

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: July 18, 2015

M. Douglass  
RPI  
January 14, 2015

**Support for Icalendar Relationships**  
**draft-douglass-ical-relations-04**

Abstract

This specification updates RELATED-TO and introduces new iCalendar properties LINK, STRUCTURED-CATEGORY and REFID to allow better linking and grouping of iCalendar components and related data.

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 July 18, 2015.

Copyright Notice

Copyright (c) 2015 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.

## Table of Contents

<a href="#">1.</a>	<a href="#">Introduction</a>	<a href="#">2</a>
<a href="#">1.1.</a>	<a href="#">Structured iCalendar relationships</a>	<a href="#">3</a>
<a href="#">1.2.</a>	<a href="#">Grouped iCalendar relationships</a>	<a href="#">3</a>
<a href="#">1.3.</a>	<a href="#">Structured category relationships</a>	<a href="#">3</a>
<a href="#">1.4.</a>	<a href="#">Linked relationships</a>	<a href="#">3</a>
<a href="#">1.5.</a>	<a href="#">Caching and offline use</a>	<a href="#">4</a>
<a href="#">1.6.</a>	<a href="#">Conventions Used in This Document</a>	<a href="#">4</a>
<a href="#">2.</a>	<a href="#">Reference Types</a>	<a href="#">4</a>
<a href="#">3.</a>	<a href="#">Link Relation Types</a>	<a href="#">5</a>
<a href="#">4.</a>	<a href="#">Redefined Relation Type Value</a>	<a href="#">5</a>
<a href="#">5.</a>	<a href="#">New Property Parameters</a>	<a href="#">8</a>
<a href="#">5.1.</a>	<a href="#">Rel</a>	<a href="#">8</a>
<a href="#">5.2.</a>	<a href="#">Gap</a>	<a href="#">8</a>
<a href="#">6.</a>	<a href="#">New Value Data Types</a>	<a href="#">9</a>
<a href="#">7.</a>	<a href="#">New Properties</a>	<a href="#">9</a>
<a href="#">7.1.</a>	<a href="#">Structured-Category</a>	<a href="#">9</a>
<a href="#">7.2.</a>	<a href="#">Link</a>	<a href="#">10</a>
<a href="#">7.3.</a>	<a href="#">Refid</a>	<a href="#">12</a>
<a href="#">8.</a>	<a href="#">Redefined RELATED-TO Property</a>	<a href="#">12</a>
<a href="#">8.1.</a>	<a href="#">RELATED-TO</a>	<a href="#">13</a>
<a href="#">9.</a>	<a href="#">Security Considerations</a>	<a href="#">15</a>
<a href="#">10.</a>	<a href="#">IANA Considerations</a>	<a href="#">15</a>
<a href="#">10.1.</a>	<a href="#">iCalendar Property Registrations</a>	<a href="#">15</a>
<a href="#">10.2.</a>	<a href="#">iCalendar Property Parameter Registrations</a>	<a href="#">15</a>
<a href="#">10.3.</a>	<a href="#">iCalendar Value Data Type Registrations</a>	<a href="#">15</a>
<a href="#">10.4.</a>	<a href="#">iCalendar RELTYPE Value Registrations</a>	<a href="#">16</a>
<a href="#">10.5.</a>	<a href="#">New Reference Type Registration</a>	<a href="#">16</a>
<a href="#">11.</a>	<a href="#">Acknowledgements</a>	<a href="#">16</a>
<a href="#">12.</a>	<a href="#">Normative References</a>	<a href="#">17</a>
<a href="#">Appendix A.</a>	<a href="#">Change log</a>	<a href="#">17</a>
	<a href="#">Author's Address</a>	<a href="#">18</a>

**[1. Introduction](#)**

Icalendar entities often need to be related to each other or to associated meta-data. These relationships can take the following forms

Structured iCalendar: Icalendar entities are related to each other in some structured way, for example as parent, sibling, before, after.

Grouped iCalendar: Icalendar entities are related to each other as a group. CATEGORIES are often used for this purpose but are problematic for application developers.

Douglass

Expires July 18, 2015

[Page 2]

Linked: Entities are linked to each other through typed references.

### **1.1. Structured iCalendar relationships**

The currently existing iCalendar [[RFC5545](#)] RELATED-TO property has no support for temporal relationships as used by standard project management tools.

The RELTYPE parameter is extended to take new values defining temporal relationships, a GAP parameter is defined to provide lead and lag values and RELATED-TO is extended to allow URI values. These changes allows the RELATED-TO property to define a richer set of relationships useful for project management.

### **1.2. Grouped iCalendar relationships**

This specification defines a new REFID property which allows arbitrary groups of entities to be associated with the same key value. The presence of a REFID property imparts no meaning to the event. It is merely a key to allow retrieval.

### **1.3. Structured category relationships**

The introduction of STRUCTURED-CATEGORY allows a more structured approach to categorization, with the possibility of namespaced and path-like values. Unlike REFID the category imparts some meaning. It is assumed that the value of this property will reference a well defined category.

### **1.4. Linked relationships**

The currently existing iCalendar standard [[RFC5545](#)] lacks a general purpose method for referencing additional, external information relating to calendar components.

This document proposes a method for referencing typed external information that can provide additional information about an iCalendar component. This new LINK property is closely aligned to the LINK header defined in [[RFC5988](#)]

The LINK property defines a typed reference or relation to external meta-data or related resources. By providing type and format information as parameters, clients and servers are able to discover interesting references and make use of them, perhaps for indexing or the presentation of interesting links for the user.

It is often necessary to relate calendar components. The current RELATED-TO property only allows for a UID which is inadequate for



many purposes. Allowing other types may help but might raise a number of backward compatibility issues. The link property can link components in different collections or even on different servers.

When publishing events it is useful to be able to refer back to the source of that information. The actual event may have been consumed from a feed or an ics file on a web site. A LINK property can provide a reference to the originator of the event.

Beyond the need to relate elements temporally, project management tools often need to be able to specify the relationships between the various events and tasks which make up a project. The LINK property provides such a mechanism.

The LINK property SHOULD NOT be treated as just another attachment. The ATTACH property is being extended to handle server-side management and stripping of inline data. Clients may choose to handle attachments differently as they are often an integral part of the message - for example, the agenda.

### **1.5. Caching and offline use**

To facilitate offline display the link type may identify important pieces of data which should be downloaded in advance.

In general, the calendar entity should be self explanatory without the need to download referenced meta-data such as a web page.

### **1.6. Conventions Used in This Document**

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC2119](#)].

## **2. Reference Types**

The actual reference value can take three forms specified by the type parameter

URI: The default type. This is a URI referring to the target.

UID: This allows for linking within a single collection and the value is assumed to be another component within that collection.

REFERENCE: An xpointer. In an XML environment it may be necessary to refer to an external XML artifact. The XPointer is defined in



[W3C.WD-xptr-xpointer-20021219] and allows addressing portions of XML documents.

### 3. Link Relation Types

[RFC5988] defines two form of relation types, registered and extension. Registered relation types are added to a registry defined by [RFC5988] while extension relation types are specified as unique unregistered URIs, (at least unregistered in the [RFC5988] registry).

The relation types defined here will be registered with IANA in accordance with the specifications in [RFC5988].

### 4. Redefined Relation Type Value

Relationship parameter type values are defined in [section 3.2.15. of \[RFC5545\]](#). This specification redefines that type to include the new temporal relationship values FINISHTOSTART, FINISHTOFINISH, STARTTOFINISH and STARTTOSTART. It also adds the DEPENDS-ON value to provide a link to an component upon which the current component depends.

Format Definition:

This property parameter is defined by the following notation:

```
reltypeparam      = "RELTYPE" "="  
                    ("PARENT"      ; Parent relationship - Default  
                      / "CHILD"     ; Child relationship  
                      / "SIBLING"    ; Sibling relationship  
                      / "DEPENDS-ON" ; refers to previous task  
                      / "REFID"      ; Relationship based on REFID  
                      / "STRUCTURED-CATEGORY"  
                        ; Relationship based on STRUCTURED-CATEGORY  
                      / "FINISHTOSTART" ; Temporal relationship  
                      / "FINISHTOFINISH" ; Temporal relationship  
                      / "STARTTOFINISH" ; Temporal relationship  
                      / "STARTTOSTART" ; Temporal relationship  
                      / iana-token    ; Some other IANA-registered  
                        ; iCalendar relationship type  
                      / x-name)      ; A non-standard, experimental  
                        ; relationship type
```

Description: This parameter can be specified on a property that references another related calendar component. The parameter may specify the hierarchical relationship type of the calendar component referenced by the property when the value is PARENT, CHILD or SIBLING. If this parameter is not specified on an





allowable property, the default relationship type is PARENT. Applications MUST treat x-name and iana-token values they don't recognize the same way as they would the PARENT value.

It defines the temporal relationship when the value is one of the project management standard relationships FINISHTOSTART, FINISHTOFINISH, STARTTOFINISH or STARTTOSTART. This property will be present in the predecessor entity and will refer to the successor entity. The GAP parameter specifies the lead or lag time between the predecessor and the successor. In the description of each temporal relationship below we refer to Task-A which contains and controls the relationship and Task-B the target of the relationship.

RELTYPE=PARENT: See [\[RFC5545\] section 3.2.15](#).

RELTYPE=CHILD: See [\[RFC5545\] section 3.2.15](#).

RELTYPE=SIBLING: See [\[RFC5545\] section 3.2.15](#).

RELTYPE=DEPENDS-ON: Indicates that the current calendar component depends on the referenced calendar component in some manner. For example a task may be blocked waiting on the other, referenced, task.

RELTYPE=REFID: Establishes a reference from the current component to components with a REFID property which matches the value given in the associated RELATED-TO property.

RELTYPE=STRUCTURED-CATEGORY: Establishes a reference from the current component to components with a STRUCTURED-CATEGORY property which matches the value given in the associated RELATED-TO property.

RELTYPE=FINISHTOSTART: Task-B cannot start until Task-A finishes. For example, when sanding is complete, painting can begin.

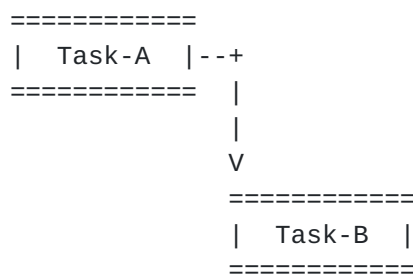


Figure 1: Finish to start relationship



RELTYPE=FINISHTOFINISH: Task-B cannot finish before Task-A is finished, that is the end of Task-A defines the end of Task-B. For example, we start the potatoes, then the meat then the peas but they should all be cooked at the same time.

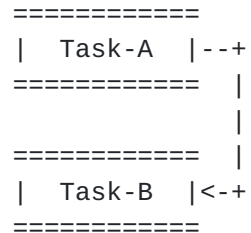


Figure 2: Finish to finish relationship

RELTYPE=STARTTOFINISH: The start of Task-A (which occurs after Task-B) controls the finish of Task-B. For example, ticket sales (Task-B) end when the game starts (Task-A).

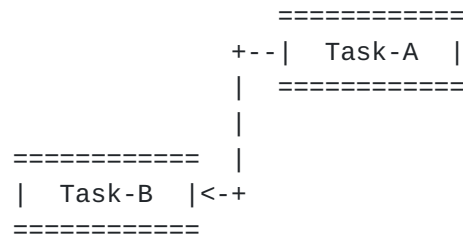


Figure 3: Start to finish relationship

RELTYPE=STARTTOSTART: The start of Task-A triggers the start of Task-B, that is Task-B can start anytime after Task-A starts.

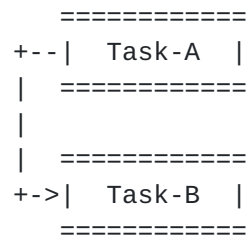


Figure 4: Start to start relationship



## 5. New Property Parameters

### 5.1. Rel

Parameter name: REL

Purpose: To specify the relationship of data referenced by a LINK property.

Format Definition:

This parameter is defined by the following notation:

```
relparam      = "REL" "="  
                ("SOURCE"      ; Link to source of this component  
                / DQUOTE uri DQUOTE  
                / x-name       ; Experimental reference type  
                / iana-token)  ; Other IANA registered type
```

Description: This parameter MUST be specified on all LINK properties, and defines the type of reference. This allows programs consuming this data to automatically scan for references they support. In addition to the values defined here any value defined in [[RFC5988](#)] may be used. There is no default relation type.

REL=SOURCE: identifies the source of the event information.

Registration: These relation types are registered in [[RFC5988](#)]

### 5.2. Gap

Parameter name: GAP

Purpose: To specify the length of the gap, positive or negative between two temporally related components.

Format Definition:

This parameter is defined by the following notation:

```
gapparam      = "GAP" "=" dur-value
```

Description: This parameter MAY be specified on the RELATED-TO property, and defines the duration of time between the predecessor and successor in an interval. When positive it defines the lag time between a task and its logical successor. When negative it defines the lead time.



An example of lag time might be if task A is "paint the room" and task B is "hang the drapes" then task A may be related to task B with RELTYPE=FINISHTOSTART with a gap long enough for the paint to dry.

An example of lead time might be to relate a 1 week task A to the end of task B with RELTYPE=STARTTOFINISH and a negative gap of 1 week so they finish at the same time.

## **6. New Value Data Types**

This specification defines the following new value types to be used with the VALUE property parameter:

UID VALUE=UID indicates that the associated value is the UID for a component.

REFERENCE VALUE=REFERENCE indicates that the associated value is an xpointer referencing an associated XML artifact.

## **7. New Properties**

### **7.1. Structured-Category**

Property name: STRUCTURED-CATEGORY

Purpose: This property defines the formal categories for a calendar component.

Value type: URI

Property Parameters: IANA, and non-standard parameters can be specified on this property.

Conformance: This property can be specified zero or more times in any iCalendar component.

Description: This property is used to specify formal categories or classifications of the calendar component. The values are useful in searching for a calendar component of a particular type and category.

Within the "VEVENT", "VTODO", or "VJOURNAL" calendar components, more than one formal category can be specified by using multiple properties.

This categorization is distinct from the more informal "tagging" of components provided by the existing CATEGORIES property. It is





expected that the value of the STRUCTURED-CATEGORY property will reference an external resource which provides information about the categorization.

In addition, a structured URI value allows for hierarchical categorization of events.

Format Definition:

This property is defined by the following notation:

```
structured-category = "STRUCTURED-CATEGORY" structcatparam ":" uri CRLF
```

```
structcatparam = *(  
    ;  
    ; The following is OPTIONAL,  
    ; and MAY occur more than once.  
    ;  
    (";" other-param)  
    ;  
    )
```

Example:

The following is an example of this property. It points to a server acting as the source for the calendar object.

```
STRUCTURED-CATEGORY:http://example.com/event-types/sports  
STRUCTURED-CATEGORY:http://example.com/event-types/arts/music  
STRUCTURED-CATEGORY:http://example.com/task-types/delivery
```

## [7.2.](#) Link

Property name: LINK

Purpose: This property provides a reference to external information about a component.

Value type: URI, TEXT or REFERENCE

Property Parameters: Non-standard, reference type or format type parameters can be specified on this property.

Conformance: This property MAY be specified in any iCalendar component.



Description: When used in a component the value of this property points to additional information related to the component. For example, it may reference the originating web server.

Format Definition:

This property is defined by the following notation:

```
link          = "LINK" linkparam ":" ( ":" uri ) /  
                (  
                  ";" "VALUE" "=" "REFERENCE"  
                  ":" text  
                )  
                CRLF
```

```
linkparam     = *(  
                ; the following is MANDATORY  
                ; and MAY occur more than once  
  
                (":" relparam) /  
  
                ; the following are MANDATORY  
                ; but MUST NOT occur more than once  
  
                (":" fmttypeparam) /  
                (":" labelparam) /  
                ; labelparam is defined in ...  
  
                ; the following is OPTIONAL  
                ; and MAY occur more than once  
  
                (":" xparam)  
                )
```

Example:

The following is an example of this property. It points to a server acting as the source for the calendar object.

```
LINK;REL=SOURCE;LABEL=The Egg:http://example.com/events
```



### **7.3. Refid**

Property name: REFID

Purpose: This property value acts as a key for associated iCalendar entities.

Value type: TEXT

Property Parameters: Non-standard parameters can be specified on this property.

Conformance: This property MAY be specified multiple times in any iCalendar component.

Description: The value of this property is a text identifier that allows associated components to be located or retrieved as a whole. For example all of the events in a travel itinerary would have the same REFID value.

Format Definition:

This property is defined by the following notation:

```
refid      = "REFID" refidparam ":" text CRLF
```

```
refidparam = *(  
    ; the following is OPTIONAL  
    ; and MAY occur more than once  
    (";" xparam)  
    )
```

Example:

The following is an example of this property.

```
REFID:itinerary-2014-11-17
```

## **8. Redefined RELATED-TO Property**



### **8.1. RELATED-TO**

Property name: RELATED-TO

Purpose: This property is used to represent a relationship or reference between one calendar component and another. The definition here extends the definition in [Section 3.8.4.5. of \[RFC5545\]](#) by allowing URI values and a GAP parameter.

Value type: URI or TEXT

Property Parameters: Non-standard, reference type, gap, value or format type parameters can be specified on this property.

Conformance: This property MAY be specified in any iCalendar component.

Description: By default or when VALUE=UID is specified, the property value consists of the persistent, globally unique identifier of another calendar component. This value would be represented in a calendar component by the "UID" property.

By default, the property value points to another calendar component that has a PARENT relationship to the referencing object. The "RELTYPE" property parameter is used to either explicitly state the default PARENT relationship type to the referenced calendar component or to override the default PARENT relationship type and specify either a CHILD or SIBLING relationship or a temporal relationship.

The PARENT relationship indicates that the calendar component is a subordinate of the referenced calendar component. The CHILD relationship indicates that the calendar component is a superior of the referenced calendar component. The SIBLING relationship indicates that the calendar component is a peer of the referenced calendar component.

The FINISHTOSTART, FINISHTOFINISH, STARTTOFINISH or STARTTOSTART relationships define temporal relationships as specified in the reltype parameter definition.

Changes to a calendar component referenced by this property can have an implicit impact on the related calendar component. For example, if a group event changes its start or end date or time, then the related, dependent events will need to have their start and end dates changed in a corresponding way. Similarly, if a PARENT calendar component is cancelled or deleted, then there is an implied impact to the related CHILD calendar components. This





property is intended only to provide information on the relationship of calendar components. It is up to the target calendar system to maintain any property implications of this relationship.

#### Format Definition:

This property is defined by the following notation:

```
related      = "RELATED-TO" relparam ( ":" text ) /  
              (  
                ";" "VALUE" "=" "UID"  
                ":" uid  
              )  
              (  
                ";" "VALUE" "=" "URI"  
                ":" uri  
              )  
              CRLF  
  
relparam     = *(  
              ;  
              ; The following are OPTIONAL,  
              ; but MUST NOT occur more than once.  
              ;  
              (";" reltypeparam) /  
              (";" gapparam) /  
              ;  
              ; The following is OPTIONAL,  
              ; and MAY occur more than once.  
              ;  
              (";" other-param)  
              ;  
              )
```

#### Example:

The following are examples of this property.

```
RELATED-TO:jsmith.part7.19960817T083000.xyzMail@example.com
```

```
RELATED-TO:19960401-080045-4000F192713-0052@example.com
```

```
RELATED-TO;VALUE=URI;RELTYPE=STARTTOFINISH:  
http://example.com/caldav/user/jb/cal/  
19960401-080045-4000F192713.ics
```



## **9. Security Considerations**

Applications using the LINK property need to be aware of the risks entailed in using the URIs provided as values. See [\[RFC3986\]](#) for a discussion of the security considerations relating to URIs.

The STRUCTURED-CATEGORY and redefined RELATED-TO property have the same issues in that values may be URIs.

## **10. IANA Considerations**

### **10.1. iCalendar Property Registrations**

The following iCalendar property names have been added to the iCalendar Properties Registry defined in [Section 8.3.2 of \[RFC5545\]](#)

Property	Status	Reference
LINK	Current	<a href="#">Section 7.2</a>
REFID	Current	<a href="#">Section 7.3</a>
STRUCTURED-CATEGORY	Current	<a href="#">Section 7.1</a>

### **10.2. iCalendar Property Parameter Registrations**

The following iCalendar property parameter names have been added to the iCalendar Parameters Registry defined in [Section 8.3.3 of \[RFC5545\]](#)

Parameter	Status	Reference
REL	Current	<a href="#">Section 5.1</a>
GAP	Current	<a href="#">Section 5.2</a>
STRUCTURED-CATEGORY	Current	<a href="#">Section 7.1</a>

### **10.3. iCalendar Value Data Type Registrations**

The following iCalendar property parameter names have been added to the iCalendar Value Data Types Registry defined in [Section 8.3.4 of \[RFC5545\]](#)



Value	Data Type	Status	Reference
UID		Current	<a href="#">Section 6</a>
REFERENCE		Current	<a href="#">Section 6</a>

#### **[10.4.](#) iCalendar RELTYPE Value Registrations**

The following iCalendar "RELTYPE" values have been added to the iCalendar Relationship Types Registry defined in [Section 8.3.8 of \[RFC5545\]](#)

Parameter	Status	Reference
REL	Current	<a href="#">Section 4</a>
DEPENDS-ON	Current	<a href="#">Section 4</a>
REFID	Current	<a href="#">Section 4</a>
STRUCTURED-CATEGORY	Current	<a href="#">Section 4</a>
FINISHTOSTART	Current	<a href="#">Section 4</a>
FINISHTOFINISH	Current	<a href="#">Section 4</a>
STARTTOFINISH	Current	<a href="#">Section 4</a>
STARTTOSTART	Current	<a href="#">Section 4</a>

#### **[10.5.](#) New Reference Type Registration**

The following link relation values have been added to the Reference Types Registry defined in [Section 6.2.2 of \[RFC5988\]](#)

Name	Status	Reference
SOURCE	Current	<a href="#">Section 5.1</a>

### **[11.](#) Acknowledgements**

The author would like to thank the members of the Calendaring and Scheduling Consortium technical committees and the following individuals for contributing their ideas, support and comments:

Adrian Apthorp, Cyrus Daboo, Marten Gajda

The authors would also like to thank the Calendaring and Scheduling Consortium for advice with this specification.



## **12. Normative References**

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC3688] Mealling, M., "The IETF XML Registry", [BCP 81](#), [RFC 3688](#), January 2004.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, [RFC 3986](#), January 2005.
- [RFC5545] Desruisseaux, B., "Internet Calendaring and Scheduling Core Object Specification (iCalendar)", [RFC 5545](#), September 2009.
- [RFC5988] Nottingham, M., "Web Linking", [RFC 5988](#), October 2010.
- [W3C.REC-xml-20060816]  
Bray, T., Paoli, J., Sperberg-McQueen, M., Maler, E., and F. Yergeau, "Extensible Markup Language (XML) 1.0 (Fourth Edition)", World Wide Web Consortium Recommendation REC-xml-20060816, August 2006,  
<<http://www.w3.org/TR/2006/REC-xml-20060816>>.
- [W3C.WD-xptr-xpointer-20021219]  
DeRose, S., Daniel, R., and E. Maler, "XPointer xpointer() Scheme", World Wide Web Consortium WD WD-xptr-xpointer-20021219, December 2002,  
<<http://www.w3.org/TR/2002/WD-xptr-xpointer-20021219>>.

## **Appendix A. Change log**

V04: 2014-11-18 MD

- o Add 5545 registrations.
- o Drop Q-name in favor of URI.
- o Change REFID value type to text.
- o Add STRUCTURED-CATEGORY property.
- o Add STRUCTURED-CATEGORY relationship to RELATED-TO.

V03: 2014-09-21 MD

- o Added Q-name specification.





- o Change RELATED-ID to REFID and change its value type to Q-name.
- o Add REFID relationship to RELATED-TO.

V02: 2013-11-22 MD

- o Added more description to GAP

V01: 2013-09-26 MD

- o Removed "Improved" from title
- o Fixed typos and examples
- o Removed gap param from link

2013-06-02 MD Initial version

#### Author's Address

Michael Douglass  
Rensselaer Polytechnic Institute  
110 8th Street  
Troy, NY 12180  
USA

Email: [dougln@rpi.edu](mailto:dougln@rpi.edu)  
URI: <http://www.rpi.edu/>

