[MS-RUBY]:
Internet Explorer Ruby Annotation 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 4
1.1 Glossary 4
1.2 References 4
1.2.1 Normative References 4
1.2.2 Informative References 4
1.3 Microsoft Implementations 4
1.4 Standards Support Requirements 5
1.5 Notation 5

2 Standards Support Statements 7
2.1 Normative Variations 7
2.1.1 [W3C-RUBY] Section 2.4, The rtc element 7
2.1.2 [W3C-RUBY] Section 2.6, The rt element 7
2.1.3 [W3C-RUBY] Section 2.7, The rp element 8
2.2 Clarifications 8
2.2.1 [W3C-RUBY] Section 2.2, The ruby element 8
2.2.2 [W3C-RUBY] Section 2.3, The rbc element 8
2.2.3 [W3C-RUBY] Section 2.5, The rb element 9
2.2.4 [W3C-RUBY] Section 2.6, The rt element 9
2.3 Error Handling 9
2.4 Security 9

3 Change Tracking 10

4 Index 12

1 Introduction

This document describes the level of support provided by Windows® Internet Explorer® for the *Ruby Annotation* [W3C-Ruby], W3C Recommendation 31 May 2001. Internet Explorer displays webpages written in HTML.

The [W3C-Ruby] specification may contain guidance for authors of webpages and browser users, in addition to user agents (browser applications). This document considers only normative language from the specification that applies directly to user agents.

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

[W3C-Ruby] Sawicki, M., Suignard, M., Ishikawa, M., Durst, M., and Texin, T., "Ruby Annotation", W3C Recommendation 31 May 2001 (Markup errors corrected 25 June 2008), http://www.w3.org/TR/ruby/

1.2.2 Informative References

None.

1.3 Microsoft Implementations

The following Microsoft products implement some portion of [W3C-Ruby]:

- Windows® Internet Explorer® 7
- Windows® Internet Explorer® 8
- Windows® Internet Explorer® 9
- Windows® Internet Explorer® 10

In addition, each version of Windows® Internet Explorer® implements multiple document modes, which can vary individually in their support of the standard. The following table lists the document modes available in each version of Internet Explorer:

Browser Version	Documents Modes Supported
Internet Explorer 7	Quirks Mode Standards Mode
Internet Explorer 8	Quirks Mode IE7 Mode IE8 Mode
Internet Explorer 9	Quirks Mode IE7 Mode IE8 Mode IE9 Mode
Internet Explorer 10	Quirks Mode IE7 Mode IE8 Mode IE9 Mode IE10 Mode

Throughout this document, the document mode appears first followed by the browser version in parentheses. Only those document modes and browser versions for which there is a variation note will be listed. If the document mode is not listed, conformance to the specification can be assumed.

Note "Standards mode" in Internet Explorer 7 and "IE7 mode" in Internet Explorer 8 refer to the same document mode. "IE7 mode" is the preferred way of referring to this document mode across all versions of the browser.

1.4 Standards Support Requirements

To conform to [W3C-Ruby] 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 [W3C-Ruby] and whether they are considered normative or informative.

Sections	Normative/Informative
1-3	Normative
4	Informative
Appendix A-E	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####	This 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####	This 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####	Because the use of extensibility points (such as optional implementation-specific data) can impair interoperability, this profile identifies such points in the target specification.

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

2 Standards Support Statements

This section contains a full list of variations, clarifications, and extension points in the Microsoft implementation of [W3C-Ruby].

- 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 [W3C-Ruby].

2.1.1 [W3C-RUBY] Section 2.4, The rtc element

V0001:

The specification states:

```
The rtc (ruby text container) element serves as the container for rt elements in
the case of complex ruby markup. One or two rtc elements may appear inside a ruby
element to associate ruby texts with a single base text, represented by an rbc
element. More than two rtc elements MUST NOT appear inside a ruby element.
```

All Document Modes (All Versions)

All **rtc** elements within **ruby** elements are displayed.

2.1.2 [W3C-RUBY] Section 2.6, The rt element

V0002:

The specification states:

```
The rt element may contain inline elements or character data as its content, but
the ruby element is not allowed as its descendant element.
```

All Document Modes (All Versions)

The **ruby** element is allowed as a descendent element to the **rt** element.

V0003:

The specification states:

```
In complex ruby markup, the rbspan attribute allows an rt element to span multiple
rb elements. The value shall be an integer value greater than zero ("0"). The
default value of this attribute is one ("1"). The rbspan attribute should not be
used in simple ruby markup, and user agents should ignore the rbspan attribute when
```

```
it appears in simple ruby markup.
```

All Document Modes (All Versions)

The **rbspan** attribute is not supported in **rt** elements.

2.1.3 [W3C-RUBY] Section 2.7, The rp element

V0004:

The specification states:

```
The rp element can be used in the case of simple ruby markup to specify characters
that can denote the beginning and end of ruby text when user agents do not have
other ways to present ruby text distinctively from the base text. Parentheses (or
similar characters) can provide an acceptable fallback. In this situation, ruby
text will only degrade to be rendered inline and enclosed in the fallback
parentheses. This is the least inappropriate rendering under the condition that
only inline rendering is available. The rp element cannot be used with complex ruby
markup.
```

All Document Modes (All Versions)

The **rp** element is allowed within complex **ruby** elements.

2.2 Clarifications

The following subsections identify clarifications to recommendations made by [W3C-Ruby].

2.2.1 [W3C-RUBY] Section 2.2, The ruby element

C0001:

The specification states:

```
The ruby element is an inline (or text-level) element that serves as an overall
container. It contains either the rb, rt and optional rp elements (simple ruby
markup) or the rbc and rtc elements (complex ruby markup).
```

All Document Modes (All Versions)

The **dir** attribute is not supported in simple or complex **ruby** elements.

2.2.2 [W3C-RUBY] Section 2.3, The rbc element

C0002:

The specification states:

```
The rbc (ruby base container) element serves as the container for rb elements in
the case of complex ruby markup. Only one rbc element may appear inside a ruby
element.
```

All Document Modes (All Versions)

All **rbc** elements within **ruby** elements are displayed.

2.2.3 [W3C-RUBY] Section 2.5, The rb element

C0003:

The specification states:

```
The rb (ruby base) element serves to markup the base text. For simple ruby markup,
only one rb element may appear. For complex ruby markup, multiple rb elements may
appear inside an rbc element. Each rb element is associated with a corresponding rt
element, for fine-grained control of ruby presentation.The rb element may contain
inline elements or character data as its content, but the ruby element is not
allowed as its descendant element.
```

All Document Modes (All Versions)

The following clarifications apply:

- All **rb** elements within simple or complex **ruby** elements are displayed.
- No error handling occurs for an **rb** element with a descendant **ruby** element.

2.2.4 [W3C-RUBY] Section 2.6, The rt element

C0004:

The specification states:

```
The rt element is the markup for ruby text. For simple ruby markup, only one rt
element may appear. For complex ruby markup, multiple rt elements may appear inside
an rtc element, and each rt element contains the ruby text for the relevant base
text, represented by the corresponding rb element.
```

All Document Modes (All Versions)

All **rt** elements within simple or complex **ruby** elements are displayed.

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-RUBY] 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 the document to remove beta tagging.	N	Content updated.

4 Index

C

Change tracking 10

G

Glossary 4

I

Informative references 4
Introduction 4

N

Normative references 4

R

References
  informative 4
  normative 4

T

The rb element 9
The rbc element 8
The rp element 8
The rt element (section 2.1.2 7, section 2.2.4 9)
The rtc element 7
The ruby element 8
Tracking changes 10