

ECRIT	K. Wolf	
Internet-Draft	nic.at	
Expires: May 13, 2010	November 09, 2009	

[TOC](#)

**Location-to-Service Translation Protocol (LoST) Extension:
ServiceListBoundary
draft-ietf-ecrit-lost-servicelistboundary-01**

Abstract

LoST maps service identifiers and location information to service contact URIs. If a LoST client wants to discover available services for a particular location, it will perform a <listServicesByLocation> query to the LoST server. However, the response from the LoST server does not provide information about the geographical region for which the returned service list is valid. Therefore, this document proposes a ServiceListBoundary to assist the client to not miss a change in available services when moving.

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on May 13, 2010.

Copyright Notice

Copyright (c) 2009 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as

described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the BSD License.

Table of Contents

- [1.](#) Introduction
- [2.](#) Terminology
- [3.](#) LoST Extensions
 - [3.1.](#) Extensions to <ListServiceByLocation>
 - [3.2.](#) Retrieving the serviceList Boundary via getServiceListBoundary
 - [3.3.](#) Service List Boundary
 - [3.4.](#) Implementation Considerations
 - [3.4.1.](#) Server Side
 - [3.4.2.](#) Client Side
- [4.](#) Security & Privacy Considerations
- [5.](#) IANA Considerations
 - [5.1.](#) Relax NG Schema Registration
 - [5.2.](#) Namespace Registration
- [6.](#) Acknowledgement
- [7.](#) Normative References
- [§](#) Author's Address

1. Introduction

[TOC](#)

Location based service providers as well as Public Safety Answering Points (PSAPs) only serve a specific geographic region. Therefore the LoST protocol [\[RFC5222\] \(Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol," August 2008.\)](#) defines the ServiceBoundary, which indicates the service region for a specific service URL. However, not all services are available everywhere. Clients can discover available services for a particular location by the <listServicesByLocation> query in LoST. The LoST server returns a list of services that are available at this particular location. But the server does not inform the client for which geographical region the returned service list is valid. This may lead to the situation where a client initially discovers all available

services by the <listServicesByLocation> query, and then moves to a different location (while refreshing the service mappings), but without noticing the availability of other services. The following imaginary example illustrates the problem for emergency calling:

The client is powered-up, does location determination (resulting in location A) and performs an initial <listServicesByLocation> query with location A requesting urn:services:sos.

The LoST server returns the following services list:

```
urn:service:sos.police
urn:service:sos.ambulance
urn:service:sos.fire
```

The client does the initial LoST mapping and discovers the dialstrings for each service. Then the client moves, refreshing the individual service mappings when necessary as told by the ServiceBoundary.

However, when arriving in location B (close to a mountain), service sos.mountainrescue is available, which was not available in location A. Nevertheless, the client does not detect this, because only the mapping of the initially discovered services (police, ambulance, fire) are refreshed. Consequently, the dialstring for the mountain rescue is not known by the client, and the emergency call to the mountain rescue service will certainly fail.

Note that the ServiceBoundary (service region for an individual service) cannot be considered as an indicator for the region a specific service list is valid for. The service list may even change within the ServiceBoundary of another service. For example, the ambulance mapping is valid for a whole state, but for a part of the state there is an additional mountain rescue service.

Consequently, there are two ways to tackle this issue:

- *clients continuously ask for the service list, although it may not have changed

- *a boundary information (telling the client that the service list does not change inside this area)

Since the LoST protocol has the ServiceBoundary concept in order to avoid that clients continuously try to refresh the mapping of a specific service, a ServiceListBoundary would provide a similar mechanism for service lists.

2. Terminology

[TOC](#)

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 \(Bradner, S.,](#)

["Key words for use in RFCs to Indicate Requirement Levels,"](#)
[March 1997.](#)) [RFC2119].

3. LoST Extensions

[TOC](#)

This chapter describes the necessary modifications to the LoST protocol in order to support the proposed ServiceListBoundary in a similar way as the ServiceBoundary.

3.1. Extensions to <ListServiceByLocation>

[TOC](#)

The query <listServicesByLocation> may contain an additional serviceListBoundary element to request the boundary for the service list, either by value or by reference. In the example below the value of the serviceListBoundary element is set to "value":

```
<?xml version="1.0" encoding="UTF-8"?>
<listServicesByLocation
  xmlns="urn:ietf:params:xml:ns:lost1"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:slb="urn:ietf:params:xml:schema:lost1:slb"
  recursive="true">
  <location id="mylocation" profile="civic">
    <civicAddress
      xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr">
      <country>AT</country>
      <A1>Lower Austria</A1>
      <A3>Wolfsthal</A3>
      <RD>Hauptplatz</RD>
      <HNO>1</HNO>
      <PC>2412</PC>
    </civicAddress>
  </location>
  <service>urn:service:sos</service>
  <slb:serviceListBoundary>value</slb:serviceListBoundary>
</listServicesByLocation>
```

A possible response is shown below:

```

<?xml version="1.0" encoding="UTF-8"?>
<listServicesByLocationResponse
  xmlns="urn:ietf:params:xml:ns:lost1">
  xmlns:slb="urn:ietf:params:xml:schema:lost1:slb"
  <serviceList expires="2010-01-01T00:00:00Z">
    urn:service:sos.ambulance
    urn:service:sos.fire
    urn:service:sos.gas
    urn:service:sos.mountain
    urn:service:sos.poison
    urn:service:sos.police
  </serviceList>
  <path>
    <via source="resolver.example"/>
    <via source="authoritative.example"/>
  </path>
  <locationUsed id="mylocation"/>
  <slb:serviceListBoundary profile="civic">
    <civicAddress
      xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr">
      <country>AT</country>
      <A1>Lower Austria</A1>
    </civicAddress>
  </slb:serviceListBoundary>
</listServicesByLocationResponse>

```

This response above indicates that the service list is valid for Lower Austria. The <listServicesByLocation> request has to be repeated by the client only when moving out of Lower Austria. However, the mappings of the services itself may have other service boundaries. Additionally, the expires attribute indicates the absolute time when this service list becomes invalid.

The boundary can also be requested by reference when setting the attribute serviceListBoundary to "reference". Then the response contains a serviceListBoundaryReference element, as shown below.

```

<?xml version="1.0" encoding="UTF-8"?>
<listServicesByLocationResponse
  xmlns="urn:ietf:params:xml:ns:lost1">
  xmlns:slb="urn:ietf:params:xml:schema:lost1:slb"
  <serviceList expires="2010-01-01T00:00:00Z">
    urn:service:sos.ambulance
    urn:service:sos.fire
    urn:service:sos.gas
    urn:service:sos.mountain
    urn:service:sos.poison
    urn:service:sos.police
  </serviceList>
  <path>
    <via source="resolver.example"/>
    <via source="authoritative.example"/>
  </path>
  <locationUsed id="mylocation"/>
  <serviceListBoundaryReference
    source="authoritative.example"
    serviceListKey="123567890123567890123567890" />
</listServicesByLocationResponse>

```

3.2. Retrieving the serviceList Boundary via getServiceListBoundary

[TOC](#)

In order to retrieve the boundary corresponding a specific serviceListKey, the client issues a <getServiceListBoundary> request, similar to the <getServiceBoundary> request.

An example is shown below:

```

<?xml version="1.0" encoding="UTF-8"?>
<getServiceListBoundary xmlns="urn:ietf:params:xml:ns:lost1"
  serviceListKey="123567890123567890123567890"/>

```

The LoST server response is shown below:

```

<?xml version="1.0" encoding="UTF-8"?>
<getServiceListBoundaryResponse xmlns="urn:ietf:params:xml:schema:lost1:slb">
  <serviceListBoundary profile="civic" expires="2010-01-01T00:00:00Z">
    <civicAddress
      xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr">
      <country>AT</country>
      <A1>Lower Austria</A1>
    </civicAddress>
  </serviceListBoundary>
  <path>
    <via source="resolver.example"/>
    <via source="authoritative.example"/>
  </path>
</getServiceListBoundaryResponse>

```

The serviceListKey uniquely identifies a serviceListBoundary as the key does for the service boundary (see Section 5.6 in RFC 5222). Therefore the serviceListKey is a random token with at least 128 bits of entropy and can be assumed globally unique. Whenever the boundary changes, a new serviceListKey MUST be assigned.

Note: since LoST does not define an attribute to indicate which profile the clients understands in a <getServiceListBoundary> request, this document also does not define one for the <getServiceListBoundary> request.

3.3. Service List Boundary

[TOC](#)

The service list boundary indicates a region within which all <listServicesByLocation> queries with the same service identifiers result in the same serviceList. A service list boundary may consist of geometric shapes (both in civic and geodetic location format), and may be non-contiguous, like the service boundary.

The mapping of the specific services within the service list boundary may be different at different locations.

The server may return the boundary information in multiple profiles, but has to use at least one profile that the client used in the request in order to ensure that the client is able to process the boundary information.

There is no need to include boundary information to a <listServicesResponse>. <ListServices> requests are purely for diagnostic purposes and do not contain location information at all, so no boundary information is reasonable.

Also note that the serviceListBoundary is optional and the LoST server may return it or not based on its local policy - like it is the case with the service boundary. However, especially for emergency services,

the serviceListBoundary might be crucial to ensure that moving clients do not miss changes in the available services.

3.4. Implementation Considerations

[TOC](#)

The subsections below discuss implementations issues for the LoST server and client for the serviceListBoundary support.

3.4.1. Server Side

[TOC](#)

The mapping architecture and framework [\[RFC5582\] \(Schulzrinne, H., "Location-to-URL Mapping Architecture and Framework," September 2009.\)](#) describes that each tree announces its coverage region (for one type of service, e.g. sos.police) to one or more forest guides. Forest guides peer with each other and synchronize their data. Hence, a forest guide has sufficient knowledge (it knows all the services and their coverage regions) to answer a listServicesByLocation query and additionally add the serviceListBoundary as well.

The calculation of the largest possible area for which the service list stays the same might be a complex task. An alternative would be to return smaller areas that are easier to compute. In such a case some unneeded queries to the LoST server are the consequence, but still the main purpose of the serviceListBoundary is achieved: Never miss a change of available services. So a reasonable trade-off between the effort to generate the boundary information and the saved queries to the LoST server has to be considered.

Probably for some countries the county (or disrict, canton, state, ...) borders would be suitable as serviceListBoundary. Some neighbouring counties may have implemented different services while a listServicesByLocation query in other neighbouring counties still results in the same serviceList. So when moving across a county border, it is at least ensured, that every device fetches a new service list from the LoST server.

Other countries might have different structures and the generation of the serviceListBoundary might follow other rules as long as it is ensured that a client is able to notice any change in the service list when moving.

[TOC](#)

3.4.2. Client Side

A mobile client that already implements LoST and evaluates the serviceBoundary has almost everything that is needed to make use of the serviceListBoundary. Since the integration into LoST follows the concept of the serviceBoundary (and also makes use of the same location profiles), just the additional serviceListBoundary has to be evaluated. Whenever moving outside a serviceListBoundary, the client must perform a new listServicesByLocation query with the new location information in order to determine a change in available services.

4. Security & Privacy Considerations

[TOC](#)

Security considerations are discussed in [\[RFC5222\] \(Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol," August 2008.\)](#).

5. IANA Considerations

[TOC](#)

This document requests two actions by IANA: a XML schema registration and namespace registration, according to the description in the following sections.

5.1. Relax NG Schema Registration

[TOC](#)

This document requests registration of the following Relax NG Schema to the IETF XML Registry [\[RFC3688\] \(Mealling, M., "The IETF XML Registry," January 2004.\)](#):

URI: urn:ietf:params:xml:schema:lost1:slb

Registrant Contact: IETF ECRIT Working Group, Karl Heinz Wolf
(karlheinz.wolf@nic.at)

Relax NG Schema:

BEGIN

```
<?xml version="1.0" encoding="UTF-8"?>
<grammar ns="urn:ietf:params:xml:ns:lost1:slb"
  xmlns="http://relaxng.org/ns/structure/1.0"
  xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
  xmlns:ls="urn:ietf:params:xml:ns:lost1"
  datatypeLibrary="http://www.w3.org/2001/XMLSchema-datatypes">

  <div xmlns:slb="urn:ietf:params:xml:ns:lost1:slb">
    <ls:listServicesByLocation>
      ...
      ...
      <a:documentation>
        Allows requesting the serviceListBoundary by reference or by value
      </a:documentation>

      <define name="serviceListBoundary">
        <element name="serviceListBoundary">
          <ref name="slb:serviceListBoundary"/>
          <choice>
            <value>value</value>
            <value>reference</value>
          </choice>
        </element>
      </define>
    </ls:listServicesByLocation>
  </div>

  <div xmlns:slb="urn:ietf:params:xml:ns:lost1:slb">
    <ls:listServicesByLocationResponse>
      ...
      ...
      <a:documentation>
        Returns the serviceListBoundary by Reference
      </a:documentation>

      <define name="serviceListBoundaryReference">
        <element name="serviceListBoundaryReference">
          <ref name="slb:serviceListBoundaryReference"/>
          <attribute name="source"/>
          <attribute name="serviceListKey"/>
        </element>
      </define>
    </ls:listServicesByLocationResponse>
  </div>
```

```
<div xmlns:slb="urn:ietf:params:xml:schema:lost1:slb">
  <ls:listServicesByLocationResponse>
    ...
    ...
    <a:documentation>
      Returns the serviceListBoundary by Value
    </a:documentation>

    <define name="serviceListBoundary">
      <element name="serviceListBoundary">
        <ref name="slb:serviceListBoundary"/>
        <attribute name="profile"/>
        <ref name="ls:locationInformation"/>
      </element>
    </define>
  </ls:listServicesByLocationResponse>
</div>
```

```
<div xmlns:slb="urn:ietf:params:xml:schema:lost1:slb">

  <a:documentation>
    Request for the serviceListBoundary
  </a:documentation>

  <define name="getServiceListBoundary">
    <element name="getServiceListBoundary">
      <ref name="slb:getServiceListBoundary"/>
      <attribute name="serviceListKey"/>
    </element>
  </define>

</div>
```

```
<div xmlns:slb="urn:ietf:params:xml:schema:lost1:slb">

  <a:documentation>
    Response to getServiceListBoundary
  </a:documentation>

  <define name="getServiceListBoundaryResponse">
    <element name="getServiceListBoundaryResponse">
      <ref name="slb:getServiceListBoundaryResponse"/>
      <attribute name="serviceListKey"/>
      <ref name="slb:serviceListBoundary"/>
    </element>
  </define>

</div>
```

```

        <ref name="ls:path"/>
    </element>
</define>

</div>
</grammar>

END

```

5.2. Namespace Registration

[TOC](#)

This document requests registration of the following namespace (below the LoST namespace defined in [\[RFC5222\]](#) (Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol," August 2008.)) to the IETF XML Registry [\[RFC3688\]](#) (Mealling, M., "The IETF XML Registry," January 2004.):

URI: urn:ietf:params:xml:ns:lost1:slb

Registrant Contact: IETF ECRIT Working Group, Karl Heinz Wolf
(karlheinz.wolf@nic.at)

XML:

BEGIN

```

<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
    "http://www.w3.org/TR/xhtml1-basic/xhtml1-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta http-equiv="content-type"
        content="text/html; charset=iso-8859-1"/>
    <title>LoST serviceListBoundary Namespace</title>
</head>
<body>
    <h1>Namespace for the LoST serviceListBoundary</h1>
    <h2>urn:ietf:params:xml:ns:lost1:slb</h2>
    <p>See <a href="http://www.rfc-editor.org/rfc/rfcXXXX.txt">
        RFCXXXX</a>.</p>
</body>
</html>

```

END

[TOC](#)

6. Acknowledgement

The author would like to thank Henning Schulzrinne for the discussion on the draft.

7. Normative References

[TOC](#)

[RFC5222]	Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, " LoST: A Location-to-Service Translation Protocol ," RFC 5222, August 2008 (TXT).
[RFC5582]	Schulzrinne, H., " Location-to-URL Mapping Architecture and Framework ," RFC 5582, September 2009 (TXT).
[RFC2119]	Bradner, S., " Key words for use in RFCs to Indicate Requirement Levels ," BCP 14, RFC 2119, March 1997 (TXT , HTML , XML).
[RFC3688]	Mealling, M., " The IETF XML Registry ," BCP 81, RFC 3688, January 2004 (TXT).

Author's Address

[TOC](#)

	Karl Heinz Wolf
	nic.at GmbH
	Karlsplatz 1/2/9
	Wien A-1010
	Austria
Phone:	+43 1 5056416 37
Email:	karlheinz.wolf@nic.at
URI:	http://www.nic.at/