[MS-RUBY]:

Internet Explorer Ruby Annotation Standards Support Document

Revision Summary

Date	Revision History	Revision Class	Comments
3/17/2010	0.1	New	Released new document.
3/26/2010	1.0	None	Introduced no new technical or language changes.
5/26/2010	1.2	None	Introduced no new technical or language changes.
9/8/2010	1.3	Major	Significantly changed the technical content.
10/13/2010	1.4	Minor	Clarified the meaning of the technical content.
2/10/2011	2.0	None	Introduced no new technical or language changes.
2/22/2012	3.0	Major	Significantly changed the technical content.
7/25/2012	3.1	Minor	Clarified the meaning of the technical content.
6/26/2013	4.0	Major	Significantly changed the technical content.
3/31/2014	4.0	None	No changes to the meaning, language, or formatting of the technical content.
1/22/2015	5.0	Major	Updated for new product version.
7/7/2015	5.1	Minor	Clarified the meaning of the technical content.
11/2/2015	5.1	None	No changes to the meaning, language, or formatting of the technical content.
1/20/2016	5.2	Minor	Clarified the meaning of the technical content.
3/22/2016	5.2	None	No changes to the meaning, language, or formatting of the technical content.
11/2/2016	5.2	None	No changes to the meaning, language, or formatting of the technical content.
3/14/2017	5.2	None	No changes to the meaning, language, or formatting of the technical content.
10/3/2017	5.2	None	No changes to the meaning, language, or formatting of the technical content.
2/22/2018	5.2	None	No changes to the meaning, language, or formatting of the technical content.
3/23/2018	5.2	None	No changes to the meaning, language, or formatting of the technical content.
8/28/2018	5.2	None	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1   Introduction

This document describes the level of support provided by Microsoft web browsers for the *Ruby Annotation* [W3C-Ruby], W3C Recommendation 31 May 2001. Internet Explorer displays web pages 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 defined in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2   References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

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.

[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] World Wide Web Consortium, "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 web browser versions implement some portion of [W3C-Ruby]:

- Windows Internet Explorer 7

- Windows Internet Explorer 8

- Windows Internet Explorer 9

- Windows Internet Explorer 10

- Internet Explorer 11

- Internet Explorer 11 for Windows 10

- Microsoft Edge

Each browser version may implement multiple document rendering modes. The modes vary from one another in support of the standard. The following table lists the document modes supported by each browser version.

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
Internet Explorer 11	Quirks Mode IE7 Mode IE8 Mode IE9 Mode IE10 Mode IE11 Mode
Internet Explorer 11 for Windows 10	Quirks Mode IE7 Mode IE8 Mode IE9 Mode IE10 Mode IE11 Mode
Microsoft Edge	EdgeHTML Mode

For each variation presented in this document there is a list of the document modes and browser versions that exhibit the behavior described by the variation. All combinations of modes and versions that are not listed conform to the specification. For example, the following list for a variation indicates that the variation exists in three document modes in all browser versions that support these modes:

Quirks Mode, IE7 Mode, and IE8 Mode (All Versions)

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 all variations and clarifications for the Microsoft implementation of [W3C-Ruby].

- Section 2.1 describes normative variations from the MUST requirements of the specification.

- Section 2.2 describes clarifications of the MAY and SHOULD requirements.

- Section 2.3 considers error handling aspects of the implementation.

- Section 2.4 considers security aspects of the implementation.

2.1 Normative Variations

The following subsections describe normative variations from the MUST requirements of [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 describe clarifications of the MAY and SHOULD requirements of [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 error handling considerations.

2.4 Security

There are no additional security considerations.

3 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

4   Index

C