Cluster Computing 5, 389–397, 2002

A Common Data Set and Framework for Representing Spatial Location Information in the Internet

MARI KORKEA-AHO and HAITAO TANG
Nokia Research Center, PO Box 407, 00045 Nokia Group, Finland

Abstract. Many organizations are currently working on how to express and provide location information to services and applications in the Internet. Each of them basically specifies their own way. This raises a problem – the various location information formats, services and applications will not be interoperable in the Internet. Interoperability can be achieved if there is a common way of expressing location information. This paper therefore proposes a common data set and an extensible framework of expressing location information in the Internet. The design aims at bridging various existing/proposed location data representation formats, as well as meeting the requirements of existing/proposed location-aware services.

Keywords: location, location-aware services, location-based services, location information, location representation formats

1. Introduction

Due to the increasing availability of position information and the surging demand on location-aware services, the interest towards location-aware services and technologies have grown rapidly in the recent years. The increasing availability of location information has awaked the opportunity for many new services and applications in the Internet, e.g., in the areas of tracking, local information, guidance and navigation, access authorization, resource announcement and discovery, billing, and network management.

Currently many organizations are working on location-related technologies, and how to express and provide location information to services and applications in the Internet. Such organizations are IETF, OpenGIS, 3GPP, LIF, WAP Forum, W3C, etc. Each of them basically specifies its own way of providing and expressing location information to services and applications. This raises a serious problem – the various location information formats, services, and applications will not be interoperable in the Internet.

Therefore, a common extendible way of expressing and transferring location information for services and applications in the Internet is needed. Similar to what is done by the IP protocol, this common way "interconnects" the naturally heterogeneous location systems and applications in the Internet. With such an approach the interoperability between the applications/systems could be enabled. Different applications and services in the Internet can use same data representation and processing methods. As an effort to design a common way the IETF Spatial Location BOF [1] has been initiated, while a lot of contributions are needed to realize the goal.

The work in this paper concentrates on the design of a common data set expressed and encoded in a common way and an extensible framework of expressing location information in the Internet, without considering a common protocol. Already a common way of expressing location information will bring interoperability, and could be reused by different applications. The proposal aims at bridging various existing and proposed data representation formats, as well as meeting the requirements on spatial location information by existing or proposed spatial location-aware services.

The rest of the paper is organized as follows. In section 2, we discuss different types and sources of location information, and existing/proposed ways of expressing such information. In section 3, the requirements on location information by different applications are analyzed. In section 4, a common data set and a framework for location information in the Internet are proposed and analyzed. After this, security issues are briefly discussed in section 5. In section 6 the work is concluded.

2. Locations and some existing spatial location expressions

2.1. Locations

A location is a place where an object or its host is "physically" situated. The location can be expressed in different ways using different reference frames. The locations consist of, e.g., absolute (spatial) location, network location, descriptive location, relative location, etc.

Absolute (spatial) location is the location of a physical object in the real world, expressed via a 2- or 3-dimensional coordinate system using a certain geodetic datum. The location can be expressed with coordinate systems, such as, Latitude–Longitude–Altitude, *XYZ* of Earth Centered–Earth Fixed (ECEF), Zone numbers of Universal Transverse Mercator (UTM), etc. There are hundreds of different geodetic datum in use around the world. One widely used system is WGS-84 that is commonly used by the GPS system [2,3].

Network location is the location of a physical object in a computer or telecommunication network. Examples are the

IP-address of a computer in the Internet, the ISDN number of phone in a PSTN network, etc.

Descriptive location is a location described through a language other than coordinate system or network address/identifier. Examples of descriptive locations are: postal address, zip code, building number, country code, personal addresses (my home, my cottage). Relative location is a specific type of descriptive location, where the location of an object is described relative to some other object, e.g., 100 meters from the store, the building next to the tower, close to me, nearest shop, etc. Generally, a descriptive location can be mapped to an absolute (spatial) location or a network location.

The different ways of expressing location cover different needs. With the help of different transformation rules [2] and services one can convert between the different location formats.

2.2. Location information for services and applications in the Internet

There are different positioning methods available for determining the location of an object. This includes different methods of satellite based positioning (e.g., GPS [4], Glonass), positioning in mobile networks (e.g., in GSM [5]), and local positioning (using, e.g., IR, RF or Bluetooth). In static objects such as routers, the location can also be preconfigured. In mobile objects the location can also be given manually.

Location information of an object is to be available to authorized Internet applications via different interfaces. Some proposed interfaces are: (1) interfaces with mobile networks providing the location of a mobile terminal, (2) interfaces in mobile terminals providing the location (determined by some external positioning device, the mobile network, or manually by the user), (3) interfaces towards local positioning systems, (4) interfaces in static devices with manually coded location, etc.

The proposed common location data representation and framework enable a common way of expressing location data by any relevant Internet services and applications. This builds interoperability among the heterogeneous sources and Internet applications/services.

2.3. Existing spatial location expressions

There are many existing or proposed location expressions from a number of organizations (e.g., IETF, OpenGIS, 3GPP, LIF, WAP Forum, and W3C). Some of them are listed below:

- Expression standardized for GSM and UMTS to be used internally in the mobile networks (called here "3GPP") [6].
- An interface towards mobile networks in consideration by LIF [7].
- The Geography Markup Language (GML) by the OpenGIS Consortium [8].
- NaVigation Markup Language (NVML) [9] and Point Of Interest eXchange Language (POIX) [10] submitted to the W3C.
- GeoTags for HTML resource discovery [11,12].
- National Marine Electronics Association (NMEA) interface and protocol [13] often used by GPS receivers.
- VCard and ICalendar [14,15] include elements to specify position.
- A Means for Expressing Location Information in the Domain Name System (DNS-LOC) [16].
- Proposed Simple Text Format for the Spatial Location Protocol (SLoP) [17].

In brief, most of the formats express location with latitude, longitude, using WGS-84 as reference datum. GML, LIF, NVML, and POIX also enable expressions using other coordinate systems and reference datum. Some allow altitude, if the data is available. In the location expressions, altitude usually means the height above WGS-84 reference ellipsoid, while it is unclear in some cases.

Most of the formats focus on the specification of the location of a point object, whereas others include also the expression of object shapes (3GPP, LIF, and GML). In DNS-LOC and NVML radial size of the object can be defined.

When the accuracy for estimating a location is defined, it is mostly expressed as horizontal and vertical error. Though, the 3GPP proposal includes more complex accuracy descriptions.

LIF, POIX, NMEA, and 3GPP include also fields for velocity/speed. It is expressed as horizontal speed in all the cases except 3GPP. The 3GPP proposal defines horizontal velocity (horizontal speed + bearing) and vertical velocity (vertical speed + vertical direction).

Direction of movement is also included in LIF, POIX, and NMEA, using true and/or magnetic North. POIX and NMEA include possibility to define the course as well.

3. Location expression requirements from location-aware services

In order to determine a common data set it is important to evaluate the requirements of existing and planned location-aware services in the Internet. In this section, we review the data requirements of location-aware services we have identified. Since we define a common data set for the Internet, the analysis does not cover applications implemented as part of mobile networks (e.g., GSM, UMTS, etc.).

3.1. Types of location-aware services

We have identified following types of services, covering the application, network application, and network infrastructure levels based on input from [5]:

- *Finding (Information Services).* For example, yellow pages, and point-of-interest information services.
- *Guidance (Navigation).* This includes services to show current position/location, and how to get to some target location.

- *Notifications.* For example, targeted ads, traffic alerts, weather services, and guided tours. Information is notified to user when he/she is in a specific region.
- *Information Memorizing and Association.* Adding and storing location information to data, e.g., html pages, e-mails, calendar entries, photos, maps, etc.
- *Tracking and Resource Management Services.* Including emergency services, fleet management, equipment management, personnel management, tracking (people, personal belongings, stolen vehicles, and deliveries), etc.
- *Authorization.* Authorization to resources, information, spaces according to location.
- *Location Specific Resource Announcement and Discovery.*
- *Location Sensitive Billing.*
- *Network Management.* Including various networking and protocol optimizations, e.g., enhancing network scalability, network node handoffs, etc.

3.2. Location data used by the services

When analyzing what location-aware services might require as input, we have considered following things: what kind of location information the services need as input, if error estimates are of any help, and what other information would be valuable? The analysis results are presented in table 1.

Through the different applications in table 1, it appears that most of them need absolute (spatial) location as input. This is also supported by the fact that most existing location measurement systems provide this information. Descriptive location information is generally created by manual input or via transformation services. The notification, tracking, resource discovery, authorization, billing services could however make use of indications of objects leaving and entering specific regions as well.

Since the applications are earth based, a geocentric coordinate system should be used. It also appears that most services will not need altitude information. The use of altitude enables, of course, the positioning on, e.g., different levels in buildings, and should be provided if available. If errors of the measurements are available they could be used, but they do not bring much added value to most of the services.

We have also investigated the added value of being able to describe the size and shape of an object. This information could principally be used in two ways; firstly to describe the object which is positioned in order to determine what region it is covering (e.g., in finding, guidance, notification, tracking, authorization, resource discovery, billing and management services), secondly to indicate the region of interest or object to attach information to (finding information and information memorizing and association). Since most of the objects for positioning are of minor size (<10 m), the size and shape of an object usually do not have significance for the lo-

Table 1
Analysis on location data required by different services, where “abs.” means absolute location, “des.” means descriptive location, “rel.” means relative location, “opt.” means optional, “reg.” means region, and “(z)” means that altitude is optional.

Application	Required data			
	Location	Accuracy	Error	Other possible data
Finding (Information Services)	abs.: x, y, (z) des: e.g., region, address rel: e.g., near to me	reg. – 1 m	opt.	orientation
Guidance (Navigation)		50 – 1 m	opt.	speed, direction, course
(a) one time	abs.: x, y, (z) des: address			
(b) continuos	abs.: x, y, (z)	50 – 1 m	opt.	speed, direction, course
Notifications				
(a) using location	abs.: x, y, (z)	reg. – 1 m	opt.	orientation, direction, speed
(b) notif. entering/leaving region	in/out region	–	–	–
Information Memorizing and Association				
(a) one time	abs.: x, y, (z) des: address	reg. – 1 m	opt.	orientation
(b) continuos	abs.: x, y, (z)	50 – 1 m	opt.	orientation, direction, speed
Tracking and Resource Management Services				
(a) one time	abs.: x, y, (z)	reg. – 1 m	opt.	direction, speed
(b) continuos	abs.: x, y, (z)	50 – 1 m	opt.	direction, speed
(c) notif. entering/leaving region	in/out region	–	–	–
Authorization	abs.: x, y, (z) in/out region	reg. – 1 m	opt.	–
Location Specific Resource Announcement and Discovery	abs.: x, y, (z) in/out region	reg. – 1 m	opt.	–
Location Sensitive Billing	abs.: x, y, (z) in/out region	reg. – 1 m	opt.	–
Network Management	abs.: x, y, (z)	reg. – 1 m	opt.	direction, speed

cation of the object. In fact, size and shape can be understood as attributes associated to a location rather than location itself. It may be the reason why location measurement approaches are point-based.

It is quite evident that in addition to location information it is important to attach the timestamp of measurement to the location. This can be essential to the processing and management of location information. Other information that could bring added value to services include the orientation of the object, its moving direction, intended course, and speed.

4. A common data set and an extendible framework for location information in the Internet

The idea with a common data set of location information is to enable Internet services and applications to express location information in an interoperable way so that the location data from various sources can be used. We propose here such a set. Based on the common set, Internet applications and services can use same processing/parsing methods.

There is already an advantage if different applications and application protocols are able to express location in an interoperable way. When used in combination with a common protocol one can create further an interoperable infrastructure in the Internet. This has been one of the objectives of the SLoP activity in the IETF [1].

Since there are so many other existing expressions, in addition to the proposed common data set, there is a need of a higher level framework that could support/carry different types of location expressions.

The advantage of such a framework is that it simplifies the interpretation and processing of the location data, while it enables generic ways of expressing and identifying the different location data element sets. With the help of a framework it is possible to express the same location in different ways, or add extensions to a certain location expression. If each of the location expression has a unique id identifying the data set, clear rules for transformations can be guaranteed. If we want to pursue guaranteed interoperability the common data set should be mandatory in each frame.

4.1. Common data set

In this section we propose a common data set. The proposal is based on identified elements important to applications, and on the available data from different devices and interfaces. The proposal is presented in table 2. Some of the elements are mandatory, while others are optional.

We have considered whether the common data element set should include only absolute point locations or if it should also include extra elements for radial size (as, e.g., in DNS-LOC) and/or different shape types (as in GML). The conclusion is that there is no need of the extra elements in the common set. These extra elements are very complex in the sense of their measurement, expression, and utilization. In addition, there may not be feasible solutions on some of them. Furthermore, most applications do not need the extra elements.

Table 2
The elements of the common data set.

Datum	– WGS-84	(Mandatory)
Coordinates	– Latitude	(Mandatory)
	– Longitude	(Mandatory)
	– Altitude above WGS-84 reference ellipsoid	(Optional)
	– Altitude above mean sea level	(Optional)
Location Accuracy	– Horizontal accuracy, by radius of a circle from the positioned point	(Optional)
	– Altitude accuracy, by range from the positioned point	(Optional)
Time	– Real time of the measurement/fix	(Mandatory)
Speed	– Ground speed	(Optional)
	– Vertical speed	(Optional)
Direction	– Direction of movement	(Optional)
Course	– Direction from the current position to a defined destination	(Optional)
Orientation	– Horizontal orientation	(Optional)
	– Vertical orientation (pitch)	(Optional)
Unspecified Attributes	– Attributes enabling some extensibility	(Optional)

Therefore, shape and size information could instead be added as an own data set or attribute in the extensible framework when needed for a specific purpose.

Here are the explanations of the elements included into the common data set:

- *Coordinates and Datum*
  When reviewing the various existing interfaces and data representation formats, we find that most of them support coordinates expressed in latitude, longitude, and altitude (optional) using WGS-84 datum. Thus we propose to use these in the common data set. In order to keep the common data set simple, no other datum or coordinate systems are supported. We have chosen to enable the optional altitude to be expressed both as the WGS-84 reference ellipsoid and mean sea level as reference.
- *Location Accuracy*
  Location accuracy is the estimation/measurement error of a location. The different interfaces include different types of accuracy information. We propose to include the most common way to express this, i.e., horizontal accuracy, by circle of radius from the positioned point, and height accuracy, by range from the positioned point.
- *Time*
  Time is the time of a measurement/fix of a location of an object. It is an important factor for location information. With the help of the time it is easier to manage location information and it enables different kind of approximations. It is a mandatory element.
- *Speed*
  Speed is an optional element and is indicated as horizontal ground and vertical speed. This expression is chosen because many systems are able to indicate horizontal ground and vertical speed.

Table 3
Syntax of the default data elements.

Element	Expression format	Example
Coordinates		
– Latitude	[N/S]degrees.minutes.seconds.f decimal fraction f in arbitrary length, range of degrees [0–90]	N60.08.00.232
– Longitude	[E/W]degrees.minutes.seconds.f decimal fraction f in arbitrary length, range of degrees [0–180]	E25.00.00.331
– Altitude above datum	[+|−]x.f meter from WGS-84 reference ellipsoid, + above, − below, decimal fraction f in arbitrary length	+12
– Altitude above mean sea level	[+|−]x.f meter from mean sea level, + above, − below, decimal fraction f in arbitrary length	+10
Location Accuracy		
– Horizontal accuracy	by circle of radius from the positioned point in x.f meter, where f decimal fraction in arbitrary length	50.0
– Height accuracy	in x.f meter, where decimal fraction in arbitrary length	2.5
Time [19,20]		
– Real time of the measurement/fix	YYYY-MM-DDThh:mm:ss.sTZD, where YYYY = four-digit year MM = two-digit month (01 = January, etc.) DD = two-digit day of month (01 through 31) hh = two digits of hour (00 through 23) mm = two digits of minute (00 through 59) ss = two digits of second (00 through 59) s = one or more digits representing a decimal fraction of a second TZD = time zone designator (Z or +hh:mm or −hh:mm)	1999-08-15T11:16:31.0+2:00
Speed		
– Ground speed	x.f [m/s | km/h | mph | knot], where f arbitrary decimal fractions, default m/s	2.0 m/s
– Vertical speed	x.f [m/s | km/h | mph | knot], where f arbitrary decimal fractions, default m/s	1.0 m/s
Direction	magnetic/true direction, 360 degrees from North clockwise	
	[M | T][0–360].f, where f fractional degrees in arbitrary length, M default	M240
Course	magnetic/true course, 360 degrees from North clockwise	
	[M | T][0–360].f, where f fractional degrees in arbitrary length, M default	M30
Orientation		
– Horizontal	magnetic/true orientation, 360 degrees from North clockwise	M240
	[M | T][0–360].f, where f fractional degrees in arbitrary length, M default	
– Vertical (pitch)	[+|−][0–180].f, where f fractional degrees in arbitrary length	0
Unspecified Attributes	attribute value(s)	car_orientation 360,40,20

- *Direction*
  Direction indicates the direction of movement. It is expressed in a 2-dimensional (horizontal) frame indicated by the magnetic (or true) North.
- *Course*
  Course indicates the direction from the current position to a defined destination. It is expressed in a 2-dimensional (horizontal) frame indicated by the magnetic (or true) North.
- *Orientation*
  Orientation describes the orientation of the positioned object. Orientation is often given with a local coordinate system as reference. Since this reference frame can be different for different objects, it will be difficult to make a common expression based on this. One possibility would be to attach an object type indicating directly the used reference framework. Instead of such a solution, we propose a method where the orientation is expressed in a 2-dimensional (horizontal) frame indicated by the magnetic (or true) North, and a vertical element expressed by the angle between horizontal plane and the main axis of the object.
- *Unspecified Attributes*
  We have incorporated an unspecified element into the common set to include some application specific elements. The attributes should be relevant for the location data set

Table 4
The XML DTD for the common data set.

```
<!-- slo_default.dtd -->
<!ELEMENT SLO (POS, ALT?, ALT_MSL?, H_ACC?,
  V_ACC?, TIME, G_SPEED?, V_SPEED?, DIR?,
  COURSE?, H_ORIENT?, V_ORIENT?, X_ATTR?)>
<!-- Coordinates -->
<!ELEMENT POS (LAT, LONG)>
<!ELEMENT LAT (#PCDATA)>
<!ELEMENT LONG (#PCDATA)>
<!-- Altitude -->
<!ELEMENT ALT (#PCDATA)>
<!ELEMENT ALT_MSL (#PCDATA)>
<!-- Location Accuracy -->
<!ELEMENT H_ACC (#PCDATA)>
<!ELEMENT V_ACC (#PCDATA)>
<!-- Time -->
<!ELEMENT TIME (#PCDATA)>
<!-- Speed -->
<!ELEMENT G_SPEED (#PCDATA)>
<!ATTLIST G_SPEED unit (ms|kmh|mph|knot) "ms">
<!ELEMENT V_SPEED (#PCDATA)>
<!ATTLIST V_SPEED unit (ms|kmh|mph|knot) "ms">
<!-- Direction -->
<!ELEMENT DIR (#PCDATA)>
<!-- Course -->
<!ELEMENT COURSE (#PCDATA)>
<!-- Orientation -->
<!ELEMENT H_ORIENT (#PCDATA)>
<!ELEMENT V_ORIENT (#PCDATA)>
<!-- Unspecified Attributes -->
<!ELEMENT X_ATTR (PARAM)>
<!ELEMENT PARAM (VALUE*)>
<!ATTLIST PARAM name CDATA #REQUIRED>
<!ELEMENT VALUE (#PCDATA)>
```

and not conflict with defined/existing attributes. This field should be used with consideration.

4.2. Syntax of the elements in the common data set

The way of expressing each data element in the common data set needs to be defined. Some of the existing data formats allow different optional ways to express the data elements and include syntax information. However, in order to keep processing as simple as possible we prefer one single way for expression only. Table 3 summarizes our proposal. A more formal syntax definition using the ABNF grammar for the common data set can be found in appendix A.

4.3. Encoding of data elements

The data elements can be encoded in many different ways, e.g., text based attribute-value pairs, binary, MIME, XML, etc. In order to enable interoperability, again, we need a common way of encoding the parameters. We propose XML. The advantages of XML are that the encoding is easily understandable, human readable, and standard tools and parsers can be used. In addition to this, many of the other proposals make use of XML. A possible disadvantage of using XML is that it is quite verbose.

In table 4, the XML-encoding of the common data set elements is presented with help of the DTD of the XML-document.

Table 5
An example of an XML-encoded location.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE IETF_SLO_Default PUBLIC "-//IETF//
  SLO default//EN" "slo_default.dtd">
<SLO>
  <POS>
      <LAT>N60.08.00.232</LAT>
      <LONG>E025.00.00.331</LONG>
  </POS>
  <ALT>+12</ALT>
  <ALT_MSL>+10</ALT_MSL>
  <H_ACC>50.0</H_ACC>
  <V_ACC>2.5</V_ACC>
  <TIME>1999-08-15T11:16:31.0 +2:00</TIME>
  <G_SPEED unit=ms>2.0</G_SPEED>
  <V_SPEED unit=ms>1.0</V_SPEED>
  <DIR>M240</DIR>
  <COURSE >M30</COURSE>
  <H_ORIENT>M240</H_ORIENT>
  <V_ORIENT>0</V_ORIENT>
  <X_ATTR>
     <PARAM name="car_orientation">
        <VALUE>360</VALUE>
        <VALUE>40</VALUE>
        <VALUE>20</VALUE>
     </PARAM>
  </X_ATTR>
</SLO>
```

Table 5 presents an example of an XML-encoded location.

4.4. Extendible framework in XML

A framework enables to express the same location in different ways, or add extensions to a certain expression. That is, one location expression frame can include several different location expression subsets. We have been considering an XML- or MIME-based framework. The advantage of an XML-based framework is that we could use the same parser and processing methods, the disadvantage again is that we are able to incorporate only XML-based data sets. The advantage of the MIME-based set is that we could include any type of encoded data in it, but on the other hand this requires a multitude of parsers, etc., possibly adding to system complexity. We thus propose an XML-based framework.

The framework consists of a structure document that can include different location data expression subsets. Each set is represented by its document type definition (DTD). Each set is further identified by an identifier (e.g., the system or public identifier of the document, or the XML-root element). The identifier helps to identify the data set and can simplify the processing and transformation of the data. There could be several generally known data sets (e.g., the common data set "slo_default", enabling guaranteed interoperability if used in all frames), as well as application specific ones. In order to avoid conflicts in the structure document, the different data sets should include unique XML-elements. Element collisions can be avoided by using the XML-namespace mechanism [21].

Table 6
An example of the XML-based framework for location data.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE LOC_FRAME [
<!ENTITY % slo_default_dtd public PUBLIC
 "-//IETF//SLO default//EN""slo_default.dtd">
<!ENTITY % my_loc_dtd public PUBLIC
 "-//MY_LOC//My Location//EN" "my_loc.dtd">
%slo_default_dtd;
%my_loc_dtd;
<!ELEMENT LOC_FRAME (SLO, MY_LOC)>
]>
<LOC_FRAME>
    <SLO>
      ...
    </SLO>
    <MY_LOC>
      ...
    </MY_LOC>
</LOC_FRAME>
```

Since we assume that the XML-parser performs validation, the framework needs to include the references to the DTDs of the subsets. We assume that the receiving party has the required DTDs, otherwise a URL pointing to the DTD should be available. One way of creating the framework is to create a frame document ("LOC_FRAME") that incorporates the DTDs of the different location representation subsets. Below is an example, where the framework incorporates two subsets, the "slo_default" subset and "my_loc" subset (table 6).

Another option is to include the different DTDs in an external DTD (e.g., SLO_MY_LOC.dtd) and then reference the external DTD in the location representation document. Another possibility to be further studied is to use the mechanisms proposed in the XHTML modularization [22].

5. Security considerations

Location information is potentially private or sensitive even though some parties (such as shops) like to release their location information to the public. The authors believe that location information should be delivered based on the policy set to the location information. In addition, certain security mechanisms should be used to protect the location information, if required (as most of the cases).

6. Conclusions

Location information has been expressed in very many different ways. With the help of a common default set we can achieve interoperability between different applications and systems in the Internet.

By analyzing various existing/proposed data representation formats, and spatial location-aware services, the authors suggest that the default set should include (1) as mandatory elements: the location of the object expressed with latitude, longitude, and altitude (recommended – should be provided when available) using the WGS-84 datum, and time of measurement, (2) as optional elements: location accuracy, speed, direction, course, orientation, and unspecified attributes.

In order to keep processing simple, the authors propose a single way for expressing and encoding the default data set, opposite to some other proposals that allow different optional ways. Due to the existence of various location formats, the authors further propose a higher level XML-based framework to support/carry the different types of expressions. The advantage of such a framework is that it simplifies the interpretation and processing of the location data, while it enables generic way of specifying and identifying the different subsets. It also enables extendibility and allows a location to be expressed in multiple ways.

Acknowledgements

The authors would like to thank James Polk (Cisco) and Kenji Takahashi (NTT) for their valuable comments.

Appendix A. Formal syntax of common data set

The syntax is specified with ABNF grammar (IETF RFC2234).

```
SLO = Coordinate Delimiter
      [Location_Accuracy Delimiter]
      Time Delimiter
      [Speed Delimiter]
      [Direction Delimiter]
      [Course Delimiter]
      [Orientation Delimiter]
      [X_Attribute]
Delimiter = <any delimiter string>          ; Delimiter depending on the coding
Coordinate = Latitude Delimiter Longitude [Delimiter (Altitude_WGS84 | Altitude_Sea)]
Latitude = ("N" | "S") Degree "." Minute "." Second "." Fraction
Degree = "0"–"90"
Minute = "0"–"60"
Second = "0"–"60"
Fraction = *Digit
Digit = "0"–"9"
```

The syntax is specified with ABNF grammar (IETF RFC2234). (Continued.)

```
Longitude = ("E" | "W") Degree_1 "." Minute "." Second "." Fraction
Degree_1 = "0"–"180"
Altitude_WGS84 = ("+" | "−") Meter "." Fraction
                                        ; height in meter from WGS-84 reference ellipsoid
Meter = *Digit
Fraction = *Digit
Altitude_Sea = ("+" | "−") Meter "." Fraction
                                        ; height in meter from mean sea level
Location_Accuracy = [Horizontal_Accuracy Delimiter] [Height_Accuracy]
Horizontal_Accuracy = Meter "." Fraction
Height_Accuracy = Meter "." Fraction
Time = YYYY "−" MM "−" DD "T" hh ":" mm ":" ss "." s TZD
YYYY = 4*4Digit
MM = "01"–"12"
DD = "01"–"31"
hh = "00"–"23"
mm = "00"–"59"
ss = "00"–"59"
s = *Digit
TZD = "Z" | (("+" | "−") hh:mm)         ; where Z means zero meridian
Speed = [Ground_speed Delimiter] [Vertical_speed]
Ground_speed = *Digit "." *Digit SP ("m/s" | "km/h" | "mph" | "knot"); default: m/s
Vertical_speed = *Digit "." *Digit SP ("m/s" | "km/h" | "mph" | "knot"); default: m/s
Direction = Magnetic_direction | True_direction
Magnetic_direction = "M" Degree_2 "." Fraction
Degree_2 = "0"–"360"
True_direction = "T" Degree_2 "." Fraction
Course = Magnetic_direction | True_direction
Orientation = Horizontal_orientation | Vertical_orientation
Horizontal_orientation = Magnetic_direction | True_direction
Vertical_orientation = ("+" | "−") Degree_1 "." Fraction
X_Attribute = <a string of an indicated/default character-set>
```


References

[1] IETF, Spatial Location BOF, http://www-nrc.nokia.com/ietf-spatial/

[2] P.H. Dana, Coordinate Systems Overview, Department of Geography, University of Texas at Austin (1997), http://www.colorado.Edu/geography/gcraft/notes/coordsys/coordsys_f.html

[3] P.H. Dana, Geodetic Datum Overview, Department of Geography, University of Texas at Austin (1997), http://www.colorado.Edu/geography/gcraft/notes/datum/datum_f.html

[4] B.W. Parkinson and J.J. Spilker, Jr. (eds.), *Global Positioning System: Theory and Applications*, Vol. 1 (American Institute of Aeronautics and Astronautics, Inc., Washington, DC, 1996).

[5] T. Rantalainen et al., Evolution of location services in GSM and UMTS networks, in: *Proceedings of the 3rd International Symposium on Wireless Personal Multimedia Communications (WPMC'00)*, Bangkok, Thailand, November 2000, pp. 1027–1032.

[6] 3rd Generation Partnership Project, Technical Specification Group Core Network, Universal Geographical Area Description (GAD), Release 1999, Technical Specification, 3G TS 23.032 V3.1.0, March 2000.

[7] Definition of a Mobile Location API, Contribution to Location Interoperability Forum (LIF), API Specification, v. 0.2, June 2000.

[8] R. Lake and A. Cuthbert (eds.), Geography Markup Language (GML) v1.0, OGC Document Number: 00-029, May 2000, http://www.opengis.org/techno/specs/00-029.pdf

[9] Sekiguchi et al., NaVigation Markup Language (NVML), W3C Note, August 1999, http://www.w3.org/TR/NVML

[10] K. Hiroyuki and K. Tomihisa, POIX: Point Of Interest eXchange Language Specification, W3C Note, June 1999, http://www.w3.org/TR/poix

[11] A. Daviel, Geographic registration of HTML documents, Internet draft, work in progress, April 2000, http://geotags.com/geo/draft-daviel-html-geo-tag-03.txt

[12] A. Daviel, Geographic extensions for HTTP transactions, Internet draft, work in progress, April 2000, http://geotags.com/geo/draft-daviel-http-geo-header-02.txt

[13] P. Bennett, The NMEA FAQ, version 6.3, April 2000, http://vancouver-webpages.com/pub/peter/nmeafaq.txt

[14] Internet Mail Consortium, vCard – The Electronic Business Card Version 2.1, September 1996, http://www.imc.org/pdi/vcard-21.txt

[15] F. Dawson and T. Howes, vCard MIME Directory Profile, IETF RFC 2426, September 1998, http://www.imc.org/rfc2426

[16] F. Dawson and D. Stenerson, Internet Calendaring and Scheduling Core Object Specification (iCalendar), IETF RFC 2445, November 1998, http://www.imc.org/rfc2445

[17] C. Davis et al., A Means for Expressing Location Information in the Domain Name System, IETF RFC 1876, January 1996, ftp://ftp.funet.fi/pub/doc/rfc/rfc1876.txt

[18] R. Mahy, A Simple Text Format for the Spatial Location Protocol (SLoP), Internet draft, work in progress, July 2000, http://search.ietf.org/internet-drafts/draft-mahy-spatial-simple-coord-00.txt

[19] M. Wolf and C. Wicksteed, W3C Note, Date and Time Formats, September 1997, http://www.w3.org/TR/1998/NOTE-datetime-19980827

[20] M. Kuhn, A Summary of the International Standard Date and Time Notation, http://www.cl.cam.ac.uk/~mgk25/iso-time.html

[21] Bray et al., Namespaces in XML, in: World Wide Web Consortium, January 1999, http://www.w3.org/TR/1999/REC-xml-names-19990114

[22] Adams et al., Modularization of XHTML™, in: World Wide Web Consortium, October 2000, http://www.w3.org/TR/2000/CR-xhtml-modularization-20001020/xhtml-modularization-20001020.html

Mari Korkea-aho was an R&D staff member in Nokia Research Center (Helsinki) when writing this publication. She received her M.Sc. degree in information technology at the Helsinki University of Technology in 1995. She is currently working as a senior technical consultant at Digiscope Oy.
E-mail: mari.korkea-aho@iki.fi

Haitao Tang is a senior R&D staff member in Nokia Research Center (Helsinki). He received his M.Sc. degree in communication and electronic systems in 1986, and his Doctor of Technology degree in computer science and engineering in 1998. He has continuously worked on data networking and applications since 1992.
E-mail: haitao.tang@nokia.com