



XPointer xmlns() Scheme

W3C Recommendation 25 March 2003

This version:

<http://www.w3.org/TR/2003/REC-xptr-xmlns-20030325/>

Latest version:

<http://www.w3.org/TR/xptr-xmlns/>

Previous version:

<http://www.w3.org/TR/2002/PR-xptr-xmlns-20021113/>

Authors and Contributors:

Steven J. DeRose <sderose@acm.org>

Ron Daniel Jr. <rdaniel@taxonomystrategies.com>

Eve Maler (Sun Microsystems) <eve.maler@sun.com>

Jonathan Marsh (Microsoft) <jmarsh@microsoft.com>

Copyright © 2003 W3C[®] (MIT, INRIA, Keio), All Rights Reserved.
W3C [liability](#), [trademark](#), [document use](#), and [software licensing](#) rules apply.

Abstract

The XPointer `xmlns()` scheme is intended to be used with the XPointer Framework [[XPtrFrame](#)] to allow correct interpretation of namespace prefixes in pointers, for instance, namespace-qualified scheme names and namespace-qualified element or attribute names appearing within scheme data.

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. The latest status of this document series is maintained at the W3C.

This document is a [Recommendation \(REC\)](#) of the W3C. It has been reviewed by W3C Members and other interested parties and has been endorsed by the Director as a W3C Recommendation. It is a stable document and may be used as reference material or cited as a normative reference from another document.

W3C's role in making the Recommendation is to draw attention to the specification and to promote its widespread deployment. This enhances the functionality and interoperability of the Web.

This document has been produced by the [W3C XML Linking Working Group](#) as part of the [XML Activity](#). It is intended to address a subset of the original [XPointer requirements](#), and to serve, along with the accompanying [XPointer Framework](#) and [XPointer element\(\) Scheme](#) specifications, as a part of a fragment identifier syntax for the XML Media types.

Public comments on this Recommendation are welcome. Please send them to the public mailing list www-xml-linking-comments@w3.org ([archive](#)).

Information about implementations relevant to this specification and the accompanying [XPointer element\(\) Scheme](#) and [XPointer xmlns\(\) Scheme](#) can be found in the [Implementation Report](#).

There are patent disclosures and license commitments associated with this Recommendation, which may be found on the [XPointer IPR Statement](#) page in conformance with [W3C policy](#).

A list of current W3C Recommendations and other technical documents can be found at <http://www.w3.org/TR/>. W3C publications may be updated, replaced, or obsoleted by other documents at any time.

Table of Contents

1. Introduction	1
2. Conformance	1
3. Language and Processing	1

Appendices

A. Normative References	2
B. Non-Normative References (Non-Normative)	2

This page is intentionally left blank.

1. Introduction

The XPointer `xmlns()` scheme is intended to be used with the XPointer Framework [XPtrFrame] to allow correct interpretation of namespace prefixes in pointers, for instance, namespace-qualified scheme names and namespace-qualified element or attribute names appearing within scheme data.

The key words *must*, *must not*, *required*, *shall*, *shall not*, *should*, *should not*, *recommended*, *may*, and *optional* in this specification are to be interpreted as described in [RFC 2119].

The terms pointer part, scheme, XPointer processor, and namespace binding context are used in this specification as [defined](#) in the XPointer Framework specification.

The formal grammar for the `xmlns()` scheme is given using simple Extended Backus-Naur Form (EBNF) notation, as described in the XML Recommendation [XML].

2. Conformance

This specification normatively depends on the XPointer Framework [XPtrFrame] specification.

Conforming XPointer processors claiming to support the `xmlns()` scheme must conform to the behavior defined in this specification and may conform to additional XPointer scheme specifications.

3. Language and Processing

This section describes the syntax and semantics of the `xmlns()` scheme and the behavior of XPointer processors with respect to this scheme.

The scheme name is “xmlns”. The scheme data syntax is as follows; if scheme data in a pointer part with the `xmlns()` scheme does not conform to the syntax defined in this section the pointer part does not contribute an entry to the namespace binding context.

xmlns() Scheme Syntax

[1] `XmlnsSchemeData ::= NCName S? '=' S? EscapedNamespaceName`

[2] `EscapedNamespaceName ::= EscapedData*`

`S` is as defined in the XML specification [XML], `NCName` is as defined in the XML Namespaces specification [XML-Names], and `EscapedData` is as defined in the XPointer Framework specification [XPtrFrame].

A pointer part with the `xmlns()` scheme declares a namespace prefix (`NCName`) to be associated with an XML namespace name (`EscapedNamespaceName` with any circumflex-escaping undone). Each pointer part that uses the `xmlns()` scheme contributes a new entry to the namespace binding context. If a pointer part defines a binding for a namespace prefix that already has an entry in the namespace binding context, the new entry overrides the old one.

For example, the following pointer part binds the `abc` prefix to the `http://example.com/ns/abc` namespace name:

```
xmlns(abc=http://example.com/ns/abc)
```

A pointer part that uses the `xmlns()` scheme never identifies a subresource and thus the XPointer processor evaluation always continues with the next pointer part. However, evaluation of the `xmlns()` pointer part has a potential effect on pointer parts to its right.

The XPointer Framework on which this specification is based dictates that the initial namespace binding context prior to evaluation of the first pointer part consists of a single entry: the `xml` prefix bound to the URI `http://www.w3.org/XML/1998/namespace`. Following the rules specified by the Framework, any attempt by a pointer part using the `xmlns()` scheme to redefine the `xml` prefix will result in no change to the namespace binding context. Any attempt to bind a prefix to the namespace name `http://www.w3.org/XML/1998/namespace` will result in no change to the namespace binding context. Any attempt to define the `xmlns` prefix will result in no change to the namespace binding context. Any attempt to bind a prefix to the namespace name `http://www.w3.org/2000/xmlns/` will result in no change to the namespace binding context.

The prefix used in pointer parts need not match the prefix (or lack of a prefix) actually used in the XML resource into which the pointer part is addressing. For example, assume the following target XML resource:

```
<customer xmlns="http://example.org/customer">
  <name xmlns="http://example.org/personal-info">John Doe</name>
</customer>
```

Evaluation of a pointer part that refers to element `customer` will not identify any subresources if the namespace binding context contains no entry that binds a prefix to the namespace name `http://example.org/customer`. The following two pointer parts allow the `xpointer()` [[XPointer](#)] part to the right to refer to the `customer` element (as `c:customer`) and the `name` element (as `p:name`):

```
xmlns(c=http://example.org/customer) xmlns(p=http://example.org/personal-info)
xpointer(/c:customer/p:name)
```

Appendix A. Normative References

RFC 2119

Scott Bradner, [RFC 2119: Key words for use in RFCs to Indicate Requirement Levels](#). Internet Engineering Task Force, 1997.

XML

Tim Bray, Jean Paoli, C.M. Sperberg-McQueen, and Eve Maler, editors. [Extensible Markup Language \(XML\) 1.0 \(Second Edition\)](#). World Wide Web Consortium, 2000.

XPtrFrame

Paul Grosso, Eve Maler, Jonathan Marsh, and Norman Walsh, editors. [XPointer Framework](#). World Wide Web Consortium, 2002.

XML-Names

Tim Bray, Dave Hollander, and Andrew Layman, editors. [Namespaces in XML](#). World Wide Web Consortium, 1999.

Appendix B. Non-Normative References (Non-Normative)

XPointer

Steve DeRose, Eve Maler, and Ron Daniels, editors. [XPointer xpointer\(\) Scheme](#). World Wide Web Consortium, 2002. *Work in progress*.