Extensible Resource Identifier (XRI) Metadata V2.0

Committee Draft 01, 14 March 2005

Document identifier:
xri-metadata-V2.0-cd-01

Location:
http://docs.oasis-open.org/xri/xri/V2.0

Editor:
Drummond Reed, Cordance <drummond.reed@cordance.net>

Contributors:
Dave McAlpin, Epok <dave.mcalpin@epok.net>
Nat Sakimura, NRI <n-sakimura@nri.co.jp>
Mike Lindelsee, Visa International <mlindels@visa.com>
Gabe Wachob, Visa International <gwachob@visa.com>

Abstract:
This document is the normative technical specification for XRI metadata. It is a companion specification to *Extensible Resource Identifier (XRI) Syntax V2.0* **[XRISyntax]** and *Extensible Resource Identifier (XRI) Resolution V2.0* **[XRIResolution]**. For a non-normative introduction to the uses and features of XRIs, see the *Introduction to XRIs* **[XRIIntro]**.

Status:
This document was last revised or approved by the XRI Technical Committee on the above date. The level of approval is also listed above. Check the current location noted above for possible later revisions of this document. This document is updated periodically on no particular schedule.

Technical Committee members should send comments on this specification to the Technical Committee's email list. Others should send comments to the Technical Committee by using the "Send A Comment" button on the Technical Committee's web page at http://www.oasis-open.org/committees/xri.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (http://www.oasis-open.org/committees/xri/ipr.php.

The non-normative errata page for this specification is located at http://www.oasis-open.org/committees/xri.

Table of Contents

1 Introduction

1.1 Purpose of this Specification

[XRISyntax] establishes five global context symbol (GCS) characters that may be used to begin an XRI authority segment for the purpose of expressing the abstract global context of an identifier. One of these GCS characters, "$", is reserved for identifiers for which the authority is a standards specification.

A second key XRI feature, cross-references (see section 2.2.2 of **[XRISyntax]**), allows XRIs to be embedded within other XRIs in order to share identifiers across contexts. This syntactic feature allows XRIs in the $ namespace to be used as metadata to describe an XRI itself. For example, the following XRI includes a cross-reference with version metadata.

```
xri://@example*resource*($v/3)
                        \----/ cross-reference with XRI metadata
```

The purpose of this specification is to define a set of XRIs in the $ namespace that function as *identifier metadata*—attributes that may be used describe an identifier itself, as opposed to attributes of the resource it identifies. This specification defines four such types of identifier metadata:

- *Language metadata* that specifies the human language in which an identifier is intended to be interpreted.
- *Date/time metadata* that specifies the point in time an identifier was established.
- *Version metadata* that specifies the identifier's position in a sequence of identifiers for the same logical resource.
- *Annotation metadata* that allows XRI producers to add annotations to XRIs or XRI segments.

The definition of each metadata type also specifies comparison rules, significance in resolution, and any other special processing rules specific to that type.

1.2 Related Specifications

This document is a companion specification to *XRI Syntax 2.0* **[XRISyntax]** and *XRI Resolution 2.0* **[XRIResolution]**. **[XRISyntax]** specifies the normative ABNF and processing rules for XRIs (including the GCS "$" character). **[XRIResolution]** specifies both a standard and a trusted resolution protocol for XRIs. It also establishes a branch of the $ namespace, "$res", for identifiers used by XRI resolution.

The *Introduction to XRIs* **[XRIIntro]** provides a non-normative introduction to the uses and features of XRIs.

1.3 Design Considerations

This section briefly enumerates the primary design criteria for XRI metadata.

1.3.1 Support for Interoperability

The primary factor in selecting the metadata types included in this specification is the degree to which they are useful across multiple domains and applications. The four types of identifier metadata included in this specification have appeared repeatedly in use cases considered by the XRI TC since its inception.

1.3.2 Simplicity and Compactness

To keep XRI metadata as simple and as lightweight as possible, all top-level $ identifiers defined in this specification are single ASCII letters. (Note that this is not a requirement of $ identifiers defined in other specifications; for example, the XRI Resolution specification **[XRIResolution]** establishes a multi-character identifier, "$res", for use in XRI resolution.) Because $ identifiers match the xri-authority production in section 2.2.1 of **[XRISyntax]**, they are normalized to lowercase per the recommendations in section 2.5.5 of **[XRISyntax]**. In addition, although this specification does not explicitly designate these $ identifiers as persistent using "!" syntax (see section 2.2.3 of **[XRISyntax]**), they should be considered effectively persistent.

1.3.3 Minimal Processing Rules

The final design principle is that $ metadata defined in this specification (as opposed to extensions, as described in section 6) must require only minimal processing rules to be useful. Note that these processing rules, including the order in which $ cross-references are placed in an XRI, are specific to each type of $ metadata.

1.4 Terminology and Notation

1.4.1 Keywords

The key words "MUST","MUST NOT","REQUIRED","SHALL","SHALL NOT","SHOULD","SHOULD NOT","RECOMMENDED","MAY", and "OPTIONAL" in this document are to be interpreted as described in **[RFC2119]**. When these words are not capitalized in this document, they are meant in their natural language sense.

1.4.2 Syntax Notation

This specification uses the same syntax notation employed in **[XRISyntax]**, i.e., Augmented Backus-Naur Form (ABNF), defined in **[RFC2234]**. The following core ABNF productions are used by this specification as defined by section 6.1 of **[RFC2234]**: ALPHA, CR, CTL, DIGIT, DQUOTE, HEXDIG, LF, OCTET, and SP.

Also note that because the prefix "xri://" is optional in absolute XRIs (which can be recognized by a leading global context symbol—see section 2.2.1.1 of **[XRISyntax]**), some example XRIs may be shown without this prefix.

```
ABNF that is reproduced from an external specification looks like this.
```

```
ABNF that is defined by this specification looks like this.
```

1.4.3 Glossary

The normative glossary for the XRI 2.0 suite of specifications is in Appendix C of **[XRISyntax]**.

2 Language Metadata ($l)

2.1 Purpose

The purpose of the $l namespace is to provide a standard means of representing the human language in which an XRI, or a sub-segment of an XRI, is intended to be understood, interpreted, or pronounced. The many uses and applications of language tags are well articulated in *Tags for Identification of Languages* **[RFC3066]** and its successor, *Tags for Identifying Languages*, **[RFC3066bis]**. (Note that the latter is a work-in-progress and is subject to change.).

2.2 Namespace Restrictions

In the absence of an extension (see section 7), the identifier in the local path immediately following the XRI authority "$l" (i.e., "$l/…") MUST be a language tag conforming to the ABNF defined in Section 2.1 of **[RFC3066]**. For convenience, this ABNF is reproduced below:

```
Language-Tag      = Primary-subtag *( "-" Subtag )
Primary-subtag    = 1*8ALPHA

Subtag            = 1*8(ALPHA / DIGIT)
```

In a future version of this specification, the XRI Technical Committee intends to incorporate **[RFC3066bis]** when it is approved. The language tags in **[RFC3066bis]** are intended to be backward-compatible with **[RFC3066]**.

2.3 Normalization and Comparison

The rules for normalization of language tags are specified in **[RFC3066]**. For comparison, the general equivalence rules in section 6 of this XRI Metadata specification apply.

2.4 Significance in Resolution

$l language tags are significant in XRI resolution. See section 2.2.6 of **[XRIResolution]**.

2.5 Special Processing Rules

If an XRI contains $l metadata as a standalone sub-segment, the $l language tag MUST be interpreted as describing all subsequent sub-segments of that XRI until another $l cross-reference is encountered.

```
xri://@France*($l/fr)*pays*place
                \-----/ applies to sub-segments "*pays" and "*place"
xri://@($l/fr)*Français*pays*place
       \-----/ applies to all following sub-segments
```

However, if an XRI contains $l metadata within another cross-reference, the $l language tag MUST be interpreted as only applying within that cross-reference.

```
xri://@France*(($l/fr)/pays)*place
                \------------/ $l/fr applies only to "/pays"
xri://@(($l/fr)*Français)*pays*place
       \----------------/ $l/fr applies only to "Français"
```

3 Date/Time Metadata ($d)

3.1 Purpose

The purpose of the $d namespace is to provide a standard means of representing the date and time that an XRI, or a sub-segment of an XRI, was assigned to a resource.

3.2 Namespace Restrictions

In the absence of an extension (see section 7), the identifier in the local path immediately following the XRI authority "$d" (i.e., "$d/…") MUST be a date/time value conforming to the specifications in Section 3.2.7 of **[XMLSchema2]** for the simple XML datatype identified as http://www.w3.org/TR/xmlschema-2/#dateTime. In addition, to promote interoperability, this date/time value SHOULD use the canonical UTC "Z" time zone and SHOULD NOT use fractional seconds.

For convenience, the relevant text of section 3.2.7 of **[XMLSchema2]** is reproduced below.

The ·lexical space· of **dateTime** consists of finite-length sequences of characters of the form: `'-'? yyyy '-' mm '-' dd 'T' hh ':' mm ':' ss ('.' s+)? (zzzzzz)?`, where

- '-'? *yyyy* is a four-or-more digit optionally negative-signed numeral that represents the year; if more than four digits, leading zeros are prohibited, and '0000' is prohibited (see the Note above (§3.2.7); also note that a plus sign is **not** permitted);
- the remaining '-'s are separators between parts of the date portion;
- the first *mm* is a two-digit numeral that represents the month;
- *dd* is a two-digit numeral that represents the day;
- 'T' is a separator indicating that time-of-day follows;
- *hh* is a two-digit numeral that represents the hour; '24' is permitted if the minutes and seconds represented are zero, and the **dateTime** value so represented is the first instant of the following day (the hour property of a **dateTime** object in the ·value space· cannot have a value greater than 23);
- ':' is a separator between parts of the time-of-day portion;
- the second *mm* is a two-digit numeral that represents the minute;
- *ss* is a two-integer-digit numeral that represents the whole seconds;
- '.' *s+* (if present) represents the fractional seconds;
- *zzzzzz* (if present) represents the timezone (as described below).

For example, 2002-10-10T12:00:00-05:00 (noon on 10 October 2002, Central Daylight Savings Time as well as Eastern Standard Time in the U.S.) is 2002-10-10T17:00:00Z, five hours later than 2002-10-10T12:00:00Z.

The definition of a timezone is as follows:

Timezones are durations with (integer-valued) hour and minute properties (with the hour magnitude limited to at most 14, and the minute magnitude limited to at most 59, except that if the hour magnitude is 14, the minute value must be 0); they may be both positive or both negative.

The lexical representation of a timezone is a string of the form: (('+' | '-') hh ':' mm) | 'Z', where

- hh is a two-digit numeral (with leading zeros as required) that represents the hours,
- mm is a two-digit numeral that represents the minutes,
- '+' indicates a nonnegative duration,
- '-' indicates a nonpositive duration.

The mapping so defined is one-to-one, except that '+00:00', '-00:00', and 'Z' all represent the same zero-length duration timezone, UTC; 'Z' is its canonical representation.

When a timezone is added to a UTC dateTime, the result is the date and time "in that timezone". For example, 2002-10-10T12:00:00+05:00 is 2002-10-10T07:00:00Z and 2002-10-10T00:00:00+05:00 is 2002-10-09T19:00:00Z.

3.3 Normalization and Comparison

The rules for normalization of date/time values are specified in **[XMLSchema2]**. Specifically, the rules for the canonical representation of a date/time value are reproduced below.

Except for trailing fractional zero digits in the seconds representation, '24:00:00' time representations, and timezone (for timezoned values), the mapping from literals to values is one-to-one. Where there is more than one possible representation, the canonical representation is as follows:

- The 2-digit numeral representing the hour must not be '24';
- The fractional second string, if present, must not end in '0';
- For timezoned values, the timezone must be represented with 'Z' (All timezoned dateTime values are UTC.).

When comparing two date/time values, the general equivalence rules in section 6 of this XRI Metadata specification apply. In addition, the equivalence rules specified in Section 3.2.7 of **[XMLSchema2]** SHOULD be applied when comparing date/time values.

3.4 Significance in Resolution

$d date/time tags are significant in XRI resolution. See section 2.2.6 of **[XRIResolution]**.

3.5 Special Processing Rules

If two or more XRIs are equivalent except for the value of a $d cross-reference, the rules for determining an order relation between them are specified in section 3.2.7.4 of **[XMLSchema2]**. For example, the following XRIs form an ordered set.

```
xri://@example/resource*($d/1994-11-05T08:15:30-05:00Z)
xri://@example/resource*($d/1995-01-15T23:45:06-05:00Z)
xri://@example/resource*($d/2004-09-23T11:52:45-05:00Z)
                        \----------------------------/ dateTime
```

The following XRIs do NOT form an ordered set because they are not equivalent outside of the $d cross-reference.

```
xri://@example/resource*($d/1994-11-05T08:15:30-05:00Z)
xri://@example/resource*($d/1995-01-15T23:45:06-05:00Z)*subresource
xri://@example/resource*($d/2004-09-23T11:52:45-05:00Z)/subresource
                        \----------------------------/ dateTime
```

4 Version Metadata ($v)

4.1 Purpose

The purpose of the $v namespace is to provide a standard means of representing the version of an XRI, or of a sub-segment of an XRI.

4.2 Namespace Restrictions

In the absence of an extension (see section 7), the identifier in the local path immediately following the XRI authority "$v" (i.e., "$v/…") MUST be a version value conforming to the following ABNF.

```
Version           = num-segment *( ( "." / "-" ) num-segment )
num-segment       = 1*DIGIT
```

This is a very simple versioning scheme consisting of only numeric segments that nonetheless permits any level of granularity and easy version comparison. To use date/time values as version values, see section 7.1 for a specific extension of the $v namespace.

4.3 Normalization and Comparison

Due to the simplicity of the default $v scheme, there are no special normalization rules.

For comparison, the general equivalence rules in section 6 apply. In addition, if two XRIs are equivalent except for the value of corresponding $v cross-references *that use the same $v namespace* (meaning either the default $v namespace or the same extension to the $v namespace), then the XRIs SHOULD NOT be considered equivalent; however applications MAY infer that these XRIs refer to the same resource in different states.

If two XRIs are equivalent except for the value of corresponding $v cross-references but *use different $v namespaces for that cross-reference* (i.e., one uses the default $v namespace and one uses an extension, or each uses a different extension), applications MAY infer that these XRIs refer to the same resource but MUST NOT infer that these XRIs represent different states, since they may actually be referring to the same state in two different ways.

For example, an application cannot make a direct state comparison between an XRI that uses the default $v namespace to specify a numeric version of a resource and an otherwise-identical XRI that uses the $v/($d) extension (defined in section 7.1) to specify a date/time version of the resource. Equivalence or ordering of XRIs that use different $v namespaces to describe versions can only be defined in the context of a specific mapping algorithm, domain, or application.

4.4 Significance in Resolution

$v version tags are significant in XRI resolution. See section 2.2.6 of **[XRIResolution]**.

4.5 Special Processing Rules

If two XRIs are equivalent except for the value of a $v cross-reference using the default numeric scheme, an order relation between these XRIs can be determined by comparing the value of each numeric segment beginning with the most significant (leftmost) and proceeding to the least significant (rightmost). The XRI containing the greater value in the first non-equivalent version segment is the later version. If one XRI contains more version segments than the other, and the

segments that exist in both XRIs (comparing left to right) are equivalent, the XRI with more version segments is the later version.

For example, the following XRIs form an ordered set.

```
xri://@example/resource*($v/3.04.2)
xri://@example/resource*($v/3.04.3)
xri://@example/resource*($v/3.05)
xri://@example/resource*($v/3.05.1)
xri://@example/resource*($v/3.05.1.7)
xri://@example/resource*($v/4)
                        \-----------/ version
```

The following XRIs do NOT form an ordered set because they are not equivalent outside of the $v cross-reference.

```
xri://@example/resource*($v/3.04.2)
xri://@example/resource*($v/3.04.2)*subresource
xri://@example/resource*($v/3.04.2)/subresource
                        \---------/ version
```

5 Annotation Metadata ($-)

5.1 Purpose

The purpose of the $- namespace is to provide a standard means for XRI producers to provide human-readable annotations in an XRI, or an XRI sub-segment, without affecting resolution or comparison. It is essentially the XRI equivalent of a comment or remark, a concept found in most programming and markup languages.

5.2 Namespace Restrictions

Because the $- namespace is reserved for human-readable annotations, there are no restrictions on this namespace other than that it must conform to valid XRI cross-reference syntax as defined in section 2.2.2 of **[XRISyntax]**.

5.3 Normalization and Comparison

Because they are intended for human-readable annotations, there are no normalization rules specified for $- cross-references.

For the purposes of comparison, cross-references that begin with $- are NOT significant and SHOULD be ignored. That is, if two XRIs are equivalent except for the presence of a $- cross-reference, the XRIs SHOULD be considered equivalent.

Note that this rule must be applied recursively to cross-references contained inside other cross-references.

5.4 Significance in Resolution

$- annotation tags are not significant in XRI resolution. See section 2.2.6 of **[XRIResolution]**.

5.5 Special Processing Rules

An XRI-aware application MAY make special use of an $- annotation when displaying an XRI to a human user. The application MAY attempt to decode a percent-encoded XRI as appropriate for the interface, particularly whitespace characters such as <space> (%20), in order to faithfully reproduce the intended presentation.

```
xri://!!1000*($-Example%20Corp.)!5678*($-West%20Coast)/resource
             \------------------/      \--------------/
display form:  "Example Corp."           "West Coast"
```

However, to help prevent semantic attacks (see section 8), an XRI-aware application SHOULD inform the user that this is only an annotation and SHOULD NOT lead the user to infer anything about the described XRI or XRI sub-segment that cannot be independently confirmed by the application.

6 Normalization and Comparison

If two XRIs contain XRI metadata describing one or more sub-segments, the general rules for determining XRI equivalence in section 2.5 of **[XRISyntax]** apply. In other words, two XRIs containing $ metadata of any type are equivalent if they are character-for-character equivalent after normalization.

If two XRIs are equivalent *except* for one or more cross-references containing XRI metadata defined as signficant for comparison, and that XRI metadata is of the same type in both XRIs, the Normalization and Comparison rules for that type of XRI metadata apply. Note that $- annotation metadata is explicitly defined as not significant for comparison (see section 5.3).

If two XRIs are equivalent except for one or more cross-references containing XRI metadata defined as signficant for comparison, and that XRI metadata is of different types in both XRIs, or if one XRI includes XRI metadata where the other does not, the rules for comparison are application-specific.

For example, if one XRI includes $l metadata to describe one or more sub-segments and another XRI does not include $l metadata to describe the same segments, it is up to the processing application to determine equivalence.

```
xri://@example*($l/en)*standards
                \-----/ language tag
xri://@example*standards
                same XRI with no language tag – only application-
                specific logic can determine equivalence
```

7 Extending XRI Metadata

The $ namespaces defined in this specification may be extended using the standard XRI extensibility mechanism, i.e., cross-references. This approach permits any authority, of any type, to extend any XRI namespace.

The general format of such an extension is:

```
$namespace/(xref/value)
```

Note that the local path portion of the extension ("/value") is optional—an extension may consist of only the cross-reference itself.

An extension MUST be globally qualified, i.e., it must be an XRI, IRI, or URI in absolute (not relative) form. For example, if an organization with the absolute XRI “@example” desired to extend the $v namespace to define a new alphabetical versioning scheme, an XRI using this extension might look like:

```
xri://@example/resource*($v/(@example/C.AB.F))
                           \---------------/ extension
```

“@example” could further extend its extension, e.g. to create an alphanumeric versioning scheme, by appending additional sub-segments to its extension XRI as follows:

```
xri://@example/resource*($v/(@example*alphanum/10A.2E))
                           \-----------------------/ extension
```

7.1 Extending the $v Namespace with $d

One example of a globally-defined extension is extending the $v namespace (section 4) with the $d namespace (section 3). Since date/time stamps are the default versioning values used in many repositories, including most file systems, this is a natural way to avoid redefining date/time syntax in the $v namespace.

To use a date/time value as a version, the $v namespace is extended using the $d namespace as follows:

```
$v/($d/value)
```

Example:

```
xri://@example/resource*($v/($d/1994-11-05T08:15:30-05:00Z))
                           \---------------------------/
                                $d extension of $v
```

8 Security and Data Protection

First, all security and data protection considerations in **[XRISyntax]** and **[XRIResolution]** apply to XRIs that include $ metadata.

Second, as a general rule, the addition of $ metadata defined in this specification to an XRI should not introduce new security and data protection considerations with the exception of semantic attacks.

8.1 Semantic Attacks

As discussed in section 3.5 of **[XRISyntax]**, XRIs can be subject to the same types of semantic attacks as URIs and IRIs. There are many variations of these attacks (e.g. malicious construction, usage of homographic characters, use of name variants), all of which are designed to trick an end user into believing a identifier represents one resource when in fact it resolves to another.

Because $ metadata adds another dimension to the expressiveness of XRIs, it can potentially be used as an additional element in semantic attacks. For example, if an application disregarded the normative instructions in section 5 and displayed $- annotation metadata to a user as if it were authoritative for an XRI (especially a persistent XRI that was otherwise not human-readable), this could be used to deceive the user into believing the XRI represented a different resource than it actually does.

For this reason, an application responsible for interpreting and displaying XRIs or XRI sub-segments to a user SHOULD NOT lead the user to infer anything about the described XRI or XRI sub-segment that cannot be independently confirmed by the application. In addition, the application SHOULD warn the user if the XRI contains $ metadata that the application does not understand or expect.

9 References

9.1 Normative

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

[RFC2234] D. H. Crocker and P. Overell, *Augmented BNF for Syntax Specifications: ABNF*, http://www.ietf.org/rfc/rfc2234.txt, RFC 2234, November 1997.

[RFC3066] H. Alvestrand, *Tags for the Identification of Languages,* http://www.ietf.org/rfc/rfc3066.txt, RFC 3066, January, 2001.

[XMLSchema2] P. Biron, A. Malhotra, *XML Schema Part 2: Datatypes Second Edition W3C Recommendation,* http://www.w3.org/TR/xmlschema-2/, October 2004.

[XRISyntax] D. Reed, D. McAlpin, *Extensible Resource Identifier (XRI) Syntax V2.0*, http://docs.oasis-open.org/xri/xri/V2.0/xri-syntax-V2.0-cd-01.pdf, March 2005.

[XRIResolution] G. Wachob, *Extensible Resource Identifier (XRI) Resolution V2.0*, http://docs.oasis-open.org/xri/xri/V2.0/xri-resolution-V2.0-cd-01.pdf, March 2005.

9.2 Informative

[RFC3066bis] A. Phillips, M. Davis, *Tags for Identifying Languages,* http://www.ietf.org/internet-drafts/draft-phillips-langtags-10.txt, Work-In-Progress, January 2005.

[XRIIntro] D. Reed. D. McAlpin, *Introduction to XRIs*, http://docs.oasis-open.org/xri/xri/V2.0/xri-intro-V2.0.pdf, Work-In-Progress, March 2005.

Appendix A. Collected XRI Metadata

Namespace	Description	Significant for Resolution and Comparison?
$l	Language metadata (language tags) per **[RFC3066]**	Yes
$d	Date/time metadata per **[XMLSchema2]**	Yes
$v	Version metadata using simple numeric hierarchy or a cross-reference to the $d namespace	Yes
$-	Annotation metadata (similar to XML comments)	No

Table 1: XRI Metadata Namespaces Summary

Appendix B. Acknowledgments

The editors would like to acknowledge the contributions of the OASIS XRI Technical Committee, whose voting members at the time of publication were:

- Geoffrey Strongin, Advanced Micro Devices
- Ajay Madhok, AmSoft Systems
- Jean-Jacques Dubray, Attachmate
- William Barnhill, Booz Allen and Hamilton
- Drummond Reed, Cordance Corporation
- Marc Le Maitre, Cordance Corporation
- Dave Mc Alpin, Epok
- Loren West, Epok
- Peter Davis, NeuStar
- Masaki Nishitani, Nomura Research
- Nat Sakimura, Nomura Research
- Tetsu Watanabe, Nomura Research
- Owen Davis, PlaNetwork
- Victor Grey, PlaNetwork
- Fen Labalme, PlaNetwork
- Mike Lindelsee, Visa International
- Gabriel Wachob, Visa International
- Dave Wentker, Visa International
- Bill Washburn, XDI.ORG

A special acknowledgement to Jerry Kindall (Epok) for a full editorial review.

Appendix C. Notices

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights.

Information on OASIS's procedures with respect to rights in OASIS specifications can be found at the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification, can be obtained from the OASIS President.

OASIS invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to implement this specification. Please address the information to the OASIS President.