GEOPRIV
Internet-Draft
Updates: 5222 (if approved)
Intended status: Standards Track
Expires: April 20, 2012

J. Winterbottom
M. Thomson
CommScope
R. Barnes
BBN Technologies
B. Rosen
NeuStar, Inc.
R. George
Huawei Technologies
October 18, 2011

# Specifying Civic Address Extensions in PIDF-LO
draft-ietf-geopriv-local-civic-02

## Abstract

New fields are occasionally added to civic addresses. A backwardly-compatible mechanism for adding civic address elements to the Geopriv civic address format is described. A formal mechanism for handling unsupported extensions when translating between XML and DHCP civic address forms is defined for entities that need to perform this translation. Intial extensions for some new elements are also defined. The LoST protocol mechanism that returns civic address element names used for validation of location information is clarified to require a namespace on each element.

Table of Contents

## 1. Introduction

The Geopriv civic location specifications ([RFC4776], [RFC5139]) define an XML and binary representtations for civic addresses that allow for the expression of civic addresses. Guidance for the use of these formats for the civic addresses in different countries is included in [RFC5774].

Subsequent to these specifications being produced, use cases for extending the civic address format with new elements have emerged. Extension elements do not readily fit existing elements, as recommended in [RFC5774].

The XML format for civic addresses [RFC5139] provides a mechanism that allows for the addition of standardized or privately understood elements. A similar facility for private extension is not provided for the DHCP format [RFC4776], though new specifications are able to define new CAtypes (civic address types).

A recipient of a civic address in either format currently has no option other than to ignore elements that it does not understand. This results in any elements that are unknown to that recipient being discarded if a recipient performs a translation between the two formats. In order for a new extension to be preserved through translation by any recipient, the recipient has to understand the extension and know how to correlate an XML element with a CAtype.

This document describes how new civic address elements are added. Extension always starts with the definition of XML elements. A mechanism for carrying the extension in the DHCP format is described. A new XML namespace containing a small number of additional civic elements is also defined and can be used as a template to illustrate how other extensions can be defined as required.

These mechanisms ensure that any translation between formats can be performed consistently and without loss of information. Translation between formats can occur without knowledge of every extension that is present.

The existing registry of numeric CAtypes is closed, and a new registry is created that advertises new namespaces and the associated civic elements to encourage maximum reuse.

These additions described in this document are backwardly compatible. Existing implementations may cause extension information to be lost, but the presence of extensions does not affect an implementation that conforms to either [RFC4776] or [RFC5139].

This document also normatively updates [RFC5222] to clarify that the namespace must be included with the element name in the lists of valid, invalid and not checked elements in the <locationValidation> part of a LoST response. While the LoST schema does not need to be changed, the example in the document is updated to show the namespaces in the lists.

### 1.1. Motivating Example

One instance where translation might be necessary is where a device receives location configuration using DHCP [RFC4776]. Conversion of DHCP information to an XML form is necessary if the device wishes to use the DHCP-provided information in a range of applications, including location-based presence services [RFC4079], and emergency calling [RFC5012].

```
 +--------+          +--------+         +-----------+
 | DHCP   |   DHCP   | Device |   XML   | Recipient | e.g., Presence
 | Server |--------->|        |-------->|           |       Agent
 +--------+          +--------+         +-----------+
```

Conversion Scenario

The Device that performs the translation between the DHCP and XML formats might not be aware of some of the extensions that are in use. Without knowledge of these extensions and how they are represented in XML, the Device is forced to discard them.

These extensions could be useful - or critical - to the ultimate consumers of this information. For instance, an extension element might provide a presence watcher with important information in locating the Device or an extension might be significant in choosing a particular call route.

### 1.2. Terminology

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].

## 2. Specifying Civic Address Extensions

The civic schema in [RFC5139] defines an ordered structure of elements that can be combined to describe a civic address. The XML extension point at the end of this sequence is used to extend the address.

New elements are defined in a new XML namespace [XMLNS]. This is true of address elements with significance within private or localized domains, as well as those that are intended for global applicability.

New elements SHOULD use the basic "caType" schema type defined in [RFC5139]. This type provides an optional "xml:lang" attribute.

For example, suppose the (fictitious) Central Devon Canals Authority wishes to introduce a new civic element called "bridge". The authority defines an XML namespace that includes a "bridge" element. The namespace needs to be a unique URI, for example "http://devon.canals.org.uk/civic".

A civic address that includes the new "bridge" element is shown in Figure 1.

```
<civicAddress xml:lang="en-GB"
     xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
     xmlns:cdc="http://devon.canals.org.uk/civic">
  <country>UK</country>
  <A1>Devon</A1>
  <A3>Monkokehampton</A3>
  <RD>Deckport</RD>
  <STS>Cross</STS>

  <cdc:bridge>21451338</cdc:bridge>

</civicAddress>
```

Figure 1: Extended Civic Address Example

An entity that receives this location information might not understand the extension address element. As long as the added element is able to be safely ignored, the remainder of the civic address can be used. The result is that the information is not as useful as it could be, but the added element does not prevent the use of the remainder of the address.

The address can be passed to other applications, such as a LoST server [RFC5222], without modification. If the application understands the added elements, it is able to make use of that information. For example, if this civic address is acquired using HELD [RFC5985], it can be included in a LoST request directly.

## 3. Translating Unsupported Elements

Unsupported civic address elements can be carried without consequence only as long as the format of the address does not change. When converting between the XML and DHCP formats, these unsupported elements are necessarily discarded: the entity performing the translation has no way to know the correct element to use in the target format.

All extensions MUST be defined using the mechanism described in this document. Extensions that use numeric CAtypes or other mechanisms cannot be safely translated between XML and DHCP representations.

An entity that does not support these extension mechanisms is expected to remove elements it doesn’t understand when performing conversions.

### 3.1. XML to DHCP Format Translation

Extensions to the XML format [RFC5139] are defined in a new XML namespace [XMLNS].

Extensions in the XML format can be added to a DHCP format civic address using an extension CAtype.

### 3.2. Extension Civic Address Type (CAtype)

The extension CAtype (CAtype code XX) [Note to IANA/RFC-Editor: please replace XX here and in the figure below with the assigned code] includes three values that uniquely identify the XML extension and its value: a namespace URI, the local name of the XML element, and the text content of that element. These three values are all included in the value of the CAtype, each separated by a single whitespace character.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  CAtype (XX)  |   Length      |  Namespace URI ...            .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.                  Namespace URI (continued) ...                .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Space (U+20) |           XML element local name ...          .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Space (U+20) |           Extension type value  ...           .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: XML Civic Address Extension CAtype

The content of a CAtype (after the CAtype code and length) is UTF-8 encoded Unicode text [RFC3629]. A maximum of 255 octets is allowed. Octets consumed by the namespace URI and local name reduce the space available for values.

This conversion only works for elements that have textual content and an optional "xml:lang" attribute. Elements with complex content or other attributes - aside from namespace bindings - MUST be ignored if they are not understood.

## 3.3. DHCP to XML Format Translation

The registration of a new CAtype following the process in [RFC4776] means that a recipient that does not know the equivalent XML is unable to produce a complete XML representation of the DHCP civic address. For this reason, this document ends the registration of new numeric CAtypes. No new registrations of numeric CAtypes can be made.

In lieu of making new numerical CAtype assignments, this document creates a new extensionCA type which is defined in a manner that lets new civic elements be described in DHCP form by carrying the name space and type name of the extension in parameters of the extensionCA type.

When converting to XML, the namespace prefix used for the extension element is selected by the entity that performs the conversion.

## 3.4. Conversion Example

The following example civic address contains two extensions:

```
<civicAddress xml:lang="en-US"
     xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
     xmlns:post="http://postsoftheworld.net/ns"
     xmlns:ap="http://example.com/airport/5.0">
  <country>US</country>
  <A1>CA</A1>

  <post:lamp>2471</post:lamp>
  <post:pylon>AQ-374-4(c)</post:pylon>

  <ap:airport>LAX</ap:airport>
  <ap:terminal>Tom Bradley</ap:terminal>
  <ap:concourse>G</ap:concourse>
  <ap:gate>36B</ap:gate>
</civicAddress>
```

Figure 3: XML Example with Multiple Extensions

This is converted to a DHCP form as follows:

```
country      = US
CAtype[0]    = en-US
CAtype[1]    = CA
CAtype[XX]  = http://postsoftheworld.net/ns lamp 2471
CAtype[XX]  = http://postsoftheworld.net/ns lamp AQ-374-4(c)
CAtype[XX]  = http://example.com/airport/5.0 airport LAX
CAtype[XX]  = http://example.com/airport/5.0 terminal Tom Bradley
CAtype[XX]  = http://example.com/airport/5.0 concourse G
CAtype[XX]  = http://example.com/airport/5.0 gate 36B
```

Figure 4: Converted DHCP Example with Multiple Extensions

## 4. CAtypes Registry

[RFC4776] created the CAtype registry. Among other things, this registry advertised available civic elements. While it has always been possible to use an extension namespace to define civic elements that are not in the CAtype registry, and this document does not change that, the registry is valuable to alert implementors of commonly used civic elements and provides guidance to clients of what elements they should suppport.

This document creates a new CAtype registry that differs from the original CAtype registry in two ways. The original registry is closed to new numeric CAtypes. The new registry adds a column to the registry called "Type". "Type" can have one of two values "A" or "B". Type A elements are intended for wide use with many applications and SHOULD be implemented by all clients unless the client is certain the element will not be encountered. Type "B" civic elements MAY be implemented by any client.

Type A civic elements require IETF review, while Type B elements only require an expert review.

## 5. Civic Extensions

We use this new extension method to define some additional civic address elements which are needed to correctly encode civic locations in several countries. The definition of these new civic address elements also serves as an example of how to define additional elements using the mechanisms described in this document.

### 5.1. Pole Number

In some areas, utility and lamp posts carry a unique identifier, which we call a pole number in this document. In some countries, the label on the lamp post also carries the local emergency service number, such as "110", encouraging callers to use the pole number to identify their location.

```
                  _.-----,===.
                 | |    (''''')
                 | |     '---'
                 | |
                 | |               ,---------,
                 | |     ,---,     |Emergency|
                 | |    /|,-.|----->| Number  |
                 | |   / |110|     '---------'
                 | |  /  |'-'|
                 |_|/    | 2 |     ,---------,
                 | |     | 1 |     |Lamp Post|
                 | |     | 2 |----->| Number  |
                 |-|     | 1 |     '---------'
                 | |\    | 0 |
                 | | \   | 1 |
                 | |  \  | 4 |
                 | |   \|,,,|
           _     | |
            ''-..|.|
                   ''--.._
                          ''--.._
```

Figure 5: Lamp post with emergency number

## 5.2. Mile Post

On some roads, and many trails, railroad rights of way and other linear features, a post with a mile or kilometer distance from one end of the feature may be found (a "milepost"). There are other cases of poles or markers with numeric indications that are not the same as a "house number" or street address number.

## 5.3. Street Type Prefix

The civic schema defined in [RFC5139] allows the definition of address "123 Colorado Boulevard", but it does not allow for the easy expression of "123 Boulevard Colorado". Adding a street-type prefix, allows street named in this manner to be more easily represented.

## 5.4. House Number Prefix

The civic schema defined in [RFC5139] provides house number suffix element, allowing one to express an address like "123A Main Street", but it does not contain a corresponding house number prefix. The house number prefix element allows the expression of address such as "Z123 Main Street".

## 5.5. XML Extension Schema

```
<?xml version="1.0"?>
<xs:schema
  targetNamespace="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr:ext"
  xmlns:ca="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:cae="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr:ext"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xs:import namespace="urn:ietf:params:xml:pidf:geopriv10:civicAddr"/>

  <!-- Post Number -->
  <xs:element name="PN" type="ca:caType"/>

  <!-- Milepost -->
  <xs:element name="MP" type="ca:caType"/>

  <!-- Street-Type prefix -->
  <xs:element name="STP" type="ca:caType"/>

  <!-- House Number Prefix -->
  <xs:element name="HNP" type="ca:caType"/>

</xs:schema>
```

## 5.6. Extension examples

```
<civicAddress xml:lang="en-US"
     xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
     xmlns:cae="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr:ext"
  <country>US</country>
  <A1>CA</A1>
  <A2>Sacramento</A2>
  <RD>I5</RD>
  <cae:MP>248</cae:MP>
  <cae:PN>22-109-689</cae:PN>
</civicAddress>
```

XML Example with Post Number and Mile Post

```
<civicAddress xml:lang="en-US"
     xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
     xmlns:cae="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr:ext"
  <country>US</country>
  <A1>CA</A1>
  <A2>Sacramento</A2>
  <RD>Colorado</RD>
  <HNO>223</HNO>
  <cae:STP>Boulevard</cae:STP>
  <cae:HNP>A</cae:HNP>
</civicAddress>
```

XML Example with Street prefix and House Number Prefix

## 6. Using Local Civic Extension with the LoST Protocol

One critical use of civic location information is in next generation emergency services applications, in particular call routing applications. In such cases location information is provided to a location-based routing service using the location to service transtion (LoST) protcol [RFC5222]. LoST is used to provide call routing information, but it is also used to validate location information to ensure that it can route to an emergency center when required.

LoST is an XML-based protocol and so the namespace extension mechansims described in this document do not impact LoST. When LoST is used for validation a <locationValidation> element is returned containing a list of valid, a list of invalid, and a list of unchecked civic elements. Figure 6 is an extract of the validation response in Figure 6 from [RFC5222].

```
<locationValidation>
    <valid>country A1 A3 A6</valid>
    <invalid>PC</invalid>
    <unchecked>HNO</unchecked>
</locationValidation>
```

Figure 6: Location Validation Example from LoST (RFC5222)

The RelaxNG schema in [RFC5222] requires the elements in each of these lists to be namespace qualified, which makes the example in Figure 6 from [RFC5222] in error. This issue is especially significant when local-civic extensions are used as the domain to which the extensions are attributed may impact their interpretation by the server or client. To ensure that local-civic extensions do not cause issues with LoST server and client implementations, all elements listed in a <valid>, <invalid>, or <unchecked> element MUST be qualified with a namespace. To illustrate this the extract above from figure 6 in [RFC5222] becomes Figure 7.

```
<locationValidation
       xmlns:ca="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr">
    <valid>ca:country ca:A1 ca:A3 ca:A6</valid>
    <invalid>ca:PC</invalid>
    <unchecked>ca:HNO</unchecked>
</locationValidation>
```

Figure 7: Corrected Location Validation Example

If validation request has also included the extensions defined in section Section 5 then the validation would response would look like Figure 8.

```
<locationValidation
       xmlns:ca="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
       xmlns:cae="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr:ext">
    <valid>ca:country ca:A1 ca:A3 ca:A6 cae:PN cae:STP</valid>
    <invalid>ca:PC</invalid>
    <unchecked>ca:HNO cae:MP cae:HNP</unchecked>
</locationValidation>
```

Figure 8: Corrected Location Validation Example

## 7. Security Considerations

This document defines a formal way to extend the existing Geopriv civic address schema. No security threats are introduced by this document.

Security threats applicable to the civic address formats are described in [RFC4776] (DHCP) and [RFC5139] (XML).

## 8. IANA Considerations

This document replaces the "CAtypes" registry established by [RFC4776].

### 8.1. CAtype Registration for Extensions

IANA has allocated a CAtype code of XX for the extension CAtype.

[[IANA/RFC-EDITOR: Please replace XX with the allocated CAtype]]

### 8.2. End of Numeric CAtype Registration

No further registration of numeric CAtypes is permitted, and the registry created as part of [RFC4776] replaced with the registry defined in this document. New registrations use the registration template in Section 8.5.

### 8.3. URN sub-namespace registration for 'urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr:ext'

This document calls for IANA to register a new XML namespace, as per the guidelines in [RFC3688].

URI: urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr:ext

Registrant Contact: IETF, GEOPRIV working group (geopriv@ietf.org), James Winterbottom (james.winterbottom@commscope.com).

XML:

```
      BEGIN
        <?xml version="1.0"?>
        <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
          "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
        <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
          <head>
            <title>GEOPRIV Civic Address Extensions</title>
          </head>
          <body>
            <h1>Additional Fields for GEOPRIV Civic Address</h1>
            <h2>urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr:ext</h2>
[[NOTE TO IANA/RFC-EDITOR: Please update RFC URL and replace XXXX
    with the RFC number for this specification.]]
            <p>See <a href="[[RFC URL]]">RFCXXXX</a>.</p>
          </body>
        </html>
      END
```

### 8.4. XML Schema Registration

This section registers an XML schema as per the procedures in [RFC3688].

URI: urn:ietf:params:xml:schema:pidf:geopriv10:civicAddr:ext

Registrant Contact: IETF, GEOPRIV working group, (geopriv@ietf.org), James Winterbottom (james.Winterbottom@commscope.com).

The XML for this schema can be found as the entirety of Section 5.5 of this document.

### 8.5. Registration Template

New registrations in the "CAtypes" registry require the following information:

CAtype: The assigned numeric CAtype. All new registrations use the value XX. [[IANA/RFC-Editor: update XX] Existing registrations use their assigned value.

Namespace URI: A unique identifier for the XML namespace used for the extension element.

Local Name: The local name of an XML element that carries the civic address element.

Description: A brief description of the semantics of the civic address element.

(Optional) Example: One or more simple examples of the element.

Contact: Contact details for the person providing the extension.

(Optional) Specification: A reference to a specification for the civic address element.

(Optional) Schema: A reference to a formal schema (XML schema, RelaxNG, or other form) that defines the extension.

Type: If Type is "A", all clients SHOULD implement this element. If Type is "B", clients MAY implement this element.

Registrations from [RFC4776] and [RFC5139] are registered with the following form:

CAtype: (The existing CAtype.)

Namespace URI: urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr

Local Name: (The contents of the PIDF column.)

Description: (The existing description for the element, including a note about the equivalent NENA field, if present.)

Contact: The IESG (iesg@ietf.org); the GEOPRIV working group (geopriv@ietf.org).

Specification: RFC4776 and RFC5139

Schema: urn:ietf:params:xml:schema:pidf:geopriv10:civicAddr

Type: A

Registration of the schema defined in this document in Section 5.5.

CAtype: The assigned numeric CAtype value is XX. [[IANA/RFC-Editor: update XX]

Namespace URI: urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr:ext

Local Name: PN, MP, STP, HNP

Description: PN: Post number that is attributed to a lamp post or utility pole.

Description: MP: Mile Post a marker indicating distance to or from a place (often a town).

Description: STP: Street Type Prefix.

Description: HNP: House Number Prefix.

Contact: The IESG (iesg@ietf.org); the GEOPRIV working group (geopriv@ietf.org).

Specification: RFCXXXX [[NOTE TO IANA/RFC-EDITOR: Please update RFC URL and replace XXXX with the RFC number for this specification.]]

Schema: urn:ietf:params:xml:schema:pidf:geopriv10:civicAddr:ext

Type: A

### 8.6. Registration Policy and Expert Guidance

The "CAtypes" registry will operate on a registration policy of "Expert Review", and optionally "Specification Required" [RFC5226] if the element being registered has a Type value of "B".

The registration rules for "Specification Required" are followed only if a registration includes a reference to a specification. Registrations can be made without a specification reference.

If the element being registered has a Type value of "A" then the registration policy is "IETF Review".

All registrations are reviewed to identify potential duplication between registered elements. Duplicated semantics are not prohibited in the registry, though it is preferred if existing elements are used. The expert review is advised to recommend the use of existing elements following the guidance in [RFC5774]. Any registration that is a duplicate or could be considered a close match for the semantics of an existing element SHOULD include a discussion of the reasons that the existing element was not reused.

## 9. Acknowledgements

Thanks to anyone who has tried to extend the civic schema and found it a little unintuitive.

## 10. References

### 10.1. 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.

[RFC4776] Schulzrinne, H., "Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information", RFC 4776, November 2006.

[RFC5139] Thomson, M. and J. Winterbottom, "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)", RFC 5139, February 2008.

[RFC5222] Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol", RFC 5222, August 2008.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[XMLNS] Hollander, D., Layman, A., Tobin, R., and T. Bray, "Namespaces in XML 1.1 (Second Edition)", World Wide Web Consortium Recommendation REC-xml-names11-20060816, August 2006, <http://www.w3.org/TR/2006/REC-xml-names11-20060816>.

## 10.2. Informative References

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.

[RFC4079] Peterson, J., "A Presence Architecture for the Distribution of GEOPRIV Location Objects", RFC 4079, July 2005.

[RFC5012] Schulzrinne, H. and R. Marshall, "Requirements for Emergency Context Resolution with Internet Technologies", RFC 5012, January 2008.

[RFC5774] Wolf, K. and A. Mayrhofer, "Considerations for Civic Addresses in the Presence Information Data Format Location Object (PIDF-LO): Guidelines and IANA Registry Definition", BCP 154, RFC 5774, March 2010.

[RFC5985] Barnes, M., "HTTP-Enabled Location Delivery (HELD)", RFC 5985, September 2010.

## Authors' Addresses

James Winterbottom
CommScope
Andrew Building (39)
Wollongong University Campus
Northfields Avenue
Wollongong, NSW  2522
AU

Phone: +61 242 212938
Email: james.winterbottom@commscope.com

Martin Thomson
CommScope
Andrew Building (39)
Wollongong University Campus
Northfields Avenue
Wollongong, NSW  2522
AU

Phone: +61 2 4221 2915
Email: martin.thomson@commscope.com

Richard Barnes
BBN Technologies
9861 Broken Land Parkway
Columbia, MD  21046
US

Phone: +1 410 290 6169
Email: rbarnes@bbn.com

Brian Rosen
NeuStar, Inc.
470 Conrad Dr
Mars, PA  16046
US

Email: br@brianrosen.net

Robins George
Huawei Technologies
Huawei Base, Bantian, Longgan District
Shenzhen, Guangdong 518129
P. R. China

Phone: +86 755 2878 8314
Email: robinsg@huawei.com

GEOPRIV H. Schulzrinne, Ed.
Internet-Draft Columbia University
Intended status: Standards Track H. Tschofenig, Ed.
Expires: April 16, 2012 Nokia Siemens Networks
J. Cuellar
Siemens
J. Polk
Cisco
J. Morris

M. Thomson
Commscope
October 14, 2011

# Geolocation Policy: A Document Format for Expressing Privacy Preferences for Location Information
draft-ietf-geopriv-policy-25.txt

## Abstract

This document defines an authorization policy language for controlling access to location information. It extends the Common Policy authorization framework to provide location-specific access control. More specifically, this document defines condition elements specific to location information in order to restrict access based on the current location of the Target.

Furthermore, this document defines two algorithms for reducing the granularity of returned location information. The first algorithm is defined for usage with civic location information while the other one applies to geodetic location information. Both algorithms come with limitations, i.e. they provide location obfuscation under certain conditions and may therefore not be appropriate for all application domains.

## Status of this Memo

Table of Contents

## 1. Introduction

Location information needs to be protected against unauthorized access to preserve the privacy of humans. In RFC 3693 [RFC3693], a protocol-independent model for access to geographic information is defined. The model includes a Location Generator (LG) that determines location information, a Location Server (LS) that authorizes access to location information, a Location Recipient (LR) that requests and receives location information, and a Rule Maker (RM) that writes authorization policies. An authorization policy is a set of rules that regulates an entity's activities with respect to privacy-sensitive information, such as location information.

The data object containing location information in the context of this document is referred to as a Location Object (LO). The basic rule set defined in the Presence Information Data Format Location Object (PIDF-LO) [RFC4119] can restrict how long the Location Recipient is allowed to retain the information, and it can prohibit further distribution. It also contains a reference to an enhanced rule set and a human readable privacy policy. The basic rule set, however, does not allow to control access to location information based on specific Location Recipients. This document describes an enhanced rule set that provides richer constraints on the distribution of LOs.

The rule set allows the entity that uses the rules defined in this document to restrict the retention and to enforce access restrictions on location data, including prohibiting any dissemination to particular individuals, during particular times or when the Target is located in a specific region. The RM can also stipulate that only certain parts of the Location Object are to be distributed to recipients or that the resolution of parts of the Location Object is reduced.

The typical sequence of operations is as follows. A Location Server receives a query for location information for a particular Target. The requestor's identity will likely be revealed as part of this request for location information. The authenticated identity of the Location Recipient, together with other information provided with the request or generally available to the server, is then used for searching through the rule set. If more than one rule matches the condition element, then the combined permission is evaluated according to the description in Section 10 of [RFC4745]. The result of the rule evalation is applied to the location information, yielding a possibly modified Location Object that is delivered to the Location Recipient.

This document does not describe the protocol used to convey location information from the Location Server to the Location Recipient.

This document extends the Common Policy framework defined in [RFC4745]. That document provides an abstract framework for expressing authorization rules. As specified there, each such rule consists of conditions, actions and transformations. Conditions determine under which circumstances the entity executing the rules, for example a Location Server, is permitted to apply actions and transformations. Transformations regulate in a location information context how a Location Server modifies the information elements that are returned to the requestor, for example, by reducing the granularity of returned location information.

This document defines two algorithms for reducing the granularity of returned location information. The first algorithm is defined for usage with civic location information (see Section 6.5.1) while the other one applies to geodetic location information (see Section 6.5.2). Both algorithms come with limitations, i.e. they provide location obfuscation under certain conditions and may therefore not be appropriate for all application domains. These limitations are documented within the security consideration section (see Section 13). It is worth pointing out that the geodetic transformation algorithm Section 6.5.2 deals with privacy risks related to targets that are stationary, as well as to moving targets. However, with respect to movement there are restriction as to what information can be hidden from an adversary. To cover applications that have more sophisticated privacy requirements additional algorithms may need to be defined. This document forsees extensions in the form of new algorithms and therefore defines a registy (see Section 11.3).

The XML schema defined in Section 9 extends the Common Policy schema by introducing new child elements to the condition and transformation elements. This document does not define child elements for the action part of a rule.

## 2. Terminology

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 RFC 2119 [RFC2119].

This document reuses the terminology of RFC 3693 [RFC3693], such as Location Server (LS), Location Recipient (LR), Rule Maker (RM), Target, Location Generator (LG) and Location Object (LO). This document uses the following terminology:

Presentity or Target:

RFC 3693 [RFC3693] uses the term Target to identify the object or person of which location information is required. The presence model described in RFC 2778 [RFC2778] uses the term presentity to describe the entity that provides presence information to a presence service. A Presentity in a presence system is a Target in a location information system.

Watcher or Location Recipient:

The receiver of location information is the Location Recipient (LR) in the terminology of RFC 3693 [RFC3693]. A watcher in a presence system, i.e., an entity that requests presence information about a presentity, is a Location Recipient in a location information system.

Authorization policy:

An authorization policy is given by a rule set. A rule set contains an unordered list of (policy) rules. Each rule has a condition, an action and a transformation component.

Permission:

The term permission refers to the action and transformation components of a rule.

In this document we use the term Location Servers as the entities that evaluate the geolocation authorization policies. The geolocation privacy architecture is, as motivated in RFC 4079 [RFC4079], aligned with the presence architecture and a Presence Server is therefore an entity that distributes location information along with other presence-specific XML data elements.

# 3. Generic Processing

## 3.1. Structure of Geolocation Authorization Documents

A geolocation authorization document is an XML document, formatted according to the schema defined in [RFC4745]. Geolocation authorization documents inherit the MIME type of common policy documents, application/auth-policy+xml. As described in [RFC4745], this document is composed of rules which contain three parts - conditions, actions, and transformations. Each action or transformation, which is also called a permission, has the property of being a positive grant of information to the Location Recipient. As a result, there is a well-defined mechanism for combining actions and transformations obtained from several sources. This mechanism is privacy safe, since the lack of any action or transformation can only result in less information being presented to a Location Recipient.

## 3.2. Rule Transport

There are two ways how the authorization rules described in this document may be conveyed between different parties:

- RFC 4119 [RFC4119] allows enhanced authorization policies to be referenced via a Uniform Resource Locator (URL) in the 'ruleset-reference' element. The ruleset-reference' element is part of the basic rules that always travel with the Location Object.

- Authorization policies might, for example, also be stored at a Location Server / Presence Server. The Rule Maker therefore needs to use a protocol to create, modify and delete the authorization policies defined in this document. Such a protocol is available with the Extensible Markup Language (XML) Configuration Access Protocol (XCAP) [RFC4825].

## 4. Location-specific Conditions

This section describes the location-specific conditions of a rule. The <conditions> element contains zero, one or an unbounded number of <location-condition> child element(s). Providing more than one <location-condition> element may not be useful since all child elements of the <conditions> element must evaluate to TRUE in order for the <conditions> element to be TRUE. The <location-condition> element MUST contain at least one <location> child element. The <location-condition> element evaluates to TRUE if any of its child elements is TRUE, i.e., a logical OR.

The <location> element has three attributes, namely 'profile', 'xml:lang' and 'label'. The 'profile' attribute allows to indicate the location profile that is included as child elements in the <location> element and each profile needs to describe under what conditions each <location> element evaluates to TRUE. This document defines two location profiles, one civic and one geodetic location profile, see Section 4.1 and Section 4.2. The 'label' attribute allows a human readable description to be added to each <location> element. The 'xml:lang' attribute contains a language tag providing further information for rendering of the content of the 'label' attribute.

The <location-condition> and the <location> elements provide extension points. If an extension is not understood by the entity evaluating the rules then this rule evaluates to FALSE.

### 4.1. Geodetic Location Condition Profile

The geodetic location profile is identified by the token 'geodetic-condition'. Rule Makers use this profile by placing a GML [GML] <Circle> element within the <location> element (as described in Section 5.2.3 of [RFC5491]).

The <location> element containing the information for the geodetic location profile evaluates to TRUE if the current location of the Target is within the described location. Note that the Target's actual location might be represented by any of the location shapes described in [RFC5491]. If the geodetic location of the Target is unknown then the <location> element containing the information for the geodetic location profile evaluates to FALSE.

Implementations are REQUIRED to support the following coordinate reference system based on WGS 84 [NIMA.TR8350.2-3e] based on the European Petroleum Survey Group (EPSG) Geodetic Parameter Dataset (as formalized by the Open Geospatial Consortium (OGC)):

2D: WGS 84 (latitude, longitude), as identified by the URN "urn:ogc:def:crs:EPSG::4326". This is a two dimensional CRS.

A CRS MUST be specified using the above URN notation only, implementations do not need to support user-defined CRSs.

Implementations MUST specify the CRS using the "srsName" attribute on the outermost geometry element. The CRS MUST NOT be changed for any sub-elements. The "srsDimension" attribute MUST be omitted, since the number of dimensions in these CRSs is known.

## 4.2. Civic Location Condition Profile

The civic location profile is identified by the token ’civic-condition’. Rule Makers use this profile by placing a <civicAddress> element, defined in [RFC5139], within the <location> element.

All child elements of <location> element that carry <civicAddress> elements MUST evaluate to TRUE (i.e., logical AND) in order for the <location> element to evaluate to TRUE. For each child element, the value of that element is compared to the value of the same element in the Target’s civic location. The child element evaluates to TRUE if the two values are identical based on a bit-by-bit comparison.

If the civic location of the Target is unknown, then the <location> element containing the information for the civic location profile evaluates to FALSE. This case may occur, for example, if location information has been removed by earlier transmitters of location information or if only the geodetic location is known. In general, it is RECOMMENDED behavior for a LS not to apply a translation from geodetic location to civic location (i.e., geocode the location).

## 5. Actions

This document does not define location-specific actions.

# 6. Transformations

This document defines several elements that allow Rule Makers to specify transformations that

- o reduce the accuracy of the returned location information, and
- o set the basic authorization policies carried inside the PIDF-LO.

## 6.1. Set Retransmission-Allowed

This element asks the LS to change or set the value of the <retransmission-allowed> element in the PIDF-LO. The data type of the <set-retransmission-allowed> element is a boolean.

If the value of the <set-retransmission-allowed> element is set to TRUE then the <retransmission-allowed> element in the PIDF-LO MUST be set to TRUE. If the value of the <set-retransmission-allowed> element is set to FALSE, then the <retransmission-allowed> element in the PIDF-LO MUST be set to FALSE.

If the <set-retransmission-allowed> element is absent then the value of the <retransmission-allowed> element in the PIDF-LO MUST be kept unchanged or, if the PIDF-LO is created for the first time, then the value MUST be set to FALSE.

## 6.2. Set Retention-Expiry

This transformation asks the LS to change or set the value of the <retention-expiry> element in the PIDF-LO. The data type of the <set-retention-expiry> element is an integer.

The value provided with the <set-retention-expiry> element indicates seconds and these seconds are added to the current date.

If the <set-retention-expiry> element is absent then the value of the <retention-expiry> element in the PIDF-LO is kept unchanged or, if the PIDF-LO is created for the first time, then the value MUST be set to the current date.

## 6.3. Set Note-Well

This transformation asks the LS to change or set the value of the <note-well> element in the PIDF-LO. The data type of the <set-note-well> element is a string.

The value provided with the <set-note-well> element contains a privacy statement as a human readable text string and an 'xml:lang' attribute denotes the language of the human readable text.

If the <set-note-well> element is absent, then the value of the <note-well> element in the PIDF-LO is kept unchanged or, if the PIDF-LO is created for the first time, then no content is provided for the <note-well> element.

### 6.4. Keep Ruleset Reference

This transformation allows to influence whether the <external-ruleset> element in the PIDF-LO carries the extended authorization rules defined in [RFC4745]. The data type of the <keep-rule-reference> element is Boolean.

If the value of the <keep-rule-reference> element is set to TRUE, then the <external-ruleset> element in the PIDF-LO is kept unchanged when included. If the value of the <keep-rule-reference> element is set to FALSE, then the <external-ruleset> element in the PIDF-LO MUST NOT contain a reference to an external rule set. The reference to the ruleset is removed and no rules are carried as MIME bodies (in case of CID URIs).

If the <keep-rule-reference> element is absent, then the value of the <external-ruleset> element in the PIDF-LO is kept unchanged when available or, if the PIDF-LO is created for the first time then the <external-ruleset> element MUST NOT be included.

### 6.5. Provide Location

The <provide-location> element contains child elements of a specific location profile that controls the granularity of returned location information. This form of location granularity reduction is also called ’obfuscation’ and is defined in [duckham05] as

> "the means of deliberately degrading the quality of information about an individual’s location in order to protect that individual’s location privacy."

Location obscuring presents a number of technical challenges. The algorithms provided in this document are provided as examples only. A discussion of the technical constraints on location obscuring is included in Section 13.4.

The functionality of location granularity reduction depends on the type of location provided as input. This document defines two profiles for reduction, namely:

o If the <provide-location> element has a <provide-civic> child element then civic location information is disclosed as described in Section 6.5.1, subject to availability.

o If the <provide-location> element has a <provide-geo> child element then geodetic location information is disclosed as described in Section 6.5.2, subject to availability.

The <provide-location> element MUST contain the 'profile' attribute if it contains child elements and the 'profile' attribute MUST match with the contained child elements.

If the <provide-location> element has no child elements then civic, as well as, geodetic location information is disclosed without reducing its granularity, subject to availability. In this case the profile attribute MUST NOT be included.

### 6.5.1. Civic Location Profile

This profile uses the token 'civic-transformation'. This profile allows civic location transformations to be specified by means of the <provide-civic> element that restricts the level of civic location information the LS is permitted to disclose. The symbols of these levels are: 'country', 'region', 'city', 'building', 'full'. Each level is given by a set of civic location data items such as <country> and <A1>, ..., <POM>, as defined in [RFC5139]. Each level includes all elements included by the lower levels.

The 'country' level includes only the <country> element; the 'region' level adds the <A1> element; the 'city' level adds the <A2> and <A3> elements; the 'building' level and the 'full' level add further civic location data as shown below.

```
                          full
      {<country>, <A1>, <A2>, <A3>, <A4>, <A5>, <A6>, <PRD>, <POD>,
       <STS>, <HNO>, <HNS>, <LMK>, <LOC>, <PC>, <NAM>, <FLR>,
       <BLD>,<UNIT>,<ROOM>,<PLC>, <PCN>, <POBOX>, <ADDCODE>, <SEAT>
       <RD>, <RDSEC>, <RDBR>, <RDSUBBR>, <PRM>, <POM>}
                           |
                           |
                        building
         {<country>, <A1>, <A2>, <A3>, <A4>, <A5>, <A6>, <PRD>
         <POD>, <STS>, <HNO>, <HNS>, <LMK>, <PC>,
         <RD>, <RDSEC>, <RDBR>, <RDSUBBR> <PRM>, <POM>}
                           |
                           |
                         city
                 {<country>, <A1>, <A2>, <A3>}
                           |
                           |
                         region
                    {<country>, <A1>}
                           |
                           |
                        country
                      {<country>}
                           |
                           |
                          none
                          {}
```

The default value is "none".

The schema of the <provide-civic> element is defined in Section 8.

### 6.5.2. Geodetic Location Profile

This profile uses the token 'geodetic-transformation' and refers only to the Coordinate Reference System (CRS) WGS 84 (urn:ogc:def:crs:EPSG::4326, 2D). This profile allows geodetic location transformations to be specified by means of the <provide-geo> element that may restrict the returned geodetic location information based on the value provided in the 'radius' attribute. The value of the 'radius' attribute expresses the radius in meters.

The schema of the <provide-geo> element is defined in Section 8.

The algorithm proceeds in 6 steps.The first two steps are independent of the measured position to be obscured. Those two steps should be run only once or rather seldom (for every region and desired uncertainty). The steps are:

1. Choose a geodesic projection with Cartesian coordinates and a surface you want to cover. The maximal distortion of the map may not be too much (see notes below).

2. Given uncertainty "d", choose a grid of so called "landmarks" at a distance (maximal) d of each other.

3. Given a measured location M=(m,n) in the surface, calculate its 4 closest landmarks on the grid, with coordinates: SW = (l,b), SE=(r,b), NW=(l,t), NE=(r,t). Thus l<=m<r and b<=n<t. See notes below.

4. Let x=(m-l)/(r-l) and y=(n-b)/(t-b)

   x and y are thus the local coordinates of the point M in the small grid square that contains it. 0<=x,y<1.

5. Let P = 0.2887 (=sqrt(3)/6) and q = 0.7113 (=1-p), determine which of the following 8 cases holds:

```
C1. x < p and y < p
C2. p <= x < q and y < x and y < 1-x
C3. q <= x and y < p

C4. p <= y < q and x <= y and y < 1-x
C5. p <= y < q and y < x and 1-x <= y

C6. x < p and q <= y
C7. p <= x < q and x <= y and 1-x <= y
C8. q <= x and q <= y
```

6. Depending on the case, let C (=Center) be

```
C1: SW
C2: SW or SE
C3: SE

C4: SW or NW
C5: SE or NE

C6: NW
C7: NW or NE
C8: NE
```

Return the circle with center C and radius d.

Notes:

Regarding Step 1:

The scale of a map is the ratio of a distance on (a straight line) on the map to the corresponding air distance on the ground. For maps covering larger areas, a map projection from a sphere (or ellipsoid) to the plane will introduce distortion and the scale of the map is not constant. Also, note that the real distance on the ground is taken along great circles, which may not correspond to straight lines in the map, depending on the projection used. Let us measure the (length) distortion of the map as the quotient between the maximal and the minimal scales in the map. The distortion MUST be below 1.5. (The minimum distortion is 1.0: If the region of the map is small, then the scale may be taken as a constant over the whole map).

Regarding Step3:

SW is mnemonic for south-west, b for bottom, l for left (SW=(l,b)), etc, but the directions of the geodesic projection may be arbitrary, and thus SW may be not south-west of M but it will be left and below M *on the map*.

This algorithm is applied to the known location every time it changes. Movement between the defined grid areas triggers a change in the location that is reported, which could reveal significantly more about the location than desired.

# 7. Examples

This section provides a few examples for authorization rules using the extensions defined in this document.

## 7.1. Rule Example with Civic Location Condition

This example illustrates a single rule that employs the civic location condition. It matches if the current location of the Target equal the content of the child elements of the <location> element. Requests match only if the Target is at a civic location with country set to ’Germany’, state (A1) set to ’Bavaria’, city (A3) set to ’Munich’, city division (A4) set to ’Perlach’, street name (A6) set to ’Otto-Hahn-Ring’ and house number (HNO) set to ’6’.

No actions and transformation child elements are provided in this rule example. The actions and transformation could include presence specific information when the Geolocation Policy framework is applied to the Presence Policy framework (see [RFC5025]).

```
<?xml version="1.0" encoding="UTF-8"?>
<ruleset xmlns="urn:ietf:params:xml:ns:common-policy"
  xmlns:gp="urn:ietf:params:xml:ns:geolocation-policy">

  <rule id="AA56i09">
    <conditions>
      <gp:location-condition>
        <gp:location
          profile="civic-condition"
          xml:lang="en"
          label="Siemens Neuperlach site ’Legoland’"
          xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr">
          <country>DE</country>
          <A1>Bavaria</A1>
          <A3>Munich</A3>
          <A4>Perlach</A4>
          <A6>Otto-Hahn-Ring</A6>
          <HNO>6</HNO>
        </gp:location>
      </gp:location-condition>
    </conditions>
    <actions/>
    <transformations/>
  </rule>
</ruleset>
```

## 7.2. Rule Example with Geodetic Location Condition

This example illustrates a rule that employs the geodetic location condition. The rule matches if the current location of the Target is inside the area specified by the polygon. The polygon uses the EPSG 4326 coordinate reference system. No altitude is included in this example.

```
<?xml version="1.0" encoding="UTF-8"?>
<ruleset
  xmlns="urn:ietf:params:xml:ns:common-policy"
  xmlns:gp="urn:ietf:params:xml:ns:geolocation-policy"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:gs="http://www.opengis.net/pidflo/1.0">

  <rule id="BB56A19">
    <conditions>
      <gp:location-condition>
        <gp:location
          xml:lang="en"
          label="Sydney Opera House"
          profile="geodetic-condition">
          <gs:Circle srsName="urn:ogc:def:crs:EPSG::4326">
            <gml:pos>-33.8570029378 151.2150070761</gml:pos>
            <gs:radius uom="urn:ogc:def:uom:EPSG::9001">1500
            </gs:radius>
          </gs:Circle>
        </gp:location>
      </gp:location-condition>
    </conditions>
    <transformations/>
  </rule>
</ruleset>
```

## 7.3. Rule Example with Civic and Geodetic Location Condition

This example illustrates a rule that employs a mixed civic and geodetic location condition. Depending on the available type of location information, namely civic or geodetic location information, one of the location elements may match.

```
<?xml version="1.0" encoding="UTF-8"?>
<ruleset
  xmlns="urn:ietf:params:xml:ns:common-policy"
  xmlns:gp="urn:ietf:params:xml:ns:geolocation-policy"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:gs="http://www.opengis.net/pidflo/1.0">

  <rule id="AA56i09">
    <conditions>
      <gp:location-condition>
        <gp:location profile="civic-condition"
          xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr">
          <country>DE</country>
          <A1>Bavaria</A1>
          <A3>Munich</A3>
          <A4>Perlach</A4>
          <A6>Otto-Hahn-Ring</A6>
          <HNO>6</HNO>
        </gp:location>
        <gp:location profile="geodetic-condition">
          <gs:Circle srsName="urn:ogc:def:crs:EPSG::4326">
             <gml:pos>-34.410649 150.87651</gml:pos>
             <gs:radius uom="urn:ogc:def:uom:EPSG::9001">1500
             </gs:radius>
          </gs:Circle>
        </gp:location>
      </gp:location-condition>
    </conditions>
    <actions/>
    <transformations/>
  </rule>
</ruleset>
```

## 7.4. Rule Example with Location-based Transformations

This example shows the transformations specified in this document. The <provide-civic> element indicates that the available civic location information is reduced to building level granularity. If geodetic location information is requested then a granularity reduction is provided as well.

```
<?xml version="1.0" encoding="UTF-8"?>
<ruleset xmlns="urn:ietf:params:xml:ns:common-policy"
  xmlns:gp="urn:ietf:params:xml:ns:geolocation-policy"
  xmlns:lp="urn:ietf:params:xml:ns:basic-location-profiles">

  <rule id="AA56i09">
    <conditions/>
    <actions/>
    <transformations>
      <gp:set-retransmission-allowed>false
      </gp:set-retransmission-allowed>
      <gp:set-retention-expiry>86400</gp:set-retention-expiry>
      <gp:set-note-well xml:lang="en">My privacy policy goes in here.
      </gp:set-note-well>
      <gp:keep-rule-reference>false
      </gp:keep-rule-reference>

      <gp:provide-location
        profile="civic-transformation">
        <lp:provide-civic>building</lp:provide-civic>
      </gp:provide-location>

      <gp:provide-location
        profile="geodetic-transformation">
        <lp:provide-geo radius="500"/>
      </gp:provide-location>

    </transformations>
  </rule>
</ruleset>
```

The following rule describes the short-hand notation for making the current location of the Target available to Location Recipients without granularity reduction.

```
<?xml version="1.0" encoding="UTF-8"?>
<ruleset xmlns="urn:ietf:params:xml:ns:common-policy"
    xmlns:gp="urn:ietf:params:xml:ns:geolocation-policy">

    <rule id="AA56ia9">
        <conditions/>
        <actions/>
        <transformations>
            <gp:provide-location/>
        </transformations>
    </rule>
</ruleset>
```

## 7.5. Location Obfuscation Example

Suppose you want a to obscure positions in the continental USA.

Step 1:

First you choose a geodesic projection. If you are measuring location as latitude and longitude, a natural choice is to take a rectangular projection. One latitudinal degree corresponds approximately to 110.6 kilometers, while a good approximation of a longitudinal degree at latitude phi is (pi/180)*M*cos(phi), where pi is approximately 3.1415, and M is the Earth's average meridional radius, approximately 6,367.5 km. For instance, one longitudinal degree at 30 degrees (say, New Orleans) is 96.39 km, while the formula given offers an estimation of 96.24, which is good for our purposes.

We will set up a grid not only for the continental US, but for the whole earth between latitudes 25 and 50 degrees, and thus will cover also the Mediterranean, South Europe, Japan and the north of China. As will be seen below, the grid distortion (for not too large grids in this region) is approx cos(25)/cos(50), which is 1.4099.

As origin of our grid, we choose the point at latitude 25 degrees and longitude 0 (Greenwich). The latitude 25 degrees is chosen to be just south of Florida and thus south of the continental US. (On the south hemisphere the origin should be north of the region to be covered; if the region crosses the Equator, the origin should be on the Equator. In his way it is guaranteed that the latitudinal degree has largest distance at the latitude of the origin).

At 25 degrees one degree in east-west direction corresponds approx to (pi/180)*M*cos(25) = 100.72 km.

The same procedure, basically, produces grids for

* 45 degrees south to 45 degrees north Tropics and subtropics
* 25 to 50 degrees (both north or south) Continental US
* 35 to 55 degrees (both north or south) South and Central Europe
* 45 to 60 degrees (both north or south) Central and North Europe
* 55 to 65 degrees (both north or south) Scandinavia

* 60 to 70 degrees (both north or south)

Since we do not want to often change grid system (this would leak more information about obscured locations when they are repeatedly visited), the algorithm should prefer to use the grids discussed above, with origin at the Greenwich meridian and at latitudes o=0, o=25, o=35, o=45, 0=55, and o=60 degrees (north) or at latitudes o=-25, o=-35, o=-45, 0=-55, and o=-60 degrees (the minus to indicate "south").

Our choice for the continental USA is o=25.

For locations close to the poles, a different projection should be used (not discussed here).

Step 2:

To construct the grid points, we start with our chosen origin and place the along the main axes (NS and EW) grid points at a distance d of each other.

We will now construct a grid for a desired uncertainty of d = 100km. At our origin, 100 km correspond roughly to d1 = 100/100.72 = 0.993 degrees on east-west direction and to d2 = 100/110.6 = 0.904 degrees in north-south direction.

The (i,j)-point in the grid (i and j are integers) has longitude d1*i and latitude 25+d2*j, measured in degrees. More generally, if the grid has origin at coordinates (0,o), measured in degrees, the (i,j)-point in the grid has coordinates (longitude = d1*i, latitude = o+d2*j). The grid has almost no distorsion at the latitude of the origin, but it has as we go further away from it.

The distance between two points in the grid at 25 degrees latitude is indeed approx 100 km, but just above the Canadian border, on the 50th degree, it is 0.993*(pi/180)*M*cos(50) = 70.92km. Thus, the grid distortion is 100/70.92 = 1.41, which is acceptable (<1.5). (On north-south direction the grid has roughly no distortion, the vertical distance between two neighboring grid points is approximately 100 km).

Step 3:

Now suppose you measure a position at M, with longitude -105 (the minus sign is used to denote 105 degrees *west*; without minus, the point is in China, 105 degrees east) and latitude 40 degrees (just north of Denver, CO). The point M is 105 degrees west and 15 degrees north of our origin (which has longitude 0 and latitude 25).

Let "floor" be the function that returns the largest integer smaller or equal to a floating point number. To calculate SW, the closest point of the grid on the south-west of M=(m,n), we calculate

i= floor(m/d1) = floor(-105/0.993) = -106

j= floor(n-o/d2) = floor(15/0.904) = 16

Those are the indexes of SW on the grid. The coordinates of SW are then: (d1*i, 25+d2*j) = (-105.242, 39.467).

Thus:

l=d1*floor(m/d1) = -105.243

r=l+d1 = -105.243+0.993 = -104.250

b=o+d2*floor(n-o/d2) = 39.467

t=b+d2 = 39.467+0.904 = 40.371

These are the formulas for l,r,b, and t in the general case of Cartesian projections based on latitude and longitude.

Step 4:

Calculate x and y, the local coordinates of the point M in the small grid square that contains it. This is easy:

x=(m-l)/(r-l) = [-105 -(-105.243)]/0.993 = 0.245

y=(n-b)/(t-b) = [40 - 39.467]/0.904 = 0.590

Step 5:

First compare x with p (0.2887) and (0.7113). x is smaller than p. Therefore, only cases 1,4 or 6 could hold.

Also compare y with p (0.2887) and (0.7113). y is between them: p <= y < q. Thus, we must be in case 4. To check, compare y (0.59) with x (0.245) and 1-x. y is larger than x and smaller than 1-x.

We are in case C4 (p <= y < q and x <= y and y < 1-x).

Step 6:

Now we choose either SW or NW as the center of the circle.

The obscured location is the Circle with radius 100 km and center in SW (coordinates: -105.243, 39.467), or NW (coordinates: -105.243, 40.371).

# 8. XML Schema for Basic Location Profiles

This section defines the location profiles used as child elements of the transformation element.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
    targetNamespace="urn:ietf:params:xml:ns:basic-location-profiles"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified">

    <!-- profile="civic-transformation" -->

    <xs:element name="provide-civic" default="none">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="full"/>
                <xs:enumeration value="building"/>
                <xs:enumeration value="city"/>
                <xs:enumeration value="region"/>
                <xs:enumeration value="country"/>
                <xs:enumeration value="none"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>

    <!-- profile="geodetic-transformation" -->

    <xs:element name="provide-geo">
        <xs:complexType>
            <xs:attribute name="radius" type="xs:integer"/>
        </xs:complexType>
    </xs:element>

</xs:schema>
```

## 9. XML Schema for Geolocation Policy

This section presents the XML schema that defines the Geolocation Policy schema described in this document. The Geolocation Policy schema extends the Common Policy schema (see [RFC4745]).

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="urn:ietf:params:xml:ns:geolocation-policy"
  xmlns:gp="urn:ietf:params:xml:ns:geolocation-policy"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <!-- Import Common Policy-->
  <xs:import namespace="urn:ietf:params:xml:ns:common-policy"/>

  <!-- This import brings in the XML language attribute xml:lang-->
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>

  <!-- Geopriv Conditions -->

  <xs:element name="location-condition"
    type="gp:locationconditionType"/>

  <xs:complexType name="locationconditionType">
    <xs:complexContent>
      <xs:restriction base="xs:anyType">
        <xs:choice minOccurs="1" maxOccurs="unbounded">
          <xs:element name="location" type="gp:locationType"
            minOccurs="1" maxOccurs="unbounded"/>
          <xs:any namespace="##other" processContents="lax"
            minOccurs="0" maxOccurs="unbounded"/>
        </xs:choice>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="locationType">
    <xs:complexContent>
      <xs:restriction base="xs:anyType">
        <xs:choice minOccurs="1" maxOccurs="unbounded">
          <xs:any namespace="##other" processContents="lax"
            minOccurs="0" maxOccurs="unbounded"/>
        </xs:choice>
        <xs:attribute name="profile" type="xs:string"/>
```

```
          <xs:attribute name="label" type="xs:string"/>
          <xs:attribute ref="xml:lang" />
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <!-- Geopriv transformations -->
    <xs:element name="set-retransmission-allowed"
      type="xs:boolean" default="false"/>
    <xs:element name="set-retention-expiry"
      type="xs:integer" default="0"/>
    <xs:element name="set-note-well"
      type="gp:notewellType"/>
    <xs:element name="keep-rule-reference"
      type="xs:boolean" default="false"/>

    <xs:element name="provide-location"
      type="gp:providelocationType"/>

    <xs:complexType name="notewellType">
      <xs:simpleContent>
        <xs:extension base="xs:string">
          <xs:attribute ref="xml:lang" />
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>

    <xs:complexType name="providelocationType">
      <xs:complexContent>
        <xs:restriction base="xs:anyType">
          <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:any namespace="##other" processContents="lax"
              minOccurs="0" maxOccurs="unbounded"/>
          </xs:choice>
          <xs:attribute name="profile" type="xs:string" />
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:schema>
```

## 10. XCAP Usage

The following section defines the details necessary for clients to manipulate geolocation privacy documents from a server using XCAP. If used as part of a presence system, it uses the same AUID as those rules. See [RFC5025] for a description of the XCAP usage in context with presence authorization rules.

### 10.1. Application Unique ID

XCAP requires application usages to define a unique application usage ID (AUID) in either the IETF tree or a vendor tree. This specification defines the "geolocation-policy" AUID within the IETF tree, via the IANA registration in Section 11.

### 10.2. XML Schema

XCAP requires application usages to define a schema for their documents. The schema for geolocation authorization documents is described in Section 9.

### 10.3. Default Namespace

XCAP requires application usages to define the default namespace for their documents. The default namespace is urn:ietf:params:xml:ns:geolocation-policy.

### 10.4. MIME Type

XCAP requires application usages to defined the MIME type for documents they carry. Geolocation privacy authorization documents inherit the MIME type of common policy documents, application/auth-policy+xml.

### 10.5. Validation Constraints

This specification does not define additional constraints.

### 10.6. Data Semantics

This document discusses the semantics of a geolocation privacy authorization.

### 10.7. Naming Conventions

When a Location Server receives a request to access location information of some user foo, it will look for all documents within http://[xcaproot]/geolocation-policy/users/foo, and use all documents found beneath that point to guide authorization policy.

## 10.8. Resource Interdependencies

This application usage does not define additional resource interdependencies.

## 10.9. Authorization Policies

This application usage does not modify the default XCAP authorization policy, which is that only a user can read, write or modify his/her own documents. A server can allow privileged users to modify documents that they do not own, but the establishment and indication of such policies is outside the scope of this document.

## 11. IANA Considerations

There are several IANA considerations associated with this specification.

### 11.1. Geolocation Policy XML Schema Registration

URI: urn:ietf:params:xml:schema:geolocation-policy

Registrant Contact: IETF Geopriv Working Group, Hannes Tschofenig (hannes.tschofenig@nsn.com).

XML: The XML schema to be registered is contained in Section 9. Its first line is

```
<?xml version="1.0" encoding="UTF-8"?>
```

and its last line is

```
</xs:schema>
```

### 11.2. Geolocation Policy Namespace Registration

URI: urn:ietf:params:xml:ns:geolocation-policy

Registrant Contact: IETF Geopriv Working Group, Hannes Tschofenig (hannes.tschofenig@nsn.com).

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
  "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="content-type"
        content="text/html;charset=iso-8859-1"/>
  <title>Geolocation Policy Namespace</title>
</head>
<body>
  <h1>Namespace for Geolocation Authorization Policies</h1>
  <h2>urn:ietf:params:xml:schema:geolocation-policy</h2>
<p>See <a href="[URL of published RFC]">RFCXXXX
    [NOTE TO IANA/RFC-EDITOR:
     Please replace XXXX with the RFC number of this
    specification.]</a>.</p>
</body>
</html>
END
```

## 11.3. Geolocation Policy Location Profile Registry

This document seeks to create a registry of location profile names for the Geolocation Policy framework. Profile names are XML tokens. This registry will operate in accordance with RFC 2434 [RFC2434], Standards Action.

This document defines the following profile names:

geodetic-condition: Defined in Section 4.1.

civic-condition: Defined in Section 4.2.

geodetic-transformation: Defined in Section 6.5.2.

civic-transformation: Defined in Section 6.5.1.

## 11.4. Basic Location Profile XML Schema Registration

URI: urn:ietf:params:xml:schema:basic-location-profiles

Registrant Contact: IETF Geopriv Working Group, Hannes Tschofenig (hannes.tschofenig@nsn.com).

XML: The XML schema to be registered is contained in Section 8. Its first line is

```
<?xml version="1.0" encoding="UTF-8"?>
```

and its last line is

```
</xs:schema>
```

## 11.5. Basic Location Profile Namespace Registration

URI: urn:ietf:params:xml:ns:basic-location-profiles

Registrant Contact: IETF Geopriv Working Group, Hannes Tschofenig (hannes.tschofenig@nsn.com).

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
  "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="content-type"
        content="text/html;charset=iso-8859-1"/>
  <title>Basic Location Profile Namespace</title>
</head>
<body>
  <h1>Namespace for Basic Location Profile</h1>
  <h2>urn:ietf:params:xml:schema:basic-location-profiles</h2>
<p>See <a href="[URL of published RFC]">RFCXXXX
    [NOTE TO IANA/RFC-EDITOR:
     Please replace XXXX with the RFC number of this
    specification.]</a>.</p>
</body>
</html>
END
```

## 11.6. XCAP Application Usage ID

This section registers an XCAP Application Usage ID (AUID) according to the IANA procedures defined in [RFC4825].

Name of the AUID: geolocation-policy

Description: Geolocation privacy rules are documents that describe the permissions that a Target has granted to Location Recipients that access information about his/her geographic location.

## 12. Internationalization Considerations

The policies described in this document are mostly meant for machine-to-machine communications; as such, many of its elements are tokens not meant for direct human consumption. If these tokens are presented to the end user, some localization may need to occur. The policies are, however, supposed to be created with the help of humans and some of the elements and attributes are subject to internationalization considerations. The content of the <label> element is meant to be provided by a human (the Rule Maker) and also displayed to a human. Furthermore, the location condition element (using the civic location profile, see Section 4.2) and the <set-note-well> element (see Section 6.3) may contain non-US-ASCII letters.

The geolocation polices utilize XML and all XML processors are required to understand UTF-8 and UTF-16 encodings, and therefore all entities processing these policies MUST understand UTF-8 and UTF-16 encoded XML. Additionally, geolocation policy aware entities MUST NOT encode XML with encodings other than UTF-8 or UTF-16.

# 13. Security Considerations

## 13.1. Introduction

This document aims to allow users to prevent unauthorized access to location information and to restrict access to information dependent on geolocation (via location based conditions). This is accomplished through the usage of authorization policies. This work builds on a series of other documents: Security requirements are described in [RFC3693] and a discussion of generic security threats is available with [RFC3694]. Aspects of combining permissions in cases of multiple occurrence are treated in [RFC4745]).

In addition to the authorization policies mechanisms for obfuscating location information are described. A theoretical treatment of location obfuscation is provided in [duckham05] and in [ifip07]. [duckham05] provides the foundation and [ifip07] illustrates three different types of location obfuscation by enlarging the radius, by shifting the center, and by reducing the radius. This algorithm for geodetic location information obfuscation makes use of these techniques.

The requirements of location information with respect to privacy protection vary. While the two obfuscation algorithm in this document provide a basis for protection they have limitations. There are certainly applications that require a different level of protection and for this purpose the ability to define and use additional algorithms has been envisioned. New algorithms can be specified via the available extension mechanism.

## 13.2. Obfuscation

Whenever location information is returned to a location recipient then it contains the location of the Target. This is also true when location is obfuscated, i.e. the location server does not lie about the Target's location but instead hides it within a larger location shape. However, even without the Target's movement there is danger to reveal information over time. While the target's location is hidden within the privacy region the size of that returned region matters as well as the precise location of the Target within that region. Returning location shapes that are randomly computed will over time real more and more informationa bout the Target.

Consider the drawing in Figure 1, which shows three ellipses, a dotted area in the middle, and the Target's true location marked as 'x'. The ellipses illustrate the location shapes as received by a potential location recipient over time for requests of a target's location information. Collecting information about the returned location information over time allows the location recipient to narrow the potential location of the target down to the dotted area in the center of the graph.

For this purpose the algorithm described in Section 6.5.2 uses a grid that ensures to report the same location information whenever the target remains in the same geographical area.

```
                ,-----.
        ,----,-'.      `-.
     ,-'    /    `-.      \
   ,'      / _...._ `.     \
  /       ,-'......`._\     :
 ;       /|...........\:    |
 |      / :.....x......+    ;
 :     |   \...........;|  /
  \    |    \........./ | /
   `.  \     `-.....,' ,''
     '-.\       `-----'|
        ''.-----'    ,'
           `._    _,'
              ''''
```

Figure 1: Obfuscation: A Static Target

When the Target is moving then the location transformations reveal information when switching from one privacy region to another one. For example, when a transformation indicates that civic location is provided at a 'building' level of granularity. Consequently, floor levels, room numbers, etc. would be hidden. However, when the Target moves from one building to the next one then the movement would still be recognizable as the disclosed location information would be reflected by the new civic location information indicating the new building. With additional knowledge about building entrances and floor plans it would be possible to learn additional amount of information.

The algorithm presented in Section 6.5.2 has some measures against leaking information when moving, switching from one privacy region to another one, and also when the user is visiting regularly the same location. The first issue is the following: if you provide a different location information (privacy region) only when the previous one is not valid anymore, you will be disclosing the moment where you are in a certain border (namely: leaving the previous privacy region). The second issue is the following: Suppose a user goes home every night. If the reported obfuscated locations are all randomly different, an analysis will probably soon reveal the home location with high precision. Of course, the combination of an obscured location with public geographic information (highways, lakes, mountains, cities, etc) may render a much precise location information than desired. But even without it, just observing movements, once or in a regular repetitive way, any obscuring algorithm will leak information about velocities or positions. Suppose a user wants to disclose location information with a radius of r. The privacy region, a circle with that radius, has an area of A = pi * r^2. An opponent, observing the movements, will deduce that the information that the target is, was, or regularly visits, a region of size A1, smaller than A. The quotient of the sizes A1/A should be, even in the worst case, larger than a fixed known number, in order that the user knows what is the maximal information leakage he has. The choices of 6.5.2 are such that this maximum leakage can be established: by any statistical procedures, without using geographic information (highways, etc. as discussed above), the quotient A1/A is larger than 0.13 (= 1/(5*1.5) ). Thus, for instance, when choosing a provided location of size 1000 km^2, he will be leaking, in worst case, the location within a region of size 130 km^2.

## 13.3. Usability

There is the risk that end users are specifying their location-based policies in such a way that very small changes in location yields a significantly different level of information disclosure. For example, a user might want to set authorization policies differently when they are in a specific geographical area (e.g., at home, in the office). Location might be the only factor in the policy that triggers a very different action and transformation to be executed. The accuracy of location information is not always sufficient to unequivocally determine whether a location is within a specific boundary [I-D.thomson-geopriv-uncertainty]. In some situations uncertainty in location information could produce unexpected results for end users. Providing adequate user feedback about potential errors arising from these limitation can help prevent unintentional information leakage.

Users might create policies that are non-sensical. To avoid such cases the software used to create the authorization policies should perform consistency checks and when authorization policies are uploaded to the policy servers then further checks are performed. When XCAP is used to upload authorization policies then built-in features of XCAP can be utilized to convey error messages back to the user about an error condition. Section 8.2.5 of [RFC4825] indicates that some degree of application specific checking is provided when authorization policies are added, modified or deleted. The XCAP protocol may return a 409 response with a response that may contain a detailed conflict report containing the <constraint-failure> element. A human readable description of the problem can be indicated in the 'phrase' attribute of that element.

### 13.4. Location Obscuring Limitations

Location obscuring attempts to remove information about the location of a Target. The effectiveness of location obscuring is determined by how much uncertainty a Location Recipient has about the location of the Target. A location obscuring algorithm is effective if the Location Recipient cannot recover a location with better uncertainty than the obscuring algorithm was instructed to add.

Effective location obscuring is difficult. The amount of information that can be recovered by a determined and resourceful Location Recipient can be considerably more than is immediately apparent. A concise summary of the challenges is included in [duckham10].

A Location Recipient in possession of external information about the Target or geographical area that is reported can make assumptions or guesses aided by that information to recover more accurate location information. This is true even when a single location is reported, but it is especially true when multiple locations are reported for the same Target over time.

Furthermore, a Location Recipient that attempts to recover past locations for a Target can use later reported locations to further refine any recovered location. A location obscuring algorithm typically does not have any information about the future location of the Target.

The degree to which location information can be effectively degraded by an obscuring algorithm depends on the information that is used by the obscuring algorithm. If the information available to the obscuring algorithm is both more extensive and more effectively employed than the information available to the Location Recipient, then location obscuring might be effective.

Obscured locations can still serve a purpose where a Location Recipient is willing to respect privacy. A privacy-respecting Location Recipient can choose to interpret the existence of uncertainty as a request from a Rule Maker to not recover location.

Location obscuring is unlikely to be effective against a more determined or resourceful adversary. Withholding location information entirely is perhaps the most effective method of ensuring that it is not recovered. However, a caution: omitted data also conveys some information.

# 14. References

## 14.1. Normative References

[GML] OpenGIS, "OpenGIS Geography Markup Language (GML) Implementation Specification, Version 3.00, OGC 02 023r4", http://www.opengeospatial.org/docs/02-023r4.pdf, January 2003.

[NIMA.TR8350.2-3e] OpenGIS, "US National Imagery and Mapping Agency, "Department of Defense (DoD) World Geodetic System 1984 (WGS 84), Third Edition, NIMA TR8350.2", , January 2000.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.

[RFC4745] Schulzrinne, H., Tschofenig, H., Morris, J., Cuellar, J., Polk, J., and J. Rosenberg, "Common Policy: A Document Format for Expressing Privacy Preferences", RFC 4745, February 2007.

[RFC5139] Thomson, M. and J. Winterbottom, "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)", RFC 5139, February 2008.

[RFC5491] Winterbottom, J., Thomson, M., and H. Tschofenig, "GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations", RFC 5491, March 2009.

## 14.2. Informative References

[I-D.thomson-geopriv-geo-shape] Thomson, M., "Geodetic Shapes for the Representation of Uncertainty in PIDF-LO", draft-thomson-geopriv-geo-shape-03 (work in progress), December 2006.

[I-D.thomson-geopriv-uncertainty] Thomson, M. and J. Winterbottom, "Representation of Uncertainty and Confidence in PIDF-LO", draft-thomson-geopriv-uncertainty-06 (work in progress), March 2011.

[RFC2434] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[RFC2778] Day, M., Rosenberg, J., and H. Sugano, "A Model for Presence and Instant Messaging", RFC 2778, February 2000.

[RFC3693] Cuellar, J., Morris, J., Mulligan, D., Peterson, J., and J. Polk, "Geopriv Requirements", RFC 3693, February 2004.

[RFC3694] Danley, M., Mulligan, D., Morris, J., and J. Peterson, "Threat Analysis of the Geopriv Protocol", RFC 3694, February 2004.

[RFC4079] Peterson, J., "A Presence Architecture for the Distribution of GEOPRIV Location Objects", RFC 4079, July 2005.

[RFC4119] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, December 2005.

[RFC4825] Rosenberg, J., "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)", RFC 4825, May 2007.

[RFC5025] Rosenberg, J., "Presence Authorization Rules", RFC 5025, December 2007.

[duckham05] Duckham, M. and L. Kulik, "A formal model of obfuscation and negotiation for location privacy. In Proc. of the 3rd International Conference PERVASIVE 2005,Munich, Germany", May 2005.

[duckham10] Duckham, M., "Moving forward: Location privacy and location awareness. In Proc. 3rd ACM SIGSPATIAL GIS Workshop on Security and Privacy in GIS and LBS (SPRINGL), ACM.", Nov 2010.

[ifip07] Ardagna, C., Cremonini, M., Damiani, E., De Capitani di Vimercati, S., and S. Samarati, "Location-privacy protection through obfuscation-based techniques, in: Proceedings of the 21st Annual IFIP WG 11.3 Working Conference on Data and Applications Security, Redondo Beach, CA, USA", July 2007.

## Appendix A. Acknowledgments

This document is informed by the discussions within the IETF GEOPRIV working group, including discussions at the GEOPRIV interim meeting in Washington, D.C., in 2003.

We particularly want to thank Allison Mankin <mankin@psg.com>, Randall Gellens <rg+ietf@qualcomm.com>, Andrew Newton <anewton@ecotroph.net>, Ted Hardie <hardie@qualcomm.com>, Jon Peterson <jon.peterson@neustar.biz> for their help in improving the quality of this document.

We would like to thank Christian Guenther for his help with an earlier version of this document. Furthermore, we would like to thank Johnny Vrancken for his document reviews in September 2006, December 2006 and January 2007. James Winterbottom provided a detailed review in November 2006. Richard Barnes gave a detailed review in February 2008.

This document uses text from [I-D.thomson-geopriv-geo-shape]. Therefore, we would like to thank Martin Thomson for his work in [I-D.thomson-geopriv-geo-shape]. We would also like to thank Martin Thomson, Matt Lepinski and Richard Barnes for their comments regarding the geodetic location transformation procedure. Richard provided us with a detailed text proposal.

Robert Sparks, Martin Thomson, and Warren Kumari deserve thanks for their input on the location obfuscation discussion. Robert implemented various versions of the algorithm in the graphical language "Processing" and thereby helped us tremendously to understand problems with the previously illustrated algorithm.

We would like to thank Dan Romascanu, Yoshiko Chong and Jari Urpalainen for their last call comments.

Finally, we would like to thank the following individuals for their feedback as part of the IESG, GenArt, and SecDir review: Jari Arkko, Eric Gray, Russ Housley, Carl Reed, Martin Thomson, Lisa Dusseault, Chris Newman, Jon Peterson, Sam Hartman, Cullen Jennings, Tim Polk, and Brian Rosen.

## Appendix B. Pseudo-Code

This section provides an informal description for the algorithm described in Section 6.5.2 in form of pseudo-code.

```
Constants

  P = sqrt(3)/6  //  approx 0.2887
  q = 1 - p      //  approx 0.7113

Parameters

  prob:  real  // prob is a parameter in the range
        //  0.5 <= prob <=1
        // recommended is a value for prob between 0.7 and 0.9
        // the default of prob is 0.8

Inputs

  M = (m,n) : real * real
        // M is a pair of reals: m and n
        // m is the longitude and n the latitude,
        // respectively, of the measured location
        // The values are given as real numbers, in the
        // range: -180 < m <= 180; -90 < n < 90
        // minus values for longitude m correspond to "West"
        // minus values for latitude n correspond to "South"

  radius : integer // the 'radius' or uncertainity,
        // measured in meters

  prev-M = (prev-m1, prev-n1): real * real
        // the *previously* provided location, if available
        // prev-m1 is the longitude and
        // prev-n1 the latitude, respectively

  o : real

  // this is the reference latitude for the geodesic projection
  // The value of 'o' is chosen according to the table below.
  // The area you want to project MUST be included in
  // between a minimal latitude and a maximal latitude
  // given by the two first columns of the table.
  // (Otherwise the transformation is not available).

  //    +------+------+--------------------------+-------+
  //    | min  | max  |                          |       |
```

```
     //     | lat  | lat  |       Examples           |  o    |
     //     +------+------+--------------------------+-------+
     //     |      |      | Tropics and subtropics   |       |
     //     | -45  |  45  | Africa                   |  0    |
     //     |      |      | Australia                |       |
     //     +------+------+--------------------------+-------+
     //     |      |      | Continental US           |       |
     //     |  25  |  50  | Mediterranean            |   25  |
     //     |      |      | most of China            |       |
     //     +------+------+--------------------------+-------+
     //     |      |      |                          |       |
     //     |  35  |  55  | South and Central        |   35  |
     //     |      |      |      Europe              |       |
     //     +------+------+--------------------------+-------+
     //     |      |      |                          |       |
     //     |  45  |  60  | Central and North        |   45  |
     //     |      |      |       Europe             |       |
     //     +------+------+--------------------------+-------+
     //     |      |      |                          |       |
     //     |  55  |  65  | most of Scandinavia      |   55  |
     //     |      |      |                          |       |
     //     +------+------+--------------------------+-------+
     //     |      |      |                          |       |
     //     |  60  |  70  |                          |   60  |
     //     |      |      |                          |       |
     //     +------+------+--------------------------+-------+
     //     |      |      | most of                  |       |
     //     | -50  | -25  |    Chile and Argentina   |  -50  |
     //     |      |      | New Zealand              |       |
     //     +------+------+--------------------------+-------+
     //     |      |      |                          |       |
     //     | -35  | -55  |                          |  -35  |
     //     |      |      |                          |       |
     //     +------+------+--------------------------+-------+
     //     |      |      |                          |       |
     //     | -45  | -60  |                          |  -45  |
     //     |      |      |                          |       |
     //     +------+------+--------------------------+-------+
     //     |      |      |                          |       |
     //     | -55  | -65  |                          |  -55  |
     //     |      |      |                          |       |
     //     +------+------+--------------------------+-------+
     //     |      |      |                          |       |
     //     | -60  | -70  |                          |  -60  |
     //     |      |      |                          |       |
     //     +------+------+--------------------------+-------+
```

Outputs

```
     M1 = (m1,n1) : real * real // longitude and latitude,
           // respectively, of the provided location

   Local Variables

     d, d1, d2, l, r, b, t, x, y: real
     SW, SE, NW, NE: real * real
        // pairs of real numbers, interpreted as coordinates
        // lomgitude and latitude, respectively

     temp : Integer[1..8]

   Function
     choose(Ma, Mb: real * real): real * real;
        // This function chooses either Ma or Mb
        // depending on the parameter 'prob'
        // and on prev-M1, the previous value of M1:
        // If prev-M1 == Ma choose Ma with probability 'prob'
        // If prev-M1 == Mb choose Mb with probability 'prob'
        // Else choose Ma or Mb with probability 1/2
     Begin
     rand:= Random[0,1];
        // a real random number between 0 and 1
     If     prev-M1 == Ma Then
            If rand < prob Then choose := Ma;
                           Else choose := Mb;  EndIf
     Elseif prev-M1 == Mb Then
            If rand < prob Then choose := Mb;
                           Else choose := Ma;  EndIf
     Else
            If rand < 0.5  Then choose := Ma;
                           Else choose := Mb;  EndIf
     End // Function choose

   Main  // main procedure
     Begin
     d := radius/1000;  // uncertainity, measured in km

     d1:= (d * 180) / (pi*M*cos(o));

     d2:= d / 110.6;

     l := d1*floor(m/d1)
           // "floor"  returns the largest integer
           // smaller or equal to a floating point number
     r := l+d1;
     b := o+d2*floor(n-o/d2);
     t := b+d2;
```

```
x := (m-l)/(r-l);
y := (n-b)/(t-b);

SW := (l,b);
SE := (r,b);
NW := (l,t);
NE := (r,t);

If     x < p and y < p      Then M1 := SW;
Elseif x < p and q <= y     Then M1 := NW;
Elseif q <= x and y < p     Then M1 := SE;
Elseif q <= x and q <= y    Then M1 := NE;
Elseif p <= x and x < q and y < x  and y < 1-x
       Then M1 := choose(SW,SE);
Elseif p <= y and y < q and x <= y and y < 1-x
       Then M1 := choose(SW,NW);
Elseif p <= y and y < q and y < x  and 1-x <= y
       Then M1 := choose(SE,NE);
Elseif p <= x and x < q and x <= y and 1-x <= y
       Then M1 := choose(NW,NE);
Endif

End //  Main
```

## Authors' Addresses

Henning Schulzrinne (editor)
Columbia University
Department of Computer Science
450 Computer Science Building
New York, NY 10027
USA

Phone: +1 212 939 7042
Email: schulzrinne@cs.columbia.edu
URI: http://www.cs.columbia.edu/~hgs

Hannes Tschofenig (editor)
Nokia Siemens Networks
Linnoitustie 6
Espoo 02600
Finland

Phone: +358 (50) 4871445
Email: Hannes.Tschofenig@gmx.net
URI: http://www.tschofenig.priv.at

Jorge R. Cuellar
Siemens
Otto-Hahn-Ring 6
Munich, Bavaria 81739
Germany

Email: Jorge.Cuellar@siemens.com

James Polk
Cisco
2200 East President George Bush Turnpike
Richardson, Texas 75082
USA

Email: jmpolk@cisco.com

John B. Morris, Jr.

Email: ietf@jmorris.org

Martin Thomson
Commscope
Andrew Building (39)
University of Wollongong Campus, Wollongong, NSW 2522
AU

Phone: +61 2 4221 2915
Email: martin.thomson@commscope.com

GEOPRIV M. Thomson
Internet-Draft Andrew Corporation
Intended status: Standards Track B. Rosen
Expires: May 3, 2012 Neustar
D. Stanley
Aruba Networks
G. Bajko
Nokia
A. Thomson
Cisco Systems, Inc.
October 31, 2011

# Relative Location Representation
draft-ietf-geopriv-relative-location-02

## Abstract

This document defines an extension to PIDF-LO (RFC4119) for the expression of location information that is defined relative to a reference point. The reference point may be expressed as a geodetic or civic location, and the relative offset may be one of several shapes. Optionally, a reference to a secondary document (such as a map image) can be included, along with the relationship of the map coordinate system to the reference/offset coordinate system to allow display of the map with the reference point and the relative offset. Also included in this document is a Type/Length/Value (TLV) representation of the relative location for use in other protocols that use TLVs.

## Status of This Memo



## Copyright Notice

Table of Contents

## 1. Introduction

This document describes a format for the expression of relative location information.

A relative location is formed of a reference location, plus a relative offset from that reference location. The reference location can be represented in either civic or geodetic form. The reference location can also have dynamic components such as velocity. The relative offset is specified in meters using a Cartesian coordinate system.

In addition to the relative location, an optional URI can be provided to a document that contains a map, floorplan or illustration. Applications could use this information to display the relative location. Additional fields allow the map to be oriented and scaled correctly.

Two formats are included: an XML form that is intended for use in PIDF-LO [RFC4119] and a TLV format for use in other protocols such as those that already convey binary representation of location information defined in [RFC4776].

## 2. Conventions used in this document

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].

## 3. Overview

This document describes an extension to PIDF-LO [RFC4119] as updated by [RFC5139] and [RFC5491], to allow the expression of a location as an offset relative to a reference.

This extension effectively allows the creator of a location object to include two location values plus an offset. The "baseline" location that is given outside of the <relative-location> element is what will be visible to a client that does not understand that extension (i.e., one that ignores the <relative-location> element). A client that does understand this extension will interpret the location within the relative element as a refinement of the baseline location, which gives the reference location for the relative offset.

Creators of location objects with relative location thus have a choice of how much information to put into the "baseline" location and how much to put into the "reference" location. For example, all location information could be put inside the <relative-location>

element, so that clients that do not understand relative location would receive no location information at all. Alternatively, the baseline location value could be precise enough to specify a building that contains the relative location, and the reference location could specify a point within the building from which the offset is measured.

The baseline location SHOULD be general enough to describe both the reference location and the relative location (reference plus offset). In particular, while it is possible to put all location information into the "reference" location (leaving an universally broad "baseline"), location objects SHOULD NOT have all location information in the baseline location. Doing this would cause clients that do not understand relative location to incorrectly interpret the baseline location (i.e., the reference point) as the actual, precise location of the client.

Both the baseline and the reference location are defined either as a geodetic location [OGC.GeoShape] or a civic address [RFC4776]. If the baseline location was expressed as a geodetic location, the reference MUST be geodetic. If the baseline location was expressed as a civic address, the reference MUST be a civic.

Baseline and reference locations MAY also include dynamic location information [RFC5962].

The relative location can be expressed using a point (2- or 3-dimensional), or a shape that includes uncertainty: circle, sphere, ellipse, ellipsoid, polygon, prism or arc-band. Descriptions of these shapes can be found in [RFC5491].

Optionally, a reference to a 'map' document can be provided. The reference is a URI. The document could be an image or dataset that represents a map, floorplan or other form. The type of document the URI points to is described as a MIME media type. Metadata in the relative location can include the location of the reference point in the map as well as an orientation (angle from North) and scale to align the document CRS with the WGS-84 CRS. The document is assumed to be useable by the application receiving the PIDF with the relative location to locate the reference point in the map. This document does not describe any mechanisms for displaying or manipulating the document other than providing the reference location, orientation and scale.

As an example, consider a relative location expressed as a point, relative to a civic location:

```
<presence xmlns="urn:ietf:params:xml:ns:pidf"
```

```
          xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
          xmlns:gp="urn:ietf:params:xml:ns:pidf:geopriv10"
          xmlns:ca="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
          xmlns:rel="urn:ietf:params:xml:ns:pidf:geopriv10:relative"
          xmlns:gml="http://www.opengis.net/gml"
          xmlns:gs="http://www.opengis.net/pidflo/1.0"
          entity="pres:relative@example.com">
  <dm:device id="relative1">
    <gp:geopriv>
      <gp:location-info>
        <ca:civicAddress xml:lang="en-AU">
          <ca:country>AU</ca:country>
          <ca:A1>NSW</ca:A1>
          <ca:A3>Wollongong</ca:A3>
          <ca:A4>North Wollongong</ca:A4>
          <ca:RD>Flinders</ca:RD>
          <ca:STS>Street</ca:STS>
          <ca:HNO>123</ca:HNO>
        </ca:civicAddress>
        <rel:relative-location>
          <rel:reference>
            <ca:civicAddress xml:lang="en-AU">
              <ca:INT N="Door" R="A">Front</ca:INT>
            </ca:civicAddress>
          </rel:reference>
          <rel:offset>
            <gml:Point xmlns:gml="http://www.opengis.net/gml"
                       srsName="urn:ietf:params:geopriv:relative:2d">
              <gml:pos>100 50</gml:pos>
            </gml:Point>
          </rel:offset>
        </rel:relative-location>
      </gp:location-info>
      <gp:usage-rules/>
      <gp:method>GPS</gp:method>
      <rel:map>
        <rel:url type="image/png">
           http://example.com/location/map.png
        </rel:url>
        <rel:offset>20. 120.</rel:offset>
        <rel:orientation>29.</rel:orientation>
        <rel:scale>20. -20.</rel:scale>
      </rel:map>
    </gp:geopriv>
    <dm:deviceID>mac:1234567890ab</dm:deviceID>
    <dm:timestamp>2007-06-22T20:57:29Z</dm:timestamp>
  </dm:device>
</presence>
```

## 4. Relative Location

Relative location is a shape (point, circle, ellipse...). The shape is defined with a CRS that has a datum defined as the reference (which appears as a civic address or geodetic location in the tuple), and the shape coordinates as meter offsets North/East of the datum measured in meters (with an optional Z offset relative to datum altitude). An optional angle allows the reference CRS be to rotated with respect to North.

### 4.1. Relative Coordinate System

The relative coordinate reference system uses a coordinate system with two or three axes.

The baseline and reference locations are used to define a relative datum. The reference location defines the origin of the coordinate system. The centroid of the reference location is used when the reference location contains any uncertainty.

The axes in this coordinate system are originally oriented based on the directions of East, North and Up from the reference location: the first (x) axis increases to the East, the second (y) axis points North, and the optional third (z) axis points Up. All axes of the coordinate system use meters as a basic unit.

Any coordinates in the relative shapes use the described Cartesian coordinate system. In the XML form, this uses a URN of "urn:ietf:params:geopriv:relative:2d" for two-dimensional shapes and "urn:ietf:params:geopriv:relative:3d" for three-dimensional shapes. The binary form uses different shape type identifiers for 2D and 3D shapes.

Dynamic location information [RFC5962] in the baseline or reference location alters relative coordinate system. The resulting Cartesian coordinate system axes are rotated so that the ’y’ axis is oriented along the direction described by the <orientation> element. The coordinate system also moves as described by the <speed> and <heading> elements.

### 4.2. Placement of XML Elements

The baseline of the reference location is represented as <location-info> like a normal PIDF-LO. Relative location adds a new <relative-location> element to <location-info> Within <relative-location> <reference> and <offset> elements are described. Within <offset> are shape elements described below. This document extends PIDF-LO as described in [I-D.ietf-geopriv-local-civic].

### 4.3. Binary Format

This document describes a way to encode the relative location in a binary TLV form for use in other protocols that use TLVs to represent location.

A type-length-value encoding is used.

```
+------+------+-----+------+-----+-----+------+
| Type |Length|  Value                      ...
+------+------+-----+------+-----+------+
|  X   |  N   |  Value label                ...
+------+------+-----+------+-----+-----+------+
```

Figure 1: TLV-tuple format

Type field (X) is defined as a single byte. The type codes used are registered an IANA managed 'RLtypes' registry defined by this document, and restricted to not include the values defined by the CAtypes registry. This restriction permits a location reference and offset to be coded with unique TLVs.

The Length field (N) is defined as an unsigned integer that is one byte in length. This field can encode values from 0 to 255. The length field describes the number of bytes in the Value. Length does not count the bytes used for the Type or Length.

The Value field is defined separately for each type.

Each element of the relative location has a unique TLV assignment. A relative location encoded in TLV would have the baseline location TLVs, a reference location TLV which contains within it the reference refinement TLVs. The reference TLVs are followed by the relative offset, and optional map TLDs described in this document.

### 4.4. Distances and Angles

All distance measures used in shapes are expressed in meters.

All orientation angles used in shapes are expressed in degrees. Orientation angles are measured from WGS84 Northing to Easting with zero at Northing. Orientation angles in the relative coordinate system start from the second coordinate axis (y or Northing) and increase toward the first axis (x or Easting).

## 4.5. Value Encoding

The binary form uses single-precision floating point values [IEEE754] to represent coordinates, distance and angle measures. Single precision values are 32-bit values with a sign bit, 8 exponent bits and 23 fractional bits.

Binary-encoded coordinate values are considered to be a single value without uncertainty. When encoding a value that cannot be exactly represented, the best approximation is chosen according to [Clinger1990].

## 4.6. Relative Location Restrictions

More than one relative shape MUST NOT be included in either a PIDF-LO or TLV encoding of location for a given reference point.

Any error in the reference point transfers to the location described by the relative location. Any errors arising from an implementation not supporting or understanding elements of the reference point directly increases the error (or uncertainty) in the resulting location.

## 4.7. Baseline TLVs

Baseline locations is described using the formats defined in [RFC4776] or [RFC6225].

## 4.8. Reference TLV

When a reference is encoded in binary form, the baseline and reference locations are combined in a reference TLV. This TLV contains civic address TLVs (if the baseline was a civic) or geo TLVs (if the baseline was a geo).

```
+------+------+------+------+-----+------+
|  111 |Length|  Reference TLVs         |
+------+------+------+------+-----+------+
```

Reference TLV

## 4.9. Shapes

Shape data is used to represent regions of uncertainty for the reference and relative locations. Shape data in the reference location uses a [WGS84] CRS. Shape data in the relative location uses a relative CRS.

The XML form for shapes uses Geography Markup Language (GML) [OGC.GML-3.1.1], consistent with the rules in [RFC5491]. Reference locations use the CRS URNs specified in [RFC5491]; relative locations use either a 2D CRS (urn:ietf:params:geopriv:relative:2d), or a 3D (urn:ietf:params:geopriv:relative:3d), depending on the shape type.

The binary form of each shape uses a different shape types for 2d and 3d shapes.

Nine shape type codes are defined.

#### 4.9.1. Point

A point "shape" describes a single point with unknown uncertainty. It consists of a single set of coordinates.

In a two-dimensional CRS, the coordinate includes two values; in a three-dimensional CRS, the coordinate includes three values.

##### 4.9.1.1. XML encoding

A point is represented in GML using the following template:

```
<gml:Point xmlns:gml="http://www.opengis.net/gml"
           srsName="$CRS-URN$">
  <gml:pos>$Coordinate-1 $Coordinate-2$ $Coordinate-3$</gml:pos>
</gml:Point>
```

GML Point Template

Where "$CRS-URN$" is replaced by a urn:ietf:params:geopriv:relative:2d or urn:ietf:params:geopriv:relative:3d and "$Coordinate-3$" is omitted if the CRS is two-dimensional.

##### 4.9.1.2. TLV encoding

The point shape is introduced by a TLV of 113 for a 2D point and 114 for a 3D point.

```
+------+------+
| 113/4|Length|
+------+------+------+------+
|  Coordinate-1             |
+------+------+------+------+
|  Coordinate-2             |
+------+------+------+------+
|  (3D-only) Coordinate-3   |
```

```
+------+------+-----+------+
```

Point Encoding

#### 4.9.2. Circle or Sphere Shape

A circle or sphere describes a single point with a single uncertainty value in meters.

In a two-dimensional CRS, the coordinate includes two values and the resulting shape forms a circle. In a three-dimensional CRS, the coordinate includes three values and the resulting shape forms a sphere.

##### 4.9.2.1. XML encoding

A circle is represented in and converted from GML using the following template:

```
<gs:Circle xmlns:gml="http://www.opengis.net/gml"
           xmlns:gs="http://www.opengis.net/pidflo/1.0"
           srsName="urn:ietf:params:geopriv:relative:2d">
  <gml:pos>$Coordinate-1 $Coordinate-2$</gml:pos>
  <gs:radius uom="urn:ogc:def:uom:EPSG::9001">
    $Radius$
  </gs:radius>
</gs:Circle>
```

GML Circle Template

A sphere is represented in and converted from GML using the following template:

```
<gs:Sphere xmlns:gml="http://www.opengis.net/gml"
           xmlns:gml="http://www.opengis.net/pidflo/1.0"
           srsName="urn:ietf:params:geopriv:relative:3d">
  <gml:pos>$Coordinate-1 $Coordinate-2$ $Coordinate-3$</gml:pos>
  <gs:radius uom="urn:ogc:def:uom:EPSG::9001">
    $Radius$
  </gs:radius>
</gs:Sphere>
```

GML Sphere Template

#### 4.9.2.2. TLV encoding

A circular shape is introduced by a type code of 115. A spherical shape is introduced by a type code of 116.

```
+------+------+
| 115/6|Length|
+------+------+-----+------+
|  Coordinate-1            |
+------+------+-----+------+
|  Coordinate-2            |
+------+------+-----+------+
|  (3D-only) Coordinate-3  |
+------+------+-----+------+
|  Radius                  |
+------+------+-----+------+
```

Circle or Sphere Encoding

### 4.9.3. Ellipse or Ellipsoid Shape

A ellipse or ellipsoid describes a point with an elliptical or ellipsoidal uncertainty region.

In a two-dimensional CRS, the coordinate includes two values, plus a semi-major axis, a semi-minor axis, a semi-major axis orientation (clockwise from North). In a three-dimensional CRS, the coordinate includes three values and in addition to the two-dimensional values, an altitude uncertainty (semi-vertical) is added.

#### 4.9.3.1. XML encoding

An ellipse is represented in and converted from GML using the following template:

```
<gs:Ellipse xmlns:gml="http://www.opengis.net/gml"
            xmlns:gs="http://www.opengis.net/pidflo/1.0"
            srsName="urn:ietf:params:geopriv:relative:2d">
  <gml:pos>$Coordinate-1 $Coordinate-2$</gml:pos>
  <gs:semiMajorAxis uom="urn:ogc:def:uom:EPSG::9001">
    $Semi-Major$
  </gs:semiMajorAxis>
  <gs:semiMinorAxis uom="urn:ogc:def:uom:EPSG::9001">
    $Semi-Minor$
  </gs:semiMinorAxis>
  <gs:orientation uom="urn:ogc:def:uom:EPSG::9102">
    $Orientation$
  </gs:orientation>
</gs:Ellipse>
```

GML Ellipse Template

An ellipsoid is represented in and converted from GML using the following template:

```
<gs:Ellipsoid xmlns:gml="http://www.opengis.net/gml"
              xmlns:gs="http://www.opengis.net/pidflo/1.0"
              srsName="urn:ietf:params:geopriv:relative:3d">
  <gml:pos>$Coordinate-1 $Coordinate-2$ $Coordinate-3$</gml:pos>
  <gs:semiMajorAxis uom="urn:ogc:def:uom:EPSG::9001">
    $Semi-Major$
  </gs:semiMajorAxis>
  <gs:semiMinorAxis uom="urn:ogc:def:uom:EPSG::9001">
    $Semi-Minor$
  </gs:semiMinorAxis>
  <gs:verticalAxis uom="urn:ogc:def:uom:EPSG::9001">
    $Semi-Vertical$
  </gs:verticalAxis>
  <gs:orientation uom="urn:ogc:def:uom:EPSG::9102">
    $Orientation$
  </gs:orientation>
</gs:Ellipsoid>
```

GML Ellipsoid Template

#### 4.9.3.2. TLV encoding

An ellipse is introduced by a type code of 117 and an ellipsoid is introduced by a type code of 118.

```
+------+------+
| 117/8|Length|
+------+------+-----+------+
|  Coordinate-1            |
+------+------+-----+------+
|  Coordinate-2            |
+------+------+-----+------+
|  (3D-only) Coordinate-3  |
+------+------+-----+------+------+------+-----+------+
|  Semi-Major Axis         |  Semi-Minor Axis         |
+------+------+-----+------+------+------+-----+------+
|  Orientation             |  (3D) Semi-Vertical Axis |
+------+------+-----+------+------+------+-----+------+
```

Ellipse or Ellipsoid Encoding

### 4.9.4. Polygon or Prism Shape

A polygon or prism include a number of points that describe the outer boundary of an uncertainty region. A prism also includes an altitude for each point and prism height.

At least 3 points MUST be included in a polygon. In order to interoperate with existing systems, an encoding SHOULD include 15 or fewer points, unless the recipient is known to support larger numbers.

#### 4.9.4.1. XML Encoding

A polygon is represented in and converted from GML using the following template:

```
<gml:Polygon xmlns:gml="http://www.opengis.net/gml"
             srsName="urn:ietf:params:geopriv:relative:2d">
  <gml:exterior>
    <gml:LinearRing>
      <gml:posList>
        $Coordinate1-1$ $Coordinate1-2$
        $Coordinate2-1$ $Coordinate2-2$
        $Coordinate3-1$ ...
        ...
        $CoordinateN-1$ $CoordinateN-2$
        $Coordinate1-1$ $Coordinate1-2$
      </gml:posList>
    </gml:LinearRing>
  </gml:exterior>
</gml:Polygon>
```

GML Polygon Template

Alternatively, a series of "pos" elements can be used in place of the single "posList". Each "pos" element contains two or three coordinate values.

Note that the first point is repeated at the end of the sequence of coordinates and no explicit count of the number of points is provided.

A GML polygon that includes altitude cannot be represented completely in binary. When converting to the binary representation, a two dimensional CRS is used and altitude is removed from each coordinate.

A prism is represented in and converted from GML using the following template:

```
<gs:Prism xmlns:gml="http://www.opengis.net/gml"
          xmlns:gs="http://www.opengis.net/pidflo/1.0"
          srsName="urn:ietf:params:geopriv:relative:3d">
  <gs:base>
    <gml:Polygon>
      <gml:exterior>
        <gml:LinearRing>
          <gml:posList>
            $Coordinate1-1$ $Coordinate1-2$ $Coordinate1-3$
            $Coordinate2-1$ $Coordinate2-2$ $Coordinate2-3$
            $Coordinate2-1$ ... ...
            ...
            $CoordinateN-1$ $CoordinateN-2$ $CoordinateN-3$
            $Coordinate1-1$ $Coordinate1-2$ $Coordinate1-3$
          </gml:posList>
        </gml:LinearRing>
      </gml:exterior>
    </gml:Polygon>
  </gs:base>
  <gs:height uom="urn:ogc:def:uom:EPSG::9001">
    $Height$
  </gs:height>
</gs:Prism>
```

GML Prism Template

Alternatively, a series of "pos" elements can be used in place of the single "posList". Each "pos" element contains three coordinate values.

#### 4.9.4.2. TLV Encoding

A polygon containing 2D points is uses a type code of 119. A polygon with 3D points uses a type code of 120. A prism uses a type code of 121.

```
+------+------+
|119-21|Length|
+------+------+-----+------+-----+------+
|  Count      |  (3D-only) Height        |
+------+------+-----+------+-----+------+
|  Coordinate1-1            |
+------+------+-----+------+
|  Coordinate1-2            |
+------+------+-----+------+
|  (3D-only) Coordinate1-3  |
+------+------+-----+------+
|  Coordinate2-1            |
+------+------+-----+------+
 ...
+------+------+-----+------+
|  CoordinateN-1            |
+------+------+-----+------+
|  CoordinateN-2            |
+------+------+-----+------+
|  (3D-only) CoordinateN-3  |
+------+------+-----+------+
```

Polygon or Prism Encoding

Note that unlike the polygon representation in GML, the first and last points are not the same point to be the same in the TLV representation. The duplicated point is removed from the binary form.

### 4.9.5. Arc-Band Shape

A arc-band describes a region constrained by a range of angles and distances from a predetermined point. This shape can only be provided for a two-dimensional CRS.

Distance and angular measures are defined in meters and degrees respectively. Both are encoded as single precision floating point values.

#### 4.9.5.1. XML encoding

An arc-band is represented in and converted from GML using the following template:

```
<gs:ArcBand xmlns:gml="http://www.opengis.net/gml"
            xmlns:gs="http://www.opengis.net/pidflo/1.0"
            srsName="urn:ietf:params:geopriv:relative:2d">
  <gml:pos>$Coordinate-1 $Coordinate-2$</gml:pos>
  <gs:innerRadius uom="urn:ogc:def:uom:EPSG::9001">
    $Inner-Radius$
  </gs:innerRadius>
  <gs:outerRadius uom="urn:ogc:def:uom:EPSG::9001">
    $Inner-Radius$
  </gs:outerRadius>
  <gs:startAngle uom="urn:ogc:def:uom:EPSG::9102">
   $Start-Angle$
  </gs:startAngle>
  <gs:openingAngle uom="urn:ogc:def:uom:EPSG::9102">
    $Opening-Angle$
  </gs:openingAngle>
</gs:Ellipsoid>
```

GML Arc-Band Template

#### 4.9.5.2. TLV Encoding

An arc-band is introduced by a type code of 122.

```
+------+------+
| 122  |Length|
+------+------+-----+------+
|  Coordinate              |
+------+------+-----+------+
|  Coordinate              |
+------+------+-----+------+------+------+-----+------+
|  Inner Radius            |  Outer Radius            |
+------+------+------+------+------+------+-----+------+
|  Start Angle             |  Opening Angle           |
+------+------+-----+------+------+------+-----+------+
```

Arc-Band Encoding

### 4.10. Dynamic Location TLVs

Dynamic location elements use the definitions in [RFC5962].

### 4.10.1. Orientation

The orientation of the target is described using one or two angles.

```
+------+------+
| 123  |Length|
+------+------+-----+------+
|         Angle            |
+------+------+-----+------+
|   (Optional) Angle       |
+------+------+-----+------+
```

Dynamic Orientation TLVs

### 4.10.2. Speed

The speed of the target is a scalar value in meters per second.

```
+------+------+
| 124  |Length|
+------+------+-----+------+
|         Length           |
+------+------+-----+------+
|         Speed            |
+------+------+-----+------+
```

Dynamic Speed TLVs

### 4.10.3. Heading

The heading, or direction of travel, is described using one or two angles.

```
+------+------+
| 125  |Length|
+------+------+-----+------+
|         Angle            |
+------+------+-----+------+
|   (Optional) Angle       |
+------+------+-----+------+
```

Dynamic Heading TLVs

### 4.11. Secondary Map Metadata

The optional "map" URL can be used to provide a user of relative location with a visual reference for the location information. This document does not describe how the recipient uses the map nor how it locates the reference or offset within the map. Maps can be simple images, vector files, 2-D or 3-D geospatial databases, or any other form of representation understood by both the sender and recipient.

#### 4.11.1. Map URL

In XML, the map is a <map> element defined within <relative-location> and contains the URL. The URL is encoded as a UTF-8 encoded string. An "http:" or "https:" URL MUST be used unless the entity creating the PIDF-LO is able to ensure that authorized recipients of this data are able to use other URI schemes. A "type" attribute MUST be present and specifies the kind of map the URL points to. Map types are specified as mime media types as recorded in the IANA Media Types registry. For example <map type="image/png">https://www.example.com/floorplans/123South/floor-2</map>. In binary, the map type is a separate TLV from the map URL:

```
+------+------+------+------+------+--  --+------+
|  126 |Length|   Map Media Type         ...
+------+------+------+------+------+--  --+------+
|  127 |Length|   Map Image URL          ...
+------+------+------+------+------+--  --+------+
```

Map URL TLVs

#### 4.11.2. Map Coordinate Reference System

The CRS used by the map depends on the type of map. For example, a map described by a 3-D geometric model of the building may contain a complete CRS description in it. For some kinds of maps, typically described as images, the CRS used within the map must define the following:

- o The CRS origin
- o The CRS axes used and their orientation
- o The unit of measure used

This document provides elements that allow for a mapping between the local coordinate reference system used for the relative location and the coordinate reference system used for the map where they are not the same.

##### 4.11.2.1. Map Reference Point Offset

This optional element identifies the coordinates of the reference point as it appears in the map. This value is measured in a map-type dependent manner, using the coordinate system of the map.

For image maps, coordinates start from the upper left corner and coordinates are first counted by column with positive values to the right; then rows are counted with positive values toward the bottom of the image. For such an image, the first item is columns, the second rows and any third value applies to any third dimension used in the image coordinate space.

The <offset> element contains 2 (or 3) coordinates similar to a GML "pos", For example:

```
<offset> 2670.0 1124.0 1022.0</offset>
```

Map Reference Point Example XML

```
+------+------+
| 128  |Length|
+------+------+------+------+
|  Coordinate-1             |
+------+------+------+------+
|  Coordinate-2             |
+------+------+------+------+
|  (3D-only) Coordinate-3   |
+------+------+------+------+
```

Map Reference Point Coordinates TLV

If omitted, a value containing all zeros is assumed. If the coordinates provided contain fewer values than are needed, the first value from the set is applied in place of any missing values.

#### 4.11.2.2. Map Orientation

The map orientation includes the orientation of the map direction in relation to the Earth. Map orientation is expressed relative to the orientation of the relative coordinate system. This means that map orientation with respect to WGS84 North is the sum of th orientation field, plus any orientation included in a dynamic portion of the reference location. Both values default to zero if no value is specified.

This type uses a single precision floating point value of degrees relative to North.

In XML, the <orientation> element contains a single floating point value, example <orientation>67.00</orientation>. In TLV form:

```
+------+------+------+------+------+------+
|  129 |Length|  Angle                    |
+------+------+------+------+------+------+
```

Map Orientation TLV

#### 4.11.2.3. Map Scale

The optional map scale describes the relationship between the units of measure used in the map, relative to the meters unit used in the relative coordinate system.

This type uses a sequence of IEEE 754 [IEEE.754] single precision floating point values to represent scale as a sequence of numeric values. The units of these values is dependent on the type of map, and could for example be pixels per meter for an image.

A scaling factor is provided for each axis in the coordinate system. For a two-dimensional coordinate system, two values are included to allow for different scaling along the x and y axes independently. For a three-dimensional coordinate system, three values are specified for the x, y and z axes.

Alternatively, a single scaling value MAY be used to apply the same scaling factor to all coordinate components.

Images that use a rows/columns coordinate system often use a left-handed coordinate system. A negative value for the y/rows-axis scaling value can be used to account for any change in direction between the y-axis used in the relative coordinate system and the rows axis of the image coordinate system.

In XML, the <scale> element may contain the single scale value, or may contain 2 (or 3) values similar to a GML "pos" with separate scale values. In TLV form:

```
+------+------+------+------+------+
|  130 |Length|  Scales       ...
+------+------+------+------+------+
```

Map Scale TLV

### 4.11.3. Map Example

An example of expressing a map is:

```
          <rel:map>
            <rel:url type="image/jpeg">
              http://example.com/map.jpg
            </rel:url>
            <rel:offset>200 210</rel:offset>
            <rel:orientation>68</rel:orientation>
            <rel:scale>2.90 -2.90</rel:scale>
          </rel:map>
```

Map Example

# 5. Examples

## 5.1. Civic PIDF with Polygon Offset


```
    <presence xmlns="urn:ietf:params:xml:ns:pidf"
              xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
              xmlns:gp="urn:ietf:params:xml:ns:pidf:geopriv10"
              xmlns:ca="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
              xmlns:rel="urn:ietf:params:xml:ns:pidf:geopriv10:relative"
              xmlns:gml="http://www.opengis.net/gml"
              xmlns:gs="http://www.opengis.net/pidflo/1.0"
              entity="pres:ness@example.com">
      <dm:device id="nesspc-1">
        <gp:geopriv>
          <gp:location-info>
            <ca:civicAddress xml:lang="en-AU">
              <ca:country>AU</ca:country>
              <ca:A1>NSW</ca:A1>
              <ca:A3>Wollongong</ca:A3>
              <ca:A4>North Wollongong</ca:A4>
              <ca:RD>Flinders</ca:RD>
              <ca:STS>Street</ca:STS>
              <ca:HNO>123</ca:HNO>
            </ca:civicAddress>
            <rel:relative-location>
              <rel:reference>
                <ca:civicAddress xml:lang="en-AU">
                  <ca:INT N="Building">A</ca:INT>
                  <ca:INT N="Level">I</ca:INT>
                  <ca:INT N="Suite">113</ca:INT>
                  <ca:INT N="Door" R="A">Front</ca:INT>
                </ca:civicAddress>
              </rel:reference>
              <rel:offset>
                 <gml:Polygon xmlns:gml="http://www.opengis.net/gml"
                      srsName="urn:ietf:params:geopriv:relative:2d">
                   <gml:exterior>
```

```
                     <gml:LinearRing>
                       <gml:pos>433.0 -734.0</gml:pos> <!--A-->
                       <gml:pos>431.0 -733.0</gml:pos> <!--F-->
                       <gml:pos>431.0 -732.0</gml:pos> <!--E-->
                       <gml:pos>433.0 -731.0</gml:pos> <!--D-->
                       <gml:pos>434.0 -732.0</gml:pos> <!--C-->
                       <gml:pos>434.0 -733.0</gml:pos> <!--B-->
                       <gml:pos>433.0 -734.0</gml:pos> <!--A-->
                     </gml:LinearRing>
                   </gml:exterior>
                </gml:Polygon>
              <rel:offset>
            </rel:relative-location>
          </gp:location-info>
         <gp:usage-rules/>
          <gp:method>GPS</gp:method>
        </gp:geopriv>
        <dm:deviceID>mac:1234567890ab</dm:deviceID>
        <dm:timestamp>2007-06-22T20:57:29Z</dm:timestamp>
      </dm:device>
    </presence>
```

## 5.2. Geo PIDF with Circle Offset

```
<?xml version="1.0" encoding="UTF-8"?>
    <presence xmlns="urn:ietf:params:xml:ns:pidf"
              xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
              xmlns:gp="urn:ietf:params:xml:ns:pidf:geopriv10"
              xmlns:rel="urn:ietf:params:xml:ns:pidf:geopriv10:relative"
              xmlns:gml="http://www.opengis.net/gml"
              entity="pres:point2d@example.com">
      <dm:device id="point2d">
        <gp:geopriv>
          <gp:location-info>
            <gml:Circle srsName="urn:ogc:def:crs:EPSG::4326">
              <gml:pos>-34.407 150.883</gml:pos>
              <gs:radius uom="urn:ogc:def:uom:EPSG::9001">
                     50.0
              </gs:radius>
            </gml:Circle>
            <rel:relative-location>
              <rel:reference>
                <gml:Point srsName="urn:ogc:def:crs:EPSG::4326">
                  <gml:pos>-34.407 150.883</gml:pos>
                </gml:Point>
              </rel:reference>
              <rel:offset>
                <gml:Circle xmlns:gml="http://www.opengis.net/gml"
```

```
                      srsName="urn:ietf:params:geopriv:relative:2d">
                    <gml:pos>500.0 750.0</gml:pos>
                    <gml:radius uom="urn:ogc:def:uom:EPSG::9001">
                       5.0
                     </gml:radius>
               </gml:Circle>
            </rel:relative-location>
            <map:map>
              <map:urltype="image/png">
                https://www.example.com/flrpln/123South/flr-2</gp:url>
              <map:offset> 2670.0 1124.0 1022.0</gp:offset>
              <map:orientation>67.00</gp:orientation>
              <map:scale>10</gp:scale>
            </map:map>
          </gp:location-info>
          <gp:usage-rules/>
          <gp:method>Wiremap</gp:method>
        </gp:geopriv>
        <dm:deviceID>mac:1234567890ab</dm:deviceID>
        <dm:timestamp>2007-06-22T20:57:29Z</dm:timestamp>
      </dm:device>
     </gp:geopriv>
    </status>
    <timestamp>2003-06-22T20:57:29Z</timestamp>
   </tuple>
  </presence>
```

## 5.3. Civic TLV with Point Offset

| Type | Value |
|---|---|
| 0 | en |
| 1 | IL |
| 3 | Chicago |
| 34 | Wacker |
| 18 | Drive |
| 19 | 3400 |
| 112 | Reference |
| 40 | BBuilding\|A |
| 40 | AFloor\|6th |
| 40 | BSuite\|213 |
| 40 | ADoor\|Front |
| 115 | 100 70 |
| 126 | image/png |
| 127 | http://maps.example.com/3400Wacker/A6 |
| 128 | 0.0 4120.0 |
| 129 | 113.0 |
| 130 | 10.6 |

## 6. Schema Definition

```
<?xml version="1.0"?>
<xs:schema
    xmlns:rel="urn:ietf:params:xml:ns:pidf:geopriv10:relative"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:gml="http://www.opengis.net/gml"
    targetNamespace="urn:ietf:params:xml:ns:pidf:geopriv10:relative"
```

```
     elementFormDefault="qualified"
     attributeFormDefault="unqualified">

   <!-- [[NOTE TO RFC-EDITOR: Please replace all instances of the URL
        'http://ietf.org/rfc/rfcXXXX.txt' with the URL of published
        document and remove this note.]] -->

   <xs:annotation>
     <xs:appinfo
         source="urn:ietf:params:xml:schema:pidf:geopriv10:relative">
       Relative Location for PIDF-LO
     </xs:appinfo>
     <xs:documentation source="http://ietf.org/rfc/rfcXXXX.txt">
       This schema defines a location representation that allows for
       the description of locations that are relative to another.
       An optional map reference is also defined.
     </xs:documentation>
   </xs:annotation>

   <xs:import namespace="http://www.opengis.net/gml"/>

   <xs:element name="relative-location" type="rel:relativeType"/>

   <xs:complexType name="relativeType">
     <xs:complexContent>
       <xs:restriction base="xs:anyType">
         <xs:sequence>
           <xs:element name="reference" type="rel:referenceType"/>
           <xs:element name="offset" type="rel:offsetType"/>
           <xs:any namespace="##any" processContents="lax"
                   minOccurs="0" maxOccurs="unbounded"/>
         </xs:sequence>
         <xs:anyAttribute namespace="##other" processContents="lax"/>
       </xs:restriction>
     </xs:complexContent>
   </xs:complexType>

   <xs:complexType name="referenceType">
     <xs:complexContent>
       <xs:restriction base="xs:anyType">
         <xs:sequence>
           <xs:any namespace="##other" processContents="lax"
                   minOccurs="0" maxOccurs="unbounded"/>
         </xs:sequence>
       </xs:restriction>
     </xs:complexContent>
   </xs:complexType>
```

```
<xs:complexType name="offsetType">
  <xs:complexContent>
    <xs:restriction base="xs:anyType">
      <xs:sequence>
        <xs:element ref="gml:_Geometry"/>
        <xs:any namespace="##other" processContents="lax"
                minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:element name="map" type="rel:mapType"/>
<xs:complexType name="mapType">
  <xs:complexContent>
    <xs:restriction base="xs:anyType">
      <xs:sequence>
        <xs:element name="url" type="rel:mapUrlType"/>
        <xs:element name="offset" type="rel:doubleList"
                    minOccurs="0"/>
        <xs:element name="orientation" type="rel:doubleList"
                    minOccurs="0"/>
        <xs:element name="scale" type="rel:doubleList"
                    minOccurs="0"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="mapUrlType">
  <xs:simpleContent>
    <xs:extension base="xs:anyURI">
      <xs:attribute name="type" type="rel:mimeType"
                    default="application/octet-stream"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<!-- From draft-ietf-httpbis-p3-payload-09, excluding
        the obsolete parts -->
<xs:simpleType name="mimeType">
  <xs:restriction base="xs:token">
    <xs:pattern value="[!#$%&amp;'\*\+\-\.\dA-Z^_`a-z\|~]+
    /[!#$%&amp;'\*\+\-\.\dA-Z^_`a-z\|~]+([\t ]*;([\t ])*[!#$%&amp;
    '\*\+\-\.\dA-Z^_`a-z\|~]+=([!#$%&amp;'\*\+\-\.\dA-Z^_`a-z\|~]+|
    &quot;([!#-\[\]-~]|[\t ]*|\\[\t !-~])*&quot;))*"/>
  </xs:restriction>
</xs:simpleType>
```

```
    <xs:simpleType name="doubleList">
      <xs:list itemType="xs:double"/>
    </xs:simpleType>

  </xs:schema>
```

xml schema relative-location

## 7. Security Considerations

This document describes a data format. To a large extent, security properties of this depend on how this data is used.

Privacy for location data is typically important. Adding relative location may increase the precision of the location, but does not otherwise alter its privacy considerations, which are discussed in [RFC4119]

[[Not that interesting, but it could be relevant ?]] The fractional bits in IEEE 754 [IEEE.754] floating point values can be used as a covert channel. For values of either zero or infinity, non-zero fraction bits could be used to convey information. If the presence of covert channels is not desired then the fractional bits MUST be set to zero. There is no need to represent NaN (not a number) in this encoding.

## 8. IANA Considerations

### 8.1. Relative Location Registry

This document creates a new registry called 'Relative Location Parameters'. As defined in [RFC5226], this registry operates under "IETF Consensus" rules.

The content of this registry includes:

Relative Location Code: Numeric identifier, assigned by IANA.

Brief description: Short description identifying the meaning of the element.

Reference to published specification: A stable reference to an RFC which describes the value in sufficient detail so that interoperability between independent implementations is possible.

IANA is requested to not permit values to be assigned into this registry which conflict with values assigned in the CAtypes registry or to permit values to be assigned into the CAtypes registry which conflict with values assigned to to this registry unless the IANA considerations section for the new value explicitly overrides this prohibition, and the document defining the value describes how conflicting TLV codes will be interpreted by implementations

The values defined are:

| RLtype | description | Reference |
|---|---|---|
| 111 | relative location reference | this RFC |
| 112 | relative location angle | this RFC |
| 113 | relative location shape 2D point | this RFC |
| 114 | relative location shape 3D point | this RFC |
| 115 | relative location shape circular | this RFC |
| 116 | relative location shape spherical | this RFC |
| 117 | relative location shape elliptical | this RFC |
| 118 | relative location shape ellipsoid | this RFC |
| 119 | relative location shape 2D polygon | this RFC |
| 120 | relative location shape 3D polygon | this RFC |
| 121 | relative location shape prism | this RFC |
| 122 | relative location shape arc-band | this RFC |
| 123 | relative location dynamic orientation | this RFC |
| 124 | relative location dynamic speed | this RFC |
| 125 | relative location dynamic heading | this RFC |
| 126 | relative location map type | this RFC |
| 127 | relative location map URI | this RFC |
| 128 | relative location map coordinates | this RFC |
| 129 | relative location map angle | this RFC |
| 130 | relative location map scale | this RFC |

## 8.2. URN Sub-Namespace Registration

This document registers a new XML namespace, as per the guidelines in [RFC3688]) that has been registered with IANA.

URI: urn:ietf:params:xml:ns:pidf:geopriv10:relative

Registrant Contact: IETF, GEOPRIV working group (geopriv@ietf.org), Martin Thomson (martin.thomson@andrew.com).

XML:

```
 BEGIN
   <?xml version="1.0"?>
   <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
        "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
   <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
     <head>
       <title>GEOPRIV Relative Location</title>
     </head>
     <body>
       <h1>Format for representing relative location in GEOPRIV</h1>
       <h2>urn:ietf:params:xml:ns:pidf:geopriv10:relative</h2>
       <p>See <a href="http://www.rfc-editor.org/rfc/rfcXXXX.txt">
              RFCXXXX</a>.</p>
     </body>
   </html>
<!-- [[NOTE TO RFC-EDITOR: Please replace all instances of RFCXXXX
     with the number of the published
     document and remove this note.]] -->
     END
```

## 8.3. XML Schema Registration

This section registers an XML schema as per the procedures in [RFC3688].

URI: urn:ietf:params:xml:schema:pidf:geopriv10:relativeLocation

Registrant Contact: IETF, GEOPRIV working group (geopriv@ietf.org), Martin Thomson (martin.thomson@andrew.com).

The XML for this schema can be found as the entirety of Section 7 of this document.

## 8.4. CRS public identifier registration

This section registers two public identifiers as per the procedures in [RFC3688].

URI: urn:ietf:params:xml:ns:pidf:geopriv10:relative:2d

```
   Registrant Contact:  IETF, GEOPRIV working group (geopriv@ietf.org),
      Martin Thomson (martin.thomson@andrew.com).

   XML:

     BEGIN
       <?xml version="1.0"?>
       <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
           "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
       <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
         <head>
           <title>GEOPRIV Relative Location 2d CRS</title>
         </head>
         <body>
           <h1>Identifier for a 2D CRS in GEOPRIV relative location</h1>
           <h2>urn:ietf:params:xml:ns:pidf:geopriv10:relative:2d</h2>
           <p>See <a href="http://www.rfc-editor.org/rfc/rfcXXXX.txt">
                 RFCXXXX</a>.</p>
         </body>
      </html>
  <!-- [[NOTE TO RFC-EDITOR: Please replace all instances of RFCXXXX
       with the number of the published document
       and remove this note.]] -->
    END

    URI:  urn:ietf:params:xml:ns:pidf:geopriv10:relative:3d

    Registrant Contact:  IETF, GEOPRIV working group (geopriv@ietf.org),
      Martin Thomson (martin.thomson@andrew.com).

   XML:

     BEGIN
       <?xml version="1.0"?>
       <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
           "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
       <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
         <head>
           <title>GEOPRIV Relative Location 3d CRS</title>
         </head>
         <body>
           <h1>Identifier for a 3D CRS in GEOPRIV relative location</h1>
           <h2>urn:ietf:params:xml:ns:pidf:geopriv10:relative:3d</h2>
           <p>See <a href="http://www.rfc-editor.org/rfc/rfcXXXX.txt">
                 RFCXXXX</a>.</p>
         </body>
      </html>
  <!-- [[NOTE TO RFC-EDITOR: Please replace all instances of RFCXXXX
```

with the number of the published
document and remove this note.]] -->
END

## 8.5. CAtype Registration

This section adds a new entry to the CAtype registry defined by [I-D.ietf-geopriv-local-civic].

Namespace URI: urn:ietf:params:xml:ns:pidf:geopriv10:relative

Local Name: REL

Description: Relative location from a reference point

Contact: The IESG (iesg@ietf.org); the GEOPRIV working group (geopriv@ietf.org).

Specification: RFCXXXX

Schema: urn:ietf:params:xml:schema:pidf:geopriv10:relativeLocation

Type: A

## 9. Acknowledgements

This is the product of a design team on relative location. Besides the authors, this team included: Marc Linsner, James Polk, and James Winterbottom.

## 10. References

### 10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4119] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, December 2005.

[RFC4776] Schulzrinne, H., "Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information", RFC 4776, November 2006.

[RFC5139] Thomson, M. and J. Winterbottom, "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)", RFC 5139, February 2008.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[RFC5491] Winterbottom, J., Thomson, M., and H. Tschofenig, "GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations", RFC 5491, March 2009.

[RFC5962] Schulzrinne, H., Singh, V., Tschofenig, H., and M. Thomson, "Dynamic Extensions to the Presence Information Data Format Location Object (PIDF-LO)", RFC 5962, September 2010.

[RFC6225] Polk, J., Linsner, M., Thomson, M., and B. Aboba, "Dynamic Host Configuration Protocol Options for Coordinate-Based Location Configuration Information", RFC 6225, July 2011.

[I-D.ietf-geopriv-local-civic] Winterbottom, J., Thomson, M., Barnes, R., Rosen, B., and R. George, "Specifying Civic Address Extensions in PIDF-LO", draft-ietf-geopriv-local-civic-02 (work in progress), October 2011.

[OGC.GML-3.1.1] Cox, S., Daisey, P., Lake, R., Portele, C., and A. Whiteside, "Geographic information - Geography Markup Language (GML)", OpenGIS 03-105r1, April 2004, <http://portal.opengeospatial.org/files/?artifact_id=4700>.

[OGC.GeoShape] Thomson, M. and C. Reed, "GML 3.1.1

PIDF-LO Shape Application Schema for use by the Internet Engineering Task Force (IETF)", OGC Best Practice 06-142r1, Version: 1.0, April 2007.

[IEEE.754] IEEE, "IEEE Standard for Binary Floating-Point Arithmetic", IEEE Standard 754-1985, January 2003.

[Clinger1990] Clinger, W., "How to Read Floating Point Numbers Accurately", Proceedings of Conference on Programming Language Design and Implementation pp. 92-101, 1990, <ftp://ftp.ccs.neu.edu/pub/people/will/howtoread.ps>.

## 10.2. Informative References

[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.

## Authors' Addresses

Martin Thomson
Andrew Corporation
Andrew Building (39)
Wollongong University Campus
Northfields Avenue
Wollongong, NSW 2522
AU

EMail: martin.thomson@andrew.com

Brian Rosen
Neustar
470 Conrad Dr
Mars, PA 16046
US

EMail: br@brianrosen.net

Dorothy Stanley
Aruba Networks
1322 Crossman Ave
Sunnyvale, CA 94089
US

EMail: dstanley@arubanetworks.com

Gabor Bajko
Nokia
323 Fairchild Drive
Mountain View, CA 94043
US

EMail: gabor.bajko@nokia.com

Allan Thomson
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134
US

EMail: althomso@cisco.com