Internationalized Resource Identifiers	M. Duerst
(iri)	Aoyama Gakuin University
Internet-Draft	L. Masinter
Intended status: BCP	Adobe
Expires: February 15, 2012	August 14, 2011

TOC

Guidelines for Internationalized Resource Identifiers with Bi-directional Characters (Bidi IRIs) draft-ietf-iri-bidi-guidelines-00

Abstract

This specification gives guidelines for selection, use, presentation of International Resource Identifiers (IRI) which include characters with in inherent right-to-left (rtl) writing direction.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as “work in progress.”

This Internet-Draft will expire on February 15, 2012.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1. **Introduction**
   1.1. **Notation**
2. **Logical Storage and Visual Presentation**
3. **Bidi IRI Structure**

4. Input of Bidi IRIs
5. Examples
6. IANA Considerations
7. Security Considerations
8. Acknowledgements
9. References
9.1. Normative References
9.2. Informative References
§ Authors' Addresses

1. Introduction

TOC

Some UCS characters, such as those used in the Arabic and Hebrew scripts, have an inherent right-to-left (rtl) writing direction. IRIs containing these characters (called bidirectional IRIs or Bidi IRIs) require additional attention because of the non-trivial relation between logical representation (used for digital representation and for reading/spelling) and visual representation (used for display/printing).

Because of the complex interaction between the logical representation, the visual representation, and the syntax of a Bidi IRI, a balance is needed between various requirements. The main requirements are

1. user-predictable conversion between visual and logical representation;
2. the ability to include a wide range of characters in various parts of the IRI; and
3. minor or no changes or restrictions for implementations.

1.1. Notation

TOC

In this document, Bidi Notation is used for bidirectional examples: Lower case letters stand for Latin letters or other letters that are written left to right, whereas upper case letters represent Arabic or Hebrew letters that are written right to left.

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in **[RFC2119]**.

2. Logical Storage and Visual Presentation

TOC

When stored or transmitted in digital representation, bidirectional IRIs MUST be in full logical order and MUST conform to the IRI syntax rules (which includes the rules relevant to their scheme). This ensures that bidirectional IRIs can be processed in the same way as other IRIs.

Bidirectional IRIs MUST be rendered by using the Unicode Bidirectional Algorithm **[UNIV6]**, **[UNI9]**. Bidirectional IRIs MUST be rendered in the same way as they would be if they were in a left-to-right embedding; i.e., as if they were preceded by U+202A, LEFT-TO-RIGHT EMBEDDING (LRE), and followed by U+202C, POP DIRECTIONAL FORMATTING (PDF). Setting the embedding direction can also be done in a higher-level protocol (e.g., the dir='ltr' attribute in HTML).

There is no requirement to use the above embedding if the display is still the same without the embedding. For example, a bidirectional IRI in a text with left-to-right base directionality (such as used for English or Cyrillic) that is preceded and followed by whitespace and strong left-to-right characters does not need an embedding. Also, a bidirectional relative IRI reference that only contains strong right-to-left characters and weak characters and that starts and ends with a strong right-to-left character and appears in a text with right-to-left base directionality (such as used for Arabic or Hebrew) and is preceded and followed by

whitespace and strong characters does not need an embedding.

In some other cases, using U+200E, LEFT-TO-RIGHT MARK (LRM), may be sufficient to force the correct display behavior. However, the details of the Unicode Bidirectional algorithm are not always easy to understand. Implementers are strongly advised to err on the side of caution and to use embedding in all cases where they are not completely sure that the display behavior is unaffected without the embedding.

The Unicode Bidirectional Algorithm (**[UNI9]**, section 4.3) permits higher-level protocols to influence bidirectional rendering. Such changes by higher-level protocols MUST NOT be used if they change the rendering of IRIs.

The bidirectional formatting characters that may be used before or after the IRI to ensure correct display are not themselves part of the IRI. IRIs MUST NOT contain bidirectional formatting characters (LRM, RLM, LRE, RLE, LRO, RLO, and PDF). They affect the visual rendering of the IRI but do not appear themselves. It would therefore not be possible to input an IRI with such characters correctly.

---

3. Bidi IRI Structure

TOC

The Unicode Bidirectional Algorithm is designed mainly for running text. To make sure that it does not affect the rendering of bidirectional IRIs too much, some restrictions on bidirectional IRIs are necessary. These restrictions are given in terms of delimiters (structural characters, mostly punctuation such as "@", ".", ":", and "/") and components (usually consisting mostly of letters and digits).

The following syntax rules from the ABNF of **[RFC3987bis]** correspond to components for the purpose of Bidi behavior: iuserinfo, ireg-name, isegment, isegment-nz, isegment-nz-nc, ireg-name, iquery, and ifragment.

Specifications that define the syntax of any of the above components MAY divide them further and define smaller parts to be components according to this document. As an example, the restrictions of **[RFC3490]** on bidirectional domain names correspond to treating each label of a domain name as a component for schemes with ireg-name as a domain name. Even where the components are not defined formally, it may be helpful to think about some syntax in terms of components and to apply the relevant restrictions. For example, for the usual name/value syntax in query parts, it is convenient to treat each name and each value as a component. As another example, the extensions in a resource name can be treated as separate components.

For each component, the following restrictions apply:

1. A component SHOULD NOT use both right-to-left and left-to-right characters.
2. A component using right-to-left characters SHOULD start and end with right-to-left characters.

The above restrictions are given as "SHOULD"s, rather than as "MUST"s. For IRIs that are never presented visually, they are not relevant. However, for IRIs in general, they are very important to ensure consistent conversion between visual presentation and logical representation, in both directions.

Note:
In some components, the above restrictions may actually be strictly enforced. For example, **[RFC3490]** requires that these restrictions apply to the labels of a host name for those schemes where ireg-name is a host name. In some other components (for example, path components) following these restrictions may not be too difficult. For other components, such as parts of the query part, it may be very difficult to enforce the restrictions because the values of query parameters may be arbitrary character sequences.

If the above restrictions cannot be satisfied otherwise, the affected component can always be mapped to URI notation using the general percent-encoding of IRI components, as described in **[RFC3987bis]**. Please note that the whole component has to be mapped (see also Example 9 below).

4. Input of Bidi IRIs

TOC

Bidi input methods MUST generate Bidi IRIs in logical order while rendering them according to **Section 2**. During input, rendering SHOULD be updated after every new character is input to avoid end-user confusion.

5. Examples

TOC

This section gives examples of bidirectional IRIs, in Bidi Notation. It shows legal IRIs with the relationship between logical and visual representation and explains how certain phenomena in this relationship may look strange to somebody not familiar with bidirectional behavior, but familiar to users of Arabic and Hebrew. It also shows what happens if the restrictions given in **Section 3** are not followed. The examples below can be seen at **[BidiEx]**, in Arabic, Hebrew, and Bidi Notation variants.

To read the bidi text in the examples, read the visual representation from left to right until you encounter a block of rtl text. Read the rtl block (including slashes and other special characters) from right to left, then continue at the next unread ltr character.

Example 1: A single component with rtl characters is inverted:
Logical representation: "http://ab.CDEFGH.ij/kl/mn/op.html"
Visual representation: "http://ab.HGFEDC.ij/kl/mn/op.html"
Components can be read one by one, and each component can be read in its natural direction.

Example 2: More than one consecutive component with rtl characters is inverted as a whole:
Logical representation: "http://ab.CDE.FGH/ij/kl/mn/op.html"
Visual representation: "http://ab.HGF.EDC/ij/kl/mn/op.html"
A sequence of rtl components is read rtl, in the same way as a sequence of rtl words is read rtl in a bidi text.

Example 3: All components of an IRI (except for the scheme) are rtl. All rtl components are inverted overall:
Logical representation: "http://AB.CD.EF/GH/IJ/KL?MN=OP;QR=ST#UV"
Visual representation: "http://VU#TS=RQ;PO=NM?LK/JI/HG/FE.DC.BA"
The whole IRI (except the scheme) is read rtl. Delimiters between rtl components stay between the respective components; delimiters between ltr and rtl components don't move.

Example 4: Each of several sequences of rtl components is inverted on its own:
Logical representation: "http://AB.CD.ef/gh/IJ/KL.html"
Visual representation: "http://DC.BA.ef/gh/LK/JI.html"
Each sequence of rtl components is read rtl, in the same way as each sequence of rtl words in an ltr text is read rtl.

Example 5: Example 2, applied to components of different kinds:
Logical representation: "http://ab.cd.EF/GH/ij/kl.html"
Visual representation: "http://ab.cd.HG/FE/ij/kl.html"
The inversion of the domain name label and the path component may be unexpected, but it is consistent with other bidi behavior. For reassurance that the domain component really is "ab.cd.EF", it may be helpful to read aloud the visual representation following the bidi algorithm. After "http://ab.cd." one reads the RTL block "E-F-slash-G-H", which corresponds to the logical representation.

Example 6: Same as Example 5, with more rtl components:
Logical representation: "http://ab.CD.EF/GH/IJ/kl.html"
Visual representation: "http://ab.JI/HG/FE.DC/kl.html"
The inversion of the domain name labels and the path components may be easier to identify because the delimiters also move.

Example 7: A single rtl component includes digits:
Logical representation: "http://ab.CDE123FGH.ij/kl/mn/op.html"
Visual representation: "http://ab.HGF123EDC.ij/kl/mn/op.html"
Numbers are written ltr in all cases but are treated as an additional embedding inside a run

of rtl characters. This is completely consistent with usual bidirectional text.

Example 8 (not allowed): Numbers are at the start or end of an rtl component:
Logical representation: "http://ab.cd.ef/GH1/2IJ/KL.html"
Visual representation: "http://ab.cd.ef/LK/JI1/2HG.html"
The sequence "1/2" is interpreted by the bidi algorithm as a fraction, fragmenting the components and leading to confusion. There are other characters that are interpreted in a special way close to numbers; in particular, "+", "-", "#", "$", "%", ",", ".", and ":".

Example 9 (not allowed): The numbers in the previous example are percent-encoded:
Logical representation: "http://ab.cd.ef/GH%31/%32IJ/KL.html",
Visual representation: "http://ab.cd.ef/LK/JI%32/%31HG.html"

Example 10 (allowed but not recommended):
Logical representation: "http://ab.CDEFGH.123/kl/mn/op.html"
Visual representation: "http://ab.123.HGFEDC/kl/mn/op.html"
Components consisting of only numbers are allowed (it would be rather difficult to prohibit them), but these may interact with adjacent RTL components in ways that are not easy to predict.

Example 11 (allowed but not recommended):
Logical representation: "http://ab.CDEFGH.123ij/kl/mn/op.html"
Visual representation: "http://ab.123.HGFEDCij/kl/mn/op.html"
Components consisting of numbers and left-to-right characters are allowed, but these may interact with adjacent RTL components in ways that are not easy to predict.

6. IANA Considerations

TOC

This document makes no changes to IANA registries.

7. Security Considerations

TOC

Confusion can occur with bidirectional IRIs, if the restrictions in **Section 3** are not followed. The same visual representation may be interpreted as different logical representations, and vice versa. It is also very important that a correct Unicode bidirectional implementation be used.

8. Acknowledgements

TOC

This document was derived from **[RFC3987]** and **[RFC3987bis]** and the acknowledgments of those documents apply.

9. References

TOC

9.1. Normative References

TOC

[ASCII] American National Standards Institute, “Coded Character Set -- 7-bit American Standard Code for Information Interchange,” ANSI X3.4, 1986.

[ISO10646] International Organization for Standardization, “ISO/IEC 10646:2003: Information Technology - Universal Multiple-Octet Coded Character Set (UCS),” ISO Standard 10646, December 2003.

[RFC2119] **Bradner, S.**, “**Key words for use in RFCs to Indicate Requirement Levels**,” BCP 14, RFC 2119, March 1997 (**TXT**, **HTML**, **XML**).

[RFC3490] Faltstrom, P., Hoffman, P., and A. Costello, “**Internationalizing Domain Names in Applications (IDNA)**,” RFC 3490, March 2003 (**TXT**).

[RFC3491] Hoffman, P. and M. Blanchet, “**Nameprep: A Stringprep Profile for Internationalized Domain Names**

(IDN)," RFC 3491, March 2003 (TXT).

[RFC3987bis] Duerst, M., Masinter, L., and M. Suignard, "Internationalized Resource Identifiers (IRIs)," August 2011.

[UNI9] Davis, M., "The Bidirectional Algorithm," Unicode Standard Annex #9, March 2004.

[UNIV6] The Unicode Consortium, "The Unicode Standard, Version 6.0.0 (Mountain View, CA, The Unicode Consortium, 2011, ISBN 978-1-936213-01-6)," October 2010.

9.2. Informative References

TOC

[BidiEx] "Examples of bidirectional IRIs."

[RFC3987] Duerst, M. and M. Suignard, "Internationalized Resource Identifiers (IRIs)," RFC 3987, January 2005 (TXT).

Authors' Addresses

TOC

Martin Duerst
Aoyama Gakuin University
5-10-1 Fuchinobe
Sagamihara, Kanagawa 229-8558
Japan
Phone: +81 42 759 6329
Fax: +81 42 759 6495
Email: duerst@it.aoyama.ac.jp
URI: http://www.sw.it.aoyama.ac.jp/D%C3%BCrst/

Larry Masinter
Adobe
345 Park Ave
San Jose, CA 95110
U.S.A.
Phone: +1-408-536-3024
Email: masinter@adobe.com
URI: http://larry.masinter.net