[MS-XMLNS]:
Microsoft XML Namespaces Standards Support Document

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.

- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.

- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.

- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft Open Specification Promise or the Community Promise. If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.

- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.

- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
03/17/2010	0.1	New	Released new document.
03/26/2010	1.0	None	Introduced no new technical or language changes.
05/26/2010	1.2	None	Introduced no new technical or language changes.
09/08/2010	1.3	Major	Significantly changed the technical content.
10/13/2010	1.4	Minor	Clarified the meaning of the technical content.
02/10/2011	2.0	No change	Introduced no new technical or language changes.
02/22/2012	3.0	Major	Significantly changed the technical content.
07/25/2012	3.1	Minor	Clarified the meaning of the technical content.

Table of Contents

1 Introduction

This document describes the level of support provided by the Microsoft XML Core Services (MSXML) 3.0 and 6.0 for the *Namespaces in XML 1.0 (Third Edition)* [XMLNS] W3C Recommendation 8 December 2009.

Windows® Internet Explorer® 7, Windows® Internet Explorer® 8, Windows® Internet Explorer® 9, and Windows® Internet Explorer® 10 support Namespaces in XML using the *Extensible Markup Language (XML) 1.0 (Fourth Edition)* [XML], W3C Recommendation 16 August 2006, edited in place 29 September 2006.

The [XMLNS] specification may contain guidance for authors of webpages and browser users, in addition to user agents (browser applications). Statements found in this document apply only to normative requirements in the specification targeted to user agents, not those targeted to authors.

1.1 Glossary

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

References to Microsoft Open Specifications documentation do not include a publishing year because links are to the latest version of the technical documents, which are updated frequently. References to other documents include a publishing year when one is available.

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.rfc-editor.org/rfc/rfc2119.txt

[RFC2141] Moats, R., "URN Syntax", RFC 2141, May 1997, http://www.rfc-editor.org/rfc/rfc2141.txt

[XML] World Wide Web Consortium, "Extensible Markup Language (XML) 1.0 (Fourth Edition)", W3C Recommendation, August 2006, http://www.w3.org/TR/2006/REC-xml-20060816/

[XMLNS] Bray, T., Hollander, D., Layman, A., et al., Eds., "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, http://www.w3.org/TR/2009/REC-xml-names-20091208/

1.2.2 Informative References

None.

1.3 Microsoft Implementations

Throughout this document, Microsoft XML Core Services (MSXML) 3.0 is referred to as *MSXML3* and Microsoft XML Core Services (MSXML) 6.0 is referred to as *MSXML6*.

MSXML3 is the only version of MSXML that is implemented in Windows® Internet Explorer® 7 and Windows® Internet Explorer® 8. Both MSXML3 and MSXML6 are implemented in Windows® Internet Explorer® 9 and Windows® Internet Explorer® 10.

1.4 Standards Support Requirements

To conform to [XMLNS], a user agent must implement all required portions of the specification. Any optional portions that have been implemented must also be implemented as described by the specification. Normative language is usually used to define both required and optional portions. (For more information, see [RFC2119].)

The following table lists the sections of [XMLNS] and whether they are considered normative or informative.

Sections	Normative/Informative
1	Informative
2-8	Normative
Appendices A-F	Informative

1.5 Notation

The following notations are used in this document to differentiate between notes of clarification, variation from the specification, and extension points.

Notation	Explanation
C####	Identifies a clarification of ambiguity in the target specification. This includes imprecise statements, omitted information, discrepancies, and errata. This does not include data formatting clarifications.
V####	Identifies an intended point of variability in the target specification such as the use of MAY, SHOULD, or RECOMMENDED. (See [RFC2119].) This does not include extensibility points.
E####	Identifies extensibility points (such as optional implementation-specific data) in the target specification, which can impair interoperability.

For document mode and browser version notation, see section 1.3.

2 Standards Support Statements

This section contains a full list of variations, clarifications, and extension points in the Microsoft implementation of [XMLNS].

- Section 2.1 includes only those variations that violate a MUST requirement in the target specification.

- Section 2.2 describes further variations from MAY and SHOULD requirements.

- Section 2.3 identifies variations in error handling.

- Section 2.4 identifies variations that impact security.

2.1 Normative Variations

The following subsections detail the normative variations from MUST requirements in [XMLNS].

2.1.1 [NamespacesXML1.1] Section 3, Declaring Namespaces

C0004:

The specification states:

```
The prefix xmlns is used only to declare namespace bindings and is by definition
bound to the namespace name http://www.w3.org/2000/xmlns/. It MUST NOT be declared. Other
prefixes MUST NOT be bound to this namespace name, and it MUST NOT be declared as the default
namespace. Element names MUST NOT have the prefix
xmlns.
```

MSXML3 and MSXML6

Other prefixes can be bound to the **namespace** name `http://www.w3.org/2000/xmlns/`.

C0005:

The specification states:

```
All other prefixes beginning with the three-letter sequence x, m, l, in any case
combination, are reserved. This means that:
• users SHOULD NOT use them except as defined by later specifications
• processors MUST NOT treat them as fatal errors.
```

MSXML3

The namespace prefix is not allowed to start with the reserved string `"xml"`, in any case combination.

MSXML6

Namespace* is allowed with the reserved string `"xml"`.

2.2 Clarifications

The following subsections identify clarifications to recommendations made by [XMLNS].

2.2.1 [NamespacesXML1.1] Section 3, Declaring Namespaces

C0001:

The specification states:

```
Definition: A namespace (or more precisely, a namespace binding) is declared using
a family of reserved attributes. Such an attribute's name must either be xmlns or
begin xmlns:. These attributes, like any other XML attributes, may be provided
directly or by default.
```

MSXML3 and MSXML6

The namespace attributes can only be provided directly and not by default.

C0002:

The specification states:

```
The attribute's normalized value MUST be either a URI reference — the namespace
name identifying the namespace — or an empty string. The namespace name, to serve
its intended purpose, SHOULD have the characteristics of uniqueness and
persistence. It is not a goal that it be directly usable for retrieval of a schema
(if any exists). Uniform Resource Names [RFC2141] is an example of a syntax that is
designed with these goals in mind. However, it should be noted that ordinary URLs
can be managed in such a way as to achieve these same goals.
```

MSXML3 and MSXML6

The normalized value of the **namespace** declaration can only be set as the value of default namespace `xmlns` but not any other specific namespace such as `xmlns:xxx`.

C0003:

The specification states:

```
The prefix xml is by definition bound to the namespace name
http://www.w3.org/XML/1998/namespace. It MAY, but need not, be declared, and MUST
NOT be bound to any other namespace name. Other prefixes MUST NOT be bound to this namespace
name, and it MUST NOT be declared as the default namespace.
```

MSXML3 and MSXML6

The following clarifications apply:

- The prefix **xml** cannot be declared even if the **namespace** name to bind to is declared.

- Other prefixes can be bound to the **namespace** name, and it can be declared as the default **namespace**.

2.2.2 [NamespacesXML1.1] Section 6.3, Uniqueness of Attributes

V0001:

The specification states:

```
In XML documents conforming to this specification, no tag may contain two
attributes which:
1.   have identical names, or
2.   have qualified names with the same local part and with prefixes which have
been bound to namespace names that are identical.

This constraint is equivalent to requiring that no element have two attributes with
the same expanded name.
```

MSXML3 and MSXML6

Two attributes with the same local part and with different prefixes are treated as different, even if
the two different prefixes are bound to the same namespace name.

2.3   Error Handling

There are no additional considerations for error handling.

2.4   Security

There are no additional security considerations.

3   Change Tracking

This section identifies changes that were made to the [MS-XMLNS] protocol document between the February 2012 and July 2012 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.

- An extensive rewrite, addition, or deletion of major portions of content.

- The removal of a document from the documentation set.

- Changes made for template compliance.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the language and formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical or language changes were introduced. The technical content of the document is identical to the last released version, but minor editorial and formatting changes, as well as updates to the header and footer information, and to the revision summary, may have been made.

Major and minor changes can be described further using the following change types:

- New content added.

- Content updated.

- Content removed.

- New product behavior note added.

- Product behavior note updated.

- Product behavior note removed.

- New protocol syntax added.

- Protocol syntax updated.

- Protocol syntax removed.

- New content added due to protocol revision.

- Content updated due to protocol revision.

- Content removed due to protocol revision.

- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.

- Protocol syntax removed due to protocol revision.

- New content added for template compliance.

- Content updated for template compliance.

- Content removed for template compliance.

- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated.**

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.

- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1 Introduction	Updated document to remove beta tagging.	N	Content updated.

4 Index

C