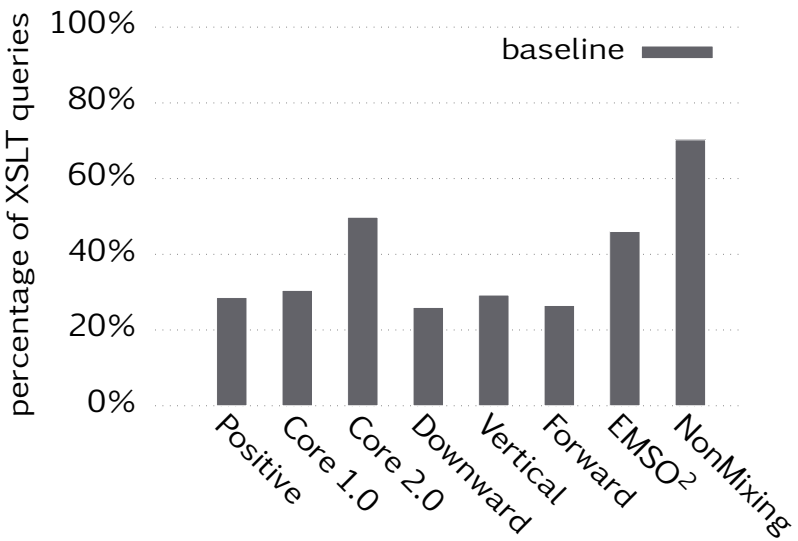


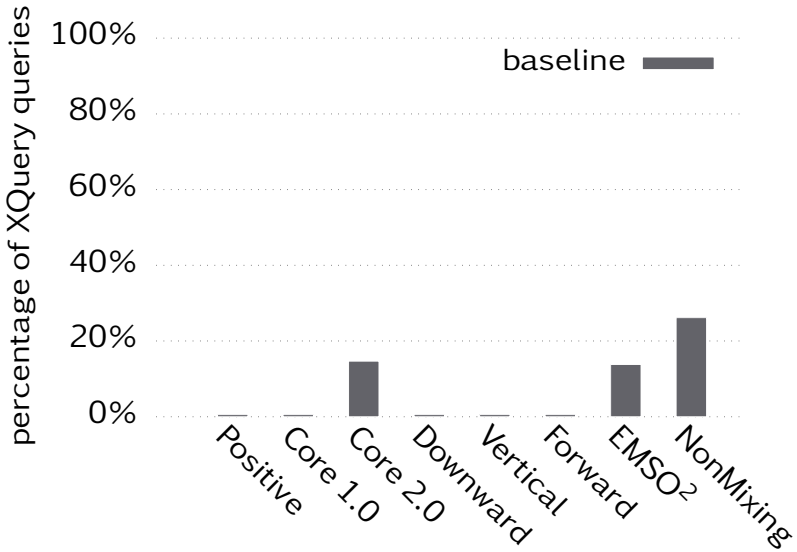
- standard functions (dark violet)
  57.23% of function calls
  76.32% in XSLT
  42.93% in XQuery
- non-standard ones (light violet)
  typically user-defined

Benchmark: Functions

Sources	Queries	Coverage			
		XPath 1.0	XPath 2.0	XPath 3.0	XPath 3.0 std
XSLT	14,675	98.4%	100.0%	100.0%	91.3%
XQuery	6,466	76.1%	87.4%	99.8%	46.7%
Total	21,141	91.6%	96.1%	99.9%	77.7%

Basic Coverage





- poor coverage
- except for NonMixing fragment
  MSO + positive data tests & data joins not mixing $=$ and $\neq$

Extending fragments

A syntactic construct can be

expressed through equivalent requests

encoded through equisatisfiable requests

Extending fragments

A syntactic construct can be

expressed through equivalent requests

encoded through equisatisfiable requests

Extending fragments

A syntactic construct can be (polynomially)

expressed through (polynomial time computable) equivalent requests

encoded through (polynomial time computable) equisatisfiable requests

defines a front-end

Extensions

$/\pi$ root navigation, e.g.

```
//firstterm
```

$\$x$ free variables, e.g.

```
$module/merge
```

$\pi \triangle d$ data tests against constants, e.g.

```
refmeta/refmiscinfo[@class = 'version']
```

$\pi \triangle \pi$ positive data joins, e.g.

```
a/@href = preceding-sibling::li/a/@href
```

`last()` one-step positional predicates, e.g.

```
tocentry[position() = last()]
```

Extensions

$/\pi$ root navigation, e.g.

```
//firstterm
```

$\$x$ free variables, e.g.

```
$module/merge
```

$\pi \triangle d$ data tests against constants, e.g.

```
refmeta/refmiscinfo[@class = 'version']
```

$\pi \triangle \pi$ positive data joins, e.g.

```
a/@href = preceding-sibling::li/a/@href
```

`last()` one-step positional predicates, e.g.

```
tocentry[position() = last()]
```

Extensions

$/\pi$ root navigation, e.g.

```
//firstterm
```

$\$x$ free variables, e.g.

```
$module/merge
```

$\pi \triangle \mathrm{d}$ data tests against constants, e.g.

```
refmeta/refmiscinfo[@class = 'version']
```

$\pi \triangle \pi$ positive data joins, e.g.

```
a/@href = preceding-sibling::li/a/@href
```

`last()` one-step positional predicates, e.g.

```
tocentry[position() = last()]
```

Extensions

$/\pi$ root navigation, e.g.

```
//firstterm
```

$\$x$ free variables, e.g.

```
$module/merge
```

$\pi \triangle d$ data tests against constants, e.g.

```
refmeta/refmiscinfo[@class = 'version']
```

$\pi \triangle \pi$ positive data joins, e.g.

```
a/@href = preceding-sibling::li/a/@href
```

`last()` one-step positional predicates, e.g.

```
tocentry[position() = last()]
```

Extensions

$/\pi$ root navigation, e.g.

```
//firstterm
```

$\$x$ free variables, e.g.

```
$module/merge
```

$\pi \triangle d$ data tests against constants, e.g.

```
refmeta/refmiscinfo[@class = 'version']
```

$\pi \triangle \pi$ positive data joins, e.g.

```
a/@href = preceding-sibling::li/a/@href
```

`last()` one-step positional predicates, e.g.

```
tocentry[position() = last()]
```

Expressible Extensions

Example (expressible root navigation)

```
//firstterm
```



```
ancestor-or-self::*[not(parent::*)]//firstterm
```

Expressible Extensions

Example (expressible root navigation)

```
//firstterm
```

```
ancestor-or-self::*[not(parent::*)]//firstterm
```

Expressible Extensions

Example (expressible root navigation)

```
//firstterm
    ⇝
ancestor-or-self::*[not(parent::*)]//firstterm
```

Fact (¬expressible)

Root navigation cannot be expressed in DownwardXPath.

Encodable Extensions

Example (encodable free variables)

`$module/merge`

⇝ extend label set Σ to $\Sigma \times 2^{\{\text{free vars}\}}$

`//.[or`$_{a \in \Sigma, \$module \in S}(a, S)$`]/self::*[or`$_S$`(merge,`$S$`)]`

Encodable Extensions

Example (encodable free variables)

`$module/merge`

$\downarrow$ extend label set Σ to $\Sigma \times 2^{\{\text{free vars}\}}$

`//.[or`$_{a \in \Sigma, \$module \in S}$`(a,S)]/self::*[or`$_S$`(merge,S)]`

Encodable Extensions

Example (encodable free variables)

$\texttt{\$module/merge}$

$\downarrow\!\!\!\!\downarrow$ extend label set Σ to $\Sigma \times 2^{\{\text{free vars}\}}$

$\texttt{//.[or}_{a \in \Sigma, \texttt{\$module} \in S}(a, S)\texttt{]/self::*[or}_S(\texttt{merge}, S)\texttt{]}$

Proposition (¬encodable, c.f. Figueira & Segoufin 2009)

Satisfiability in ForwardXPath extended with root navigation or free variables is undecidable.

Encodable Extensions

Example (encodable free variables)

`$module/merge`

$\downarrow$ extend label set Σ to $\Sigma \times 2^{\{\text{free vars}\}}$

`//.[or`$_{a \in \Sigma, \$module \in S}$`(a,S)]/self::*[or`$_S$`(merge,S)]`

Proposition (¬encodable, c.f. Figueira & Segoufin 2009)
Satisfiability in ForwardXPath extended with root navigation or free variables is undecidable.

Proposition (¬poly. enc., c.f. Figueira & Segoufin 2009)
Satisfiability in DownwardXPath extended with `[position()=last()]` and `[position()!=last()]` is ACKERMANN-hard.

Extensions

	Positive	*Core 1.0*	*Core 2.0*	*Downward*	*Vertical*	*Forward*	*EMSO²*	*NonMixing*
$/\pi$								
$\$x$							nat.	nat.
$\pi \triangle d$	nat.							nat.†
$\pi \triangle \pi$	nat.			nat.*	nat.*	nat.*		nat.†
`last()`								

* support limited by available axes † support limited to non-mixing

Extensions

	Positive	*Core 1.0*	*Core 2.0*	*Downward*	*Vertical*	*Forward*	*EMSO²*	*NonMixing*
$/\pi$	expr.	expr.	expr.		expr.		expr.	expr.
$x							nat.	nat.
$\pi \triangle d$	nat.							nat.†
$\pi \triangle \pi$	nat.			nat.*	nat.*	nat.*		nat.†
last()			expr.					expr.

* support limited by available axes † support limited to non-mixing

Extensions

	Positive	*Core 1.0*	*Core 2.0*	*Downward*	*Vertical*	*Forward*	*EMSO²*	*NonMixing*
$/\pi$	expr.	expr.	expr.		expr.		expr.	expr.
$x	enc.	enc.	enc.		enc.		nat.	nat.
$\pi \triangle d$	nat.	enc.	enc.	enc.	enc.	enc.	enc.	nat.†
$\pi \triangle \pi$	nat.	enc.	enc.	nat.*	nat.*	nat.*	enc.	nat.†
last()		enc.	expr.		enc.*	enc.*		expr.

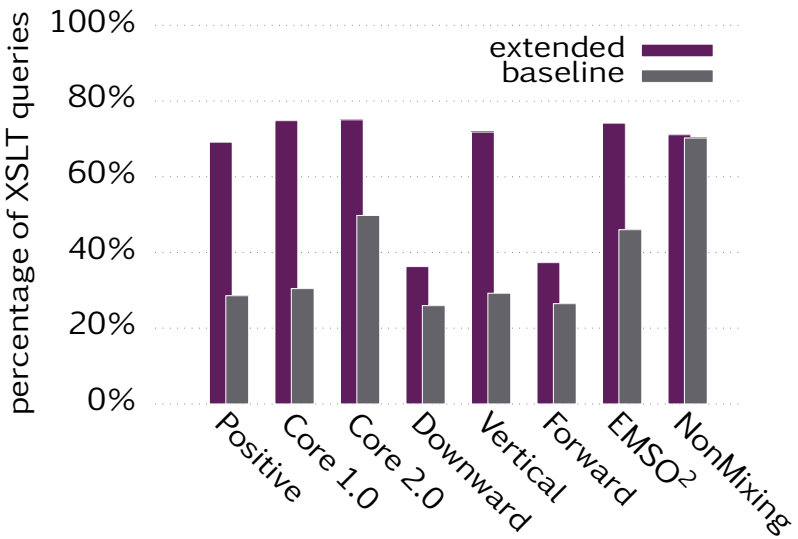
* support limited by available axes † support limited to non-mixing

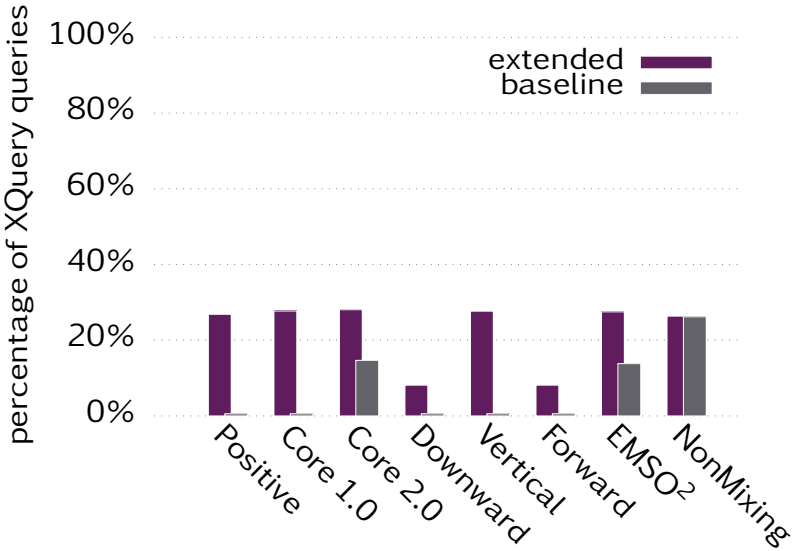
Extensions

	Positive	*Core 1.0*	*Core 2.0*	*Downward*	*Vertical*	*Forward*	*EMSO²*	*NonMixing*
$/\pi$	expr.	expr.	expr.	¬expr.	expr.	¬enc.	expr.	expr.
$x	enc.	enc.	enc.	¬expr.	enc.	¬enc.	nat.	nat.
$\pi \triangle d$	nat.	enc.	enc.	enc.	enc.	enc.	enc.	nat.†
$\pi \triangle \pi$	nat.	enc.	enc.	nat.*	nat.*	nat.*	enc.	nat.†
last()		enc.	expr.	¬p.enc.	enc.*	enc.*		expr.

* support limited by available axes † support limited to non-mixing

Extended Coverage

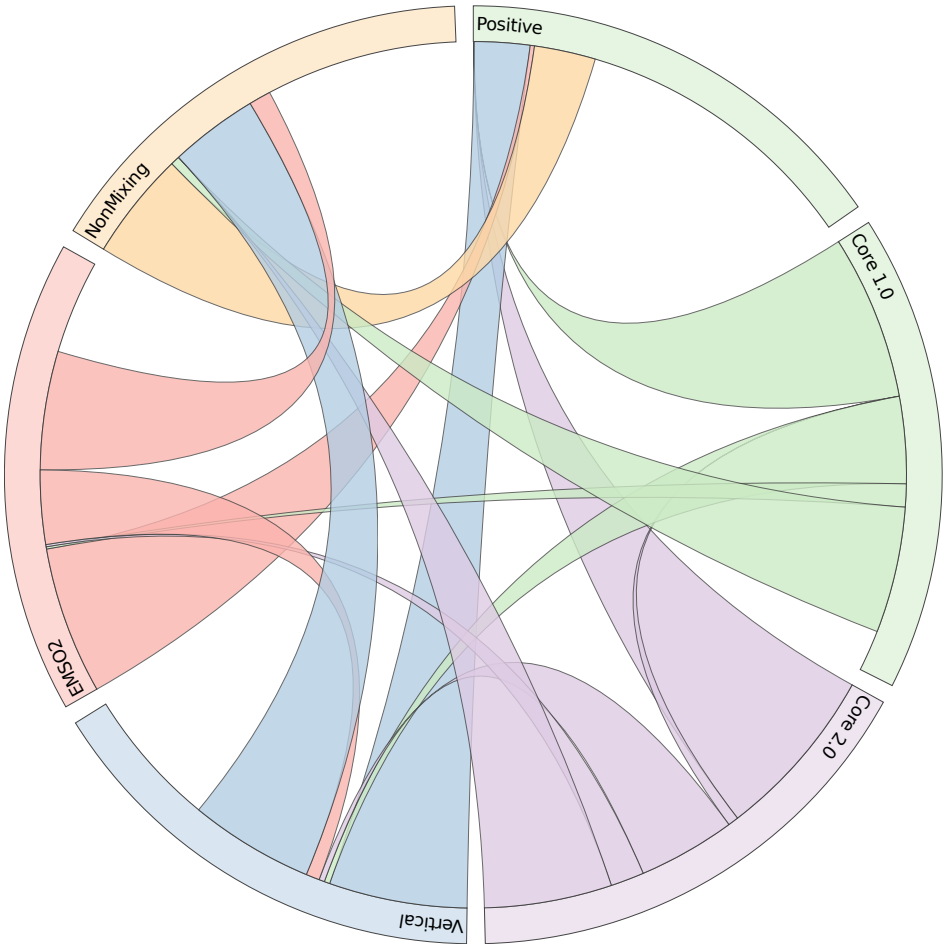




combined 60.86%

Core 2.0 75.03% of XSLT and 28.08% of XQuery

Comparison



Interactive interface

`http://www.lsv.fr/~schmitz/xpparser`

Function Support

Coverage with respect to XPath 3.0 std.

combined 78.33%

Core 2.0 82.14% of XSLT and 60.00% of XQuery

Extra support through SMT?

- interval encoding of trees for Positive fragment
- support for linear arithmetic and string functions like `concat()`, `contains()`, `string-length()`, etc.
- 62.75% coverage, 84.77% of XSLT wrt. XPath 3.0 std

Function Support

Coverage with respect to XPath 3.0 std.

combined 78.33%

Core 2.0 82.14% of XSLT and 60.00% of XQuery

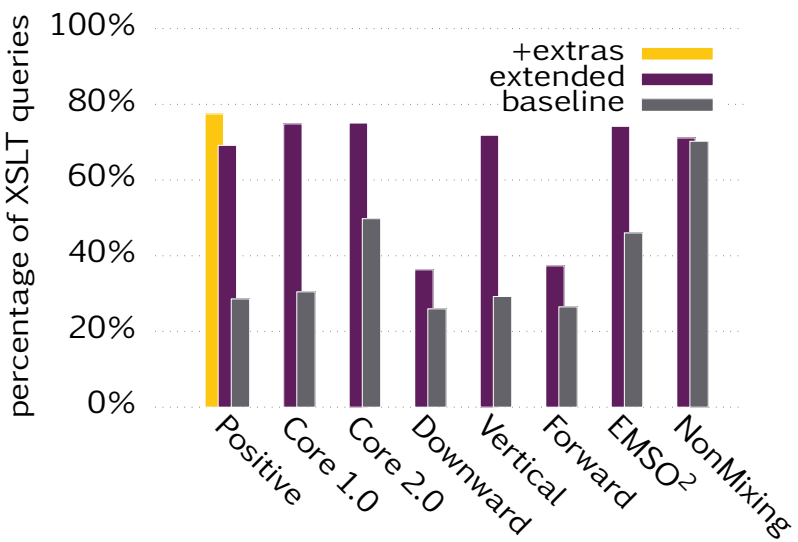
Extra support through SMT?

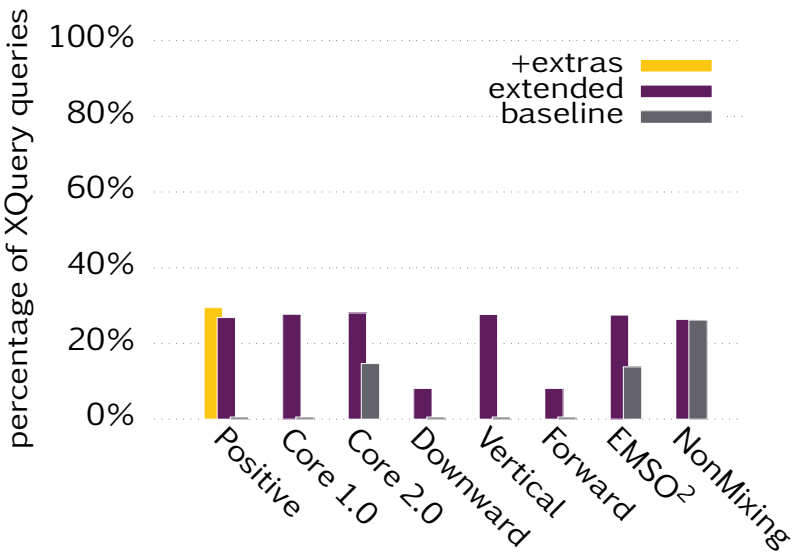
- interval encoding of trees for Positive fragment
- support for linear arithmetic and string functions like `concat()`, `contains()`, `string-length()`, etc.
- 62.75% coverage, 84.77% of XSLT wrt. XPath 3.0 std

Function Support

Extra support through SMT?

- interval encoding of trees for Positive fragment
- support for linear arithmetic and string functions like `concat()`, `contains()`, `string-length()`, etc.
- 62.75% coverage, 84.77% of XSLT wrt. XPath 3.0 std





Conclusions

benchmark of 21,141 XPath queries

xpparser: swh:1:dir:081194c82832b17c4d1229313e85753dfb3ec4cd
benchmark: swh:1:dir:1ea68cf5bb3f9f3f2fe8c7995f1802ebadf17fb5

satisfiability

- importance of front-end
- XSLT support vs. XQuery support
- challenge: function support
- future? SMT techniques

Conclusions

benchmark of 21,141 XPath queries

xpparser: swh:1:dir:081194c82832b17c4d1229313e85753dfb3ec4cd
benchmark: swh:1:dir:1ea68cf5bb3f9f3f2fe8c7995f1802ebadf17fb5

satisfiability

- importance of front-end
- XSLT support vs. XQuery support
- challenge: function support
- future? SMT techniques

Conclusions

benchmark of 21,141 XPath queries

xpparser: swh:1:dir:081194c82832b17c4d1229313e85753dfb3ec4cd
benchmark: swh:1:dir:1ea68cf5bb3f9f3f2fe8c7995f1802ebadf17fb5

satisfiability

- importance of front-end
- XSLT support vs. XQuery support
- challenge: function support
- future? SMT techniques

Conclusions

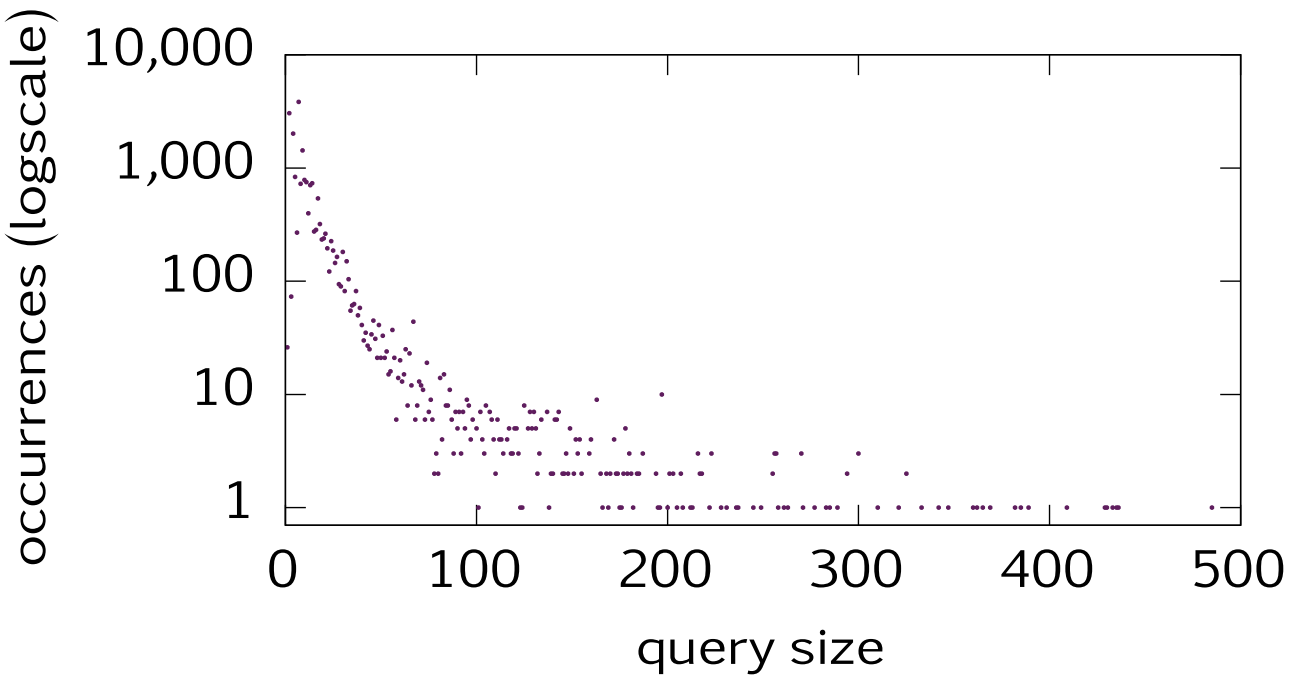
benchmark of 21,141 XPath queries

xpparser: swh:1:dir:081194c82832b17c4d1229313e85753dfb3ec4cd
benchmark: swh:1:dir:1ea68cf5bb3f9f3f2fe8c7995f1802ebadf17fb5

satisfiability

- importance of front-end
- XSLT support vs. XQuery support
- challenge: function support
- future? SMT techniques

Benchmark: Size Distribution



Synthetic Benchmarks

Sources	Queries	Coverage XPath 1.0	XPath 2.0	XPath 3.0
XPathMark-FT	64	100.0%	100.0%	100.0%
XPathMark-PT	38	100.0%	100.0%	100.0%
XMark	66	92.4%	93.9%	100.0%
Total	168	97.0%	97.6%	100.0%